Translating and the Computer 38

17-18 November 2016
One Birdcage Walk, London

Proceedings

ISBN 978-2-9701095-0-1

Editions Tradulex, Geneva

These proceedings are downloadable from www.tradulex.com
These proceedings are downloadable from www.asling.org

Acknowledgements

AsLing wishes to thank and acknowledge the support of the sponsors of TC38:

Gold Sponsors

Silver sponsor

Exhibitor

Media Sponsor

Bag Sponsor

Preface

For the past 38 years the international conference Translating and the Computer has been a leading and distinctive forum for academics, users, developers and vendors of computer aids for translators and, increasingly, other translation technology tools. The event offers translators, interpreters, researchers and business people from translation companies, international organisations, universities and research labs, as well as freelance professionals the opportunity to discuss the latest developments and trends and exchange ideas. AsLing (International Association for Advancement in Language Technology), which took over the organisation of this conference in 2014, is proud to present the proceedings of Translating and the Computer 38 Conference (TC38), taking place in London on 17 and 18 November 2016.

This year's conference continues the tradition of hosting quality speakers and panellists on a wide range of topics related to technology for translators and terminologists, as well as for interpreters. These range from translation tools, through machine translation, translation workflow, hybrid translation technologies, subtitling, terminology, standards and quality assessment. This year we are proud to continue the focus on the new technologies concerning interpreting where revolutionary changes are under way. We are confident that the e-proceedings featuring these contributions, accepted after a competitive reviewing process, will be an important reference and stimulus for future work. We are delighted to present our keynote speakers: Dieter Rummel and Henry Liu, representing respectively the by far largest and most complex translation service, The Directorate-General for Translation of the European Commission, and the world's premier professional organization for translators, FIT, the International Federation of Translators. We are also confident that you will find that all the presentations and posters, as well as the panels and workshops, will provide interesting user perspectives and opportunities for inspiring discussions.

We would like to thank all those who submitted proposals to the conference and all the authors who produced full versions of accepted papers for the proceedings. A special thank-you also to all the delegates who have come from so many countries to attend this conference and thus provide a living acknowledgement of this distinctive event.

We are also grateful to the members of the Programme Committee who carefully reviewed all the submissions: Juanjo Arevalillo, David Chambers, Gloria Corpas Pastor, Joanna Drugan, David Filip, Sarah Griffin-Mason, Bruno Pouliquen, Paola Valli, Nelson Verástegui and David Verhofstadt. Many thanks to our publication chair Ivelina Nikolova for producing these e proceedings. A big thank-you also goes to our Technical Advisor Jean-Marie Vande Walle.

Last but not least, a big thanks to our sponsors.

Conference Chairs
João Esteves-Ferreira, Juliet Margaret Macan, Ruslan Mitkov, Olaf-Michael Stefanov

London, November 2016

Conference Chairs and Editors of the Proceedings

João Esteves-Ferreira, Tradulex - International Association for Quality Translation
Juliet Macan, independent translation technology consultant
Ruslan Mitkov, University of Wolverhampton
Olaf-Michael Stefanov, JIAMCATT, United Nations (ret.)

Programme Committee

Juan José Arevalillo, Hermes Traducciònes
David Chambers, World Intellectual Property Organization (ret)
Gloria Corpas Pastor, University of Malaga
Joanna Drugan, University of East Anglia
David Filip, CNGL / ADAPT
Sarah Griffin-Mason, Institute of Translation and Interpreting
Bruno Pouliquen, World Intellectual Property Organization
Paola Valli, TAUS and University of Trieste
Nelson Verástegui, International Telecommunications Union (ret.)
David Verhofstadt, International Atomic Energy Agency (IAEA)

Conference Management

Juliet Margaret Macan, Coordinator
Olaf-Michael Stefanov, Conference management system and speaker coordinator

Technical Advisor

Jean-Marie Vande Walle

Publication Chair

Ivelina Nikolova, Bulgarian Academy of Sciences

Table of Contents

Translating and the Computer TC 38

At One Birdcage Walk, Westminster, London (UK)

Day-1: Thursday, 17 November 2016

Morning Session

8:30 - 9:00 **Registration** *in the Marble Hall and Gallery*

Lecture Theatre
Ground level

Morning Session Chair:
Olaf-Michael Stefanov

9:00 - 9:15 **Introductions**
AsLing President – João Esteves-Ferreira
TC38 Coordinator – Juliet Margaret Macan

9:15 - 10:00 **Sponsors' Thought Leadership talks**

9:15 - 9:30 Focusing on Tighter Integration of CAT Tools with Corpora
Miloš Jakubíček, on behalf of our Gold Sponsor, Lexical Computing

9:30 - 9:45 Selling Translation Online. A Path to Success
Emanuele Caronia, on behalf of our Gold Sponsor, MateCat

9:45 - 10:00 The Future of Translation Technology
Massimo Ghislandi, on behalf of our Silver Sponsor, SDL

10:00 - 10:45 **KEYNOTE:**
"Will Curiosity Kill the CAT? – Thoughts on the Future of the Computer Assisted Translation Environment"
***Dieter Rummel**, European Commission, DG Translation, Head of Unit – Informatics*

10:45 - 11:15 **Break**
in the Marble Hall and Gallery

11:15 - 11:45 Automated Detection and Correction of Errors in Real-time Speech-to-text: a Research Approach
Lindsay Bywood and Andrew Lambourne

11:45 - 12:15 Translation Quality Evaluation of MWEs from French into English Using an SMT System
Emmanuelle Esperança-Rodier

12:15 - 12:45 What's in a Name?
Jon Riding and Neil Boulton

12:45 - 14:10 **Lunch**
in the Marble Hall and Gallery

Education Room
On Lower ground floor

11:00 - 11:45 *Workshop*
Calculating the Percentage Reduction in Translator Effort when Using Machine Translation
Andrzej Zydroń

11:45 - 12:45 *SDL - Silver Sponsor Workshop:*
Building your Ideal Translation Environment with Apps and APIs from the SDL AppStore
Clementine Tissier, SDL

13:30 - 13:50 *Poster*
How Translators Can Improve Multilingual Terminology in a Link: Teaching Case Study Examples
Rocio Palomares-Perraut and Carmen Gomez-Camarero

Translating and the Computer TC 38

At One Birdcage Walk, Westminster, London (UK)

Day-1: Thursday, 17 November 2016

Afternoon Session

	Lecture Theatre *Ground level*	Education Room *Both rooms on Lower ground floor*	Energy Room *Both rooms on Lower ground floor*
	Afternoon Session Chair: *Ruslan Mitkov*		
14:10 - 14:40	Drawing a Route Map of Making a Small, Domain-specific, Parallel Corpus for Translators and Beyond *Xiaotian (Fred) Guo*	A Case Study of German into English by Machine Translation: to Evaluate Moses Using Moses for Mere Mortals *Roger Haycock*	14:00 - 17:00 *MateCat - Gold Sponsor Workshop* • Translated and MateCat • What is MateCat? • Translating with MateCat: the fast way • Advanced features • Outsourcing with MateCat • Q&A • Hands-on: practical session on MateCat Workshop participants will be issued a certificate by MateCat. *Annalisa De Petra and Daniele Cocozza*
14:40 –16:15	*Panel debate* To Align or Not to Align? Is it Useful for a Translator to Use Alignment Tools? To create what? Corpora or TMs? *Panellists: Gloria Corpas Pastor, Miloš Jakubíček, Balazs Kis, and Andrzej Zydroń* *Moderator: Juliet Macan*	14:45 - 15:25 *Workshop* The Art of Subtitling within the European Institutions *Ayten Dersan*	
		15:30 - 16:10 *Workshop* The Annotation System *Ronan Martin*	
16:15 - 16:45	**Break** *in the Marble Hall and Gallery*		
16:45 - 17:15	Potential Impact of QT21 *Eleanor Cornelius*		
17:15 - 17:30	Honorary Member Ceremony ***President & Chairs***		
17:30	**Close of Day-1**		

Evening 19:30 Networking Gala Dinner - at Piccolino Restaurant – Heddon Street.

Translating and the Computer TC 38

At One Birdcage Walk, Westminster, London (UK)

Day-2: Friday, 18 November 2016

Morning Session

8:30 - 9:00 **Registration** *in the Marble Hall and Gallery*

Lecture Theatre
Ground level

Morning Session Chair:
Juliet Margaret Macan

9:00 - 9:30 How to Configure Statistical Machine Translation with Linked Open Data
Ankit Srivastava

Education Room
On Lower ground floor

9:00 - 9:30 Can you Trust a TM? Results of an Experiment Conducted in November 2015 at CenTraS @ UCL.
Daniela Ford

9:30 - 10:20 **KEYNOTE:**
"Asset Bubbles, Derivatives, Crisis and Translation. But I won't talk about Brexit!"
***Henry Liu**, President, International Federation of Translators (FIT)*

10:20 - 10:45 **Break**
in the Marble Hall and Gallery

10:45 - 11:15 InterpretBank. Redefining Computer-Assisted Interpreting Tools
Claudio Fantinuoli

10:45 - 11:15 Machine Translation & Translator Training: Exploration of Students' Abilities and Needs
Khetam Al Sharou

11:15 - 11:45 From IATE to IATE 2, or When Technologies are Agents of Change and Means to Improve User Satisfaction
Denis Dechandon

11:15 - 11:55 *Workshop*
Interpreters' Workflows and Fees in the Digital Era
Anja Rütten

11:45 - 12:15 Why XLIFF and Why XLIFF 2?
David Filip

11:55 - 12:35 *Workshop*
Lost for Words - Maximizing Terminological Quality and Value at an LSP
David Calvert

12:15 - 12:45 A Crowd-sourced Comparative Evaluation of Phrase-Based SMT and Neural Machine Translation
Joss Moorkens

12:45 - 14:15 **Lunch**
in the Marble Hall and Gallery

13:35 *Poster* Combining Different Tools to Build a Semi-supervised Data Collection Model to Increase MT Quality and Performance
Mark Unitt

Translating and the Computer TC 38

At One Birdcage Walk, Westminster, London (UK)

Day-2: Friday, 18 November 2016
Afternoon Session

Lecture Theatre
Ground level

Afternoon Session Chair:
João Esteves-Ferreira

14:15 - 14:45 From CATs to KATs
Félix do Carmo

14:45 - 16:45 *Panel debate*
Professional Translation in a Pre-Singularity World

Panellists:
Alan Melby, Joanna Drugan, Mikel Forcada, Dieter Rummel, David Wood
Moderator: *Olaf-Michael Stefanov*

16:45 - 17:15 **Break**
in the Marble Hall and Gallery

Education Room
On Lower ground floor

14:00 - 17:00 *Lexical Computing - Gold Sponsor Workshops*

A - Introduction to Sketch Engine for Translators and Terminologists

B - Sketch Engine for Translation and Terminology: Interfacing Corpora with CAT Tools

Miloš Jakubíček and Ondřej Matuška

17:15 - 17:30 **Prizes and Closing** **President, Coordinator and Chairs**

Please highlight these dates in your diary:

will organise:

Translating and the Computer TC 39
16-17 November 2017
London (UK)

For information on next year's **39th Translating and the Computer** conference, **TC39**, please check

http://asling.org/tc39/

for how and when to submit proposals for talks, workshops and posters, and check out other useful information, as these become available.

TC39 will have a special session with a strong focus on **technology tools for interpreters**.

Workshop: Lost for Words—Maximizing Terminological Quality and Value at an LSP

David J. Calvert
TransForm Gesellschaft für Sprachen- und Mediendienste mbH
Dürener Str. 177–179
Köln 50931
Germany
d.calvert@transformcologne.de

Abstract

Part 1: Theory. Although the economics of the business preclude large-scale investment in terminology, I believe that an iterative approach to collecting and improving terminological data can pay off. The quality and value of terminology are discussed from an LSP's viewpoint and defined for an LSP. The features of an optimal terminology process and the process' relationship to the ISO17100 translation process are identified. The interests of the other parties in the translation process are reviewed and best practices for terminology work are identified for the different parties involved. The objectives of a terminology process are formulated and discussed. The features of two standard terminology modules are compared and my choice of terminology server is explained. A standard terminological record structure for termbases is introduced. Part 2: Practice. The second part of the workshop will present an implementation of termbases using this term record structure. This will include the ways in which TransForm is dealing with the strengths and weaknesses of the terminology server used and an iterative process for improving the value of terminological records. Different approaches to automatic term matching will be evaluated, with particular attention paid to the problem of false positive results in QA checks.

1 Theory

Terminology work is often written about and discussed. Yet the terminology work discussed in conference papers and academic textbooks is mostly concerned with single-language terminology and starts from a completely different perspective to that of a language services provider (LSP) or translation services provider (TSP).

1.1 Why do we do it?

I am looking at the subject of terminology from the point of view of a small LSP. My company specializes in various forms of communication, mostly concerned with corporate image as presented to customers, employees or more specific target groups. A large proportion of our work comes from corporate publishers and is destined for publication in print, online or on multiple channels. The range of subjects covered is correspondingly broad, so we have to deal with a wide range of specialized areas, many of which have their own specialized terminology.

Even within specific subject areas, different clients follow different external and internal standards, and may use different regional variants of their corporate language for different parts of the company.

So we need to keep track of terminology—to ensure that we use the appropriate term for the language variant, for the customer, for the subject area, and for any applicable standard. This is a quality-based argument. There are also economically based arguments for terminology work. These include lower costs thanks to a reduction in the amount of research

Proceedings of the 38th Conference Translating and the Computer, pages 1–9,
London, UK, November 17-18, 2016.

necessary prior to or during the translation and review phases, fewer complaints and increased customer loyalty.

Although the economics of the business preclude large-scale investment in terminology, I believe that a well-planned iterative approach to collecting and improving terminological data can pay off for an LSP.

In short, we do it because it saves money and makes our lives easier.

1.2 What are we doing?

"Terminology is the study of terms and their use," writes Wikipedia.[1] That sounds logical, but it doesn't go very far.

TermNet introduces its website with a quotation from Confucius.

ISO TC 37 defines a terminology as "a set of designations belonging to one language for special purposes" and goes on to define such a language as "a language used in a subject field and characterised by the use of a specific linguistic means of expression."

Pavel, in her *Handbook of Terminology*[2], offers two definitions: "The first meaning of the word 'terminology' is 'the set of special words belonging to a science, an art, an author, or a social entity,' for example the terminology of medicine or the terminology of computer specialists." She then goes on to say, "The same term, in a more restrictive sense, means 'the language discipline dedicated to the scientific study of the concepts and terms used in specialized languages.' General language is that used in daily life, while a specialized language is used to facilitate unambiguous communication in a particular area of knowledge, based on a vocabulary and language usage specific to that area."

So it is clear that one term can have two different meanings, i.e. that there are two different approaches to terminology. For the purposes of an LSP, the first definition—a set of words with specific meanings in a specific context—is what we need. The second is the province of professional terminologists, and of practitioners of computational linguistics. As an LSP, we may sometimes rely on the work of such people, but their skills do not form part of our core expertise. It is also important to note that the subject fields referred to in ISO TC 37 span all areas of human activity including commercial activities within vertical industrial or economic sectors[3], so terminology can also be taken to include terms such as department names and job titles, which can be very important to an LSP.

1.3 What are we not doing?

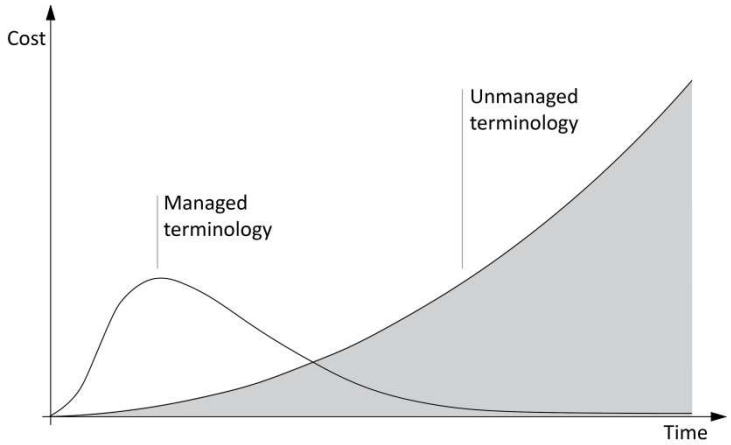
Source language terminology is the customer's job. Any work we do here is a by-product unless the customer is specifically paying us to work on their terminology—in which case they are probably paying the wrong people.

There appears to be a disconnect between expressed belief and real-world practice, at least in Germany, where an online survey in 2013[4] found that 2/3 of the 504 respondents believed that consistent terminology made work substantially easier or easier, a slightly smaller proportion believed it saved time, and well over 80% believed it made similar improvements in quality and customer understanding of technical documentation. The same survey found that over 40% of the respondents stated that terminology was of little or very little importance in their company.

[1] https://en.wikipedia.org/wiki/Terminology
[2] http://itia.ir/farsi/documents/ha.pdf
[3] Warburton, K. after Rondeau, G. Tekom Proceedings tcworld 2013, CHAT 1.
[4] Straub, D and Schmitz, K-D., Tekom Proceedings tcworld 2015, TERM07

1.4 Is it about the money?

The financial return of source-language terminology work can be quantified. Approaches usually attempt to define expenditures and resulting cost savings in order to establish a ROI. These can range from simple "back-of-an-envelope" calculations to more detailed models such as that implemented by ZVEI—the German Electrical and Electronic Manufacturers' Association—in an Excel spreadsheet. The pain curve illustrating the costs of managed vs. unmanaged terminology provides a conceptual basis for this type of calculation.[5] It is important to note that these models are intended for use by manufacturing companies, and that the primary focus is usually on source language terminology. The cost-benefit considerations for translation are usually a combination of subsets of those for technical documentation and marketing communication, and as such may be substantially different to the overall picture.



Terminology pain curve (after Dunne, 2002)

1.5 Costs and benefits

Cost vs. benefit is different for every customer, and frequently for different jobs for the same customer. The amount of effort an LSP can dedicate to terminology work is extremely limited. Most customers are not prepared to pay for terminology work. They simply expect it to be right, although they may not always notice if it isn't. So we do terminology work for the benefits it offers us as an LSP. Our investment is determined by the potential returns, so we have a much greater incentive to invest on behalf of a customer offering a relatively high volume or one providing intermittent but well-paid work. For us, terminology tends to be either rushed for a new customer, or slow and steady for an established one.

1.6 How we work

Terminology for translation must be: identified, researched, and recorded. It may be verified and further documentation may be added. It may then be published or fed back to the originating company. These activities give rise to costs. The first cost, that of the system for storing and managing the terminology, may be covered by the purchase of a computer-aided translation system which includes a terminology management application. Some such systems provide only basic terminological functions, and it may be necessary to purchase a separate program to provide adequate functionality.

Directly attributable costs for terminology arise when a customer delivers terminology associated with a particular project, or when the decision is taken to invest in preparing terminology for a job. Costs for maintaining terminology are also directly attributable. I do

[5] R. Herwartz: „Wann macht sich Terminologie bezahlt? Erstellung einer Kosten-Nutzen-Analyse", in: technische Kommunikation 5/2011, pub. TEKOM

not see scope for a small LSP employing terminologists on anything other than a contract basis related to specific jobs.

The other costs associated with terminology work tend to be difficult to isolate from general overhead.[6] This is also true in an LSP environment, where a significant amount of terminology is identified, researched and (hopefully) documented during the translation and revision processes.

1.7 Types of project and the importance of terminology work

Both the costs and the savings underlying the pain curve follow significantly different patterns for the different types of project dealt with by an LSP. In our case, these types can be classified by volume, frequency, and nature of material.

Recurring regular projects such as employee magazines with a regular publication cycle have characteristic terminological requirements. Such magazines are usually published by corporate communications departments, often with the help of a corporate publisher. Here the target readers are corporate employees, possibly at plants spread throughout the world, and the main purposes of the magazine are promoting a universal corporate culture and a sense of belonging along with conveying essential company information. Articles in such magazines often showcase specific departments, product developments, or management initiatives. Here, vital terminology starts with names—of departments, initiatives or products. Getting the Vice President's department name or job title wrong is as bad as misspelling his or her name. Advance investment in terminology is strongly advisable for this type of project, as is a good system of documentation for terms such as feature names. At the very least, a copy of the previous issue in the target language will clear up the question of whether the section was entitled "In Focus" or "In the Spotlight". Careful TM maintenance also helps in this area. There is often a great deal of overlap between documenting terminology and creating a complete style guide.

In-house magazines for industrial companies also frequently require cooperation between internal and external translation providers. End customers frequently require that individual features, special sections or even whole magazines concentrating on research or innovation be translated in house due to the technical nature of the texts. The corporate publisher will then require translation of headlines, captions and general texts such as editorials followed by at least a copy desk process where conformity with English grammar, spelling, and the house style is checked. Here the problem of maintaining consistency is greater, as there is often no access to a source text, and so no way of applying automated checks for terminological consistency.

Company reports, e. g. quarterly or annual financial reports, or reports covering sustainability and/or corporate social responsibility require the use of specialist terminology defined by specific organizations and subject to change. In particular, financial reporting in Europe usually makes use of the International Financial Reporting Standards and International Accounting Standards defined by the International Accounting Standards Board. These standards are subject to change every year. Similarly, sustainability reports are often subject to the guidelines of the GRI Global Reporting Initiative.

Reports generally involve a significant effort to establish source and equivalent target terminology before the translation of the first issue by an LSP. This effort will usually involve previous issues if available, plus general terminology from standards such as IFRS. At this point, problems such as mismatched regional variants may become apparent, e.g. when a company with US English as its corporate language publishes an annual report on the basis of IFRS/IAS, which are written in British English. The process of terminology collection prior to the first issue is usually similar to but more intensive than that required for magazines.

[6] TSS2009_FS_EconomicIssues, Prof Dr Frieda Steurs, Lessius/KULeuven

Terminology for projects belonging to an account with a more or less regular flow of jobs with common topics, e.g. press releases, technical documentation and websites, follows the classic pain curve, with an initial peak subsiding to a low level of effort required for the addition of new terminology and occasional weeding out of obsolete or deprecated terms.

Terminology for projects belonging to an account with an irregular flow of jobs or covering a wide range of (non-repeating) topics is usually less cost-effective, as the initial peak of the pain curve repeats with every new topic. The decision to invest in terminology is on the basis of risks/rewards for the individual job plus a speculative component dependent on the likelihood of the customer sending more regular work.

It is seldom worth carrying out substantial terminology work for apparently one-off jobs with no reasonable expectation of follow-up work, e.g. contracts or static websites. Here, the time covered by the pain curve is the duration of the single project, so the peak of cost due to terminology management may be greater than the costs incurred by not managing the terminology. The decision to invest in terminology is on the basis of risks/rewards for the individual job. One significant exception to this is where the LSP has been brought in to work on a pitch. Providing a limited amount of terminology work as part of a pitch is clearly a gamble, but does demonstrate the team's commitment to quality. This is also a good way to increase customer loyalty.

Terminology linked to a specific account is not available for general use, as it will contain company-specific material and material subject to copyright and confidentiality.

Terminology that is not linked to a specific account is available for general use but is strongly constrained by subject area.

1.8 Risks associated with terminology

The risks associated with incorrect use of terminology are a subset of those associated with incorrect translation. The consequences range from causing amusement among colleagues to bearing responsibility for death or injury due to incorrect operating instructions or service documentation. By drawing up matrices of likelihood of specific consequences occurring vs. the consequences themselves for specific types of terminology error we can determine the level of risk posed by incorrect translation of terminology. Possible immediate consequences can be graded in order of increasing severity, from e.g. Internal communication impaired, no material consequences, to Danger to life or limb. This approach makes clear that while getting the job title of an executive or the name of a department wrong will lead to embarrassment and may lead to a loss of trust, the overall risk is less than that incurred when a product name or description is wrong, as there is a significant risk of expensive corrections at a late stage in prepress work, or worse if the presses have already started to roll.

1.9 Cooperation

The key factor in enabling effective cooperation is making it easy by removing barriers. Translators will not provide services free of charge if they do not see an immediate and direct benefit from doing so. The same applies to convincing in-house staff to willingly identify and record terminology.

1.10 Relationship to ISO 17100

The translation workflow as specified by ISO 17100 Annex A only mentions terminology once, under Section 4, Pre-production processes and activities. It is specified as an optional step in point 4.6.3.2, which states that "…the client and the TSP can agree that the TSP shall ensure that the appropriate terminology is available…". Point 5.3.1 a) of Section 5, Production process, specifies compliance with domain and client terminology and maintenance of terminological consistency. A significant part of the challenge for LSPs is to

obtain and validate the terminology in the first place, and this is an area where the tool vendors are a long way from supplying optimal solutions. Although the ability to capture terminology on the fly has been around for some years, there is no simple way of returning such captured terminology as part of a job package.

1.11 Interested parties

The interests of the client are best served by delivering a translation which does not expressly contradict the end client's existing documentation and material, unless such contradiction has expressly been requested, as part of a product relaunch, for example.

Subcontractors usually want to deliver a product which conforms to the customer's expectations at the lowest possible cost to themselves.

For suppliers, the best practices in the translation process can basically be summed up as consistency, documentation and communication. Consistency, because it makes problems easy to fix; documentation, because it makes it possible to recognize and avoid the problem the next time around; and communication, because it ensures that people are aware of both the problem and its solution. The most constructive practice from the LSP side is to facilitate and encourage feedback of terminological problems and of the proposed solutions from suppliers. Naturally, this requires LSPs to form close, long-term relationships with selected suppliers.

For clients, the picture is more varied. From the LSP's viewpoint, the most important best practice is the use of professionally prepared source language terminology in source language documents. The second most important one is to have their target language terminology reviewed by someone who is both familiar with the domain and a native speaker of the target language. Generally, however, the LSP's role here is limited to asking what, if anything, exists and is available.

It is also important for LSPs to distinguish between different types of client. Agency and publisher customers rarely have the expertise or the need to receive terminology in any form other than a glossary supplied as a PDF. In-house translation departments, on the other hand, are more interested in receiving terminology in a form compatible with their system. End customers will often have their own specific input format specific to their implementation of a terminology database.

1.12 Subject-specific challenges

Different customer accounts present different challenges. Linguistic problems are always present. For example, translating German financial reports into English involves problems such as the German word *Rechnungsabgrenzungsposten*, which translates as *prepaid expenses* when it appears on the assets side of the balance sheet and as *deferred income* when it appears on the liabilities side. Or the German word *Umsatz*, which is variously translated as *sales*, *revenue*, *revenues* or *turnover* for different German companies. If the original accounts have been worked out according to the German HGB standard, then many of the terms used will be conceptually different from English accounting terminology and the text will require a degree of localization. Researching specific subject areas can be problematic; for example, "older" areas of industry such as railway technology are not as well documented online as IT and telecommunications. Technical issues such as the nature and format of available terminology also arise and call for different approaches.

1.13 Starting points

The most common starting point for terminology work for a new account is probably one or more PDF documents. These may be exports from an in-house system, or (possibly protected) PDFs of last year's annual report. Excel spreadsheets are also popular among users. Possible

challenges here include problems caused by the fact that Excel's default text delimiter varies according to the regional settings of the version of Windows under which it is running. For example, the straight double quote used by Excel in English is also the symbol for inches and can cause problems in Excel glossaries.

End customers' terminology is usually in a form suitable for the customer's own use, i.e. arranged as a dictionary or glossary. It usually has not been lemmatized or edited for automatic term recognition.

1.14 What is quality?

The idea of "fit for purpose" is a fundamental tenet of quality assurance. There is no point in wasting effort on producing something that exceeds the required specification. The primary purpose of terminology work at an LSP is satisfying the customer. So it follows that the main considerations on the quality side are:

- Customer acceptance
- Consistency
- Correctness

Correctness here is taken to mean that the term is intelligible to the rest of the world—the Humpty Dumpty problem—and that it does not contradict other established uses. Trade-offs between customer acceptance and correctness are almost always decided in favour of customer acceptance—at least initially.

However, from an LSP's point of view, we also want to maximize returns and minimize costs. These objectives are achieved by optimizing the content of our termbases and the automatic term recognition settings to maximize the hit rate of terms recognized, while minimizing the rate of incorrect recognitions and false positives generated during automatic quality control. From the LSP's point of view, the cost-effectiveness of a termbase is its main quality criterion.

1.15 The ideal and the real world

The ideal customer has a well-defined collection of source-language terminology put together by a professional terminologist and coupled with target-language terms approved by in-country reviewers with the relevant expertise. And this terminology is available in TBX or some other form of XML.

In practice, one or all of these features will be missing. Even where the target language terminology exists, it may well have been prepared by interns or students, hopefully working under the direction of a terminologist. It may have been obtained from the development department, and be heavily influenced by the source language, or from an overseas subsidiary, and have little relationship with the source language concepts. Or it may have been crowdsourced.

The LSP's task here is to convert any existing terminology into a cost-effective termbase with the minimum of effort.

1.16 So where do we want to go?

We want to abolish duplication of effort.

We want to be able to benefit from our efforts by reusing their results.

We know that the journey never ends.

1.17 And how do we intend to get there?

We have to establish a working system and ensure that it minimizes effort and maximises returns. The situation represented by the terminology pain curve is, however, an idealized representation of the cost of terminology for an end customer. It does not take into account

the effects of such events as new product launches or version releases, let alone corporate reorganizations or changing documentation standards. There is also little point in an LSP implementing monthly updates to a termbase if the termbase is only used for translating a customer magazine twice a year. This is even more so when the updates need significant effort to port them into the TLS's system. So the real picture of terminological cost is characterized by occasional peaks either immediately prior to the translation of an issue or immediately after, when feedback has been received after the customer's review of the translation.

One effective mechanism for improving existing terminology is by iterating through feedback loops. In addition to documenting new terms, reviewers can note problems with existing terminology. The logs from any quality control tool used provide valuable indications of which terms are causing false recognition results and how the results from the termbase can be improved.

1.18 The journey to date

TransForm's first termbase system was MultiTerm in its file-based incarnation as a part of Trados Translator's Workbench for Windows. As the technology vendors moved from file-based to server-based systems, software costs for operations of our size increased dramatically. File-based systems were effectively removed from the market and the capabilities of non-server systems were restricted to single users on networks without domain controllers. Server-based systems for around five users were relatively expensive, so we had to find an alternative strategy. This was achieved by making increased use of Wordfast, which was already in use as our backup system and was widely used by our freelancers. We also used an intranet-based system for collecting terminology on a project basis.

Wordfast uses glossaries for terminology. For Wordfast Classic and Wordfast Server, these are simple tag-delimited text files with the first three fields defined as Source Language Term, Target Language Term, and Comment, and three further, user-definable fields which can be used for attributes. The glossaries for Wordfast Pro 3 and 4 can also import TBX, although only a subset of TBX can be accommodated by a glossary structure. Wordfast also enables the use of Blacklists of forbidden terms. Wordfast distinguishes between automatic fuzzy terminology recognition, which does not require editing the source terms in the glossary, and manual fuzzy terminology recognition, which makes use of asterisks in the source terms as wildcards.[7] The asterisks can be placed at the beginning or the end of the term, or in the middle of the term. The trade-off between automatic and manual terminology recognition is less accuracy vs. more initial effort required.

memoQ Translator Pro and memoQ Server use a concept-oriented termbase structure. However, the termbase definition is fixed, and the user is limited to Kilgray's choice of fields. It has the classic TBX-style three-level structure with concept, language, and term levels. It also contains Kilgray-specific fields and, as previously mentioned, omits some fields which are extremely useful to LSP users. However, Kilgray also supplies a terminology server, known as QTerm. This runs on the Web server integrated into memoQ Server and supports user-defined fields within the three-layer structure. Some of the Kilgray-specific fields from the standard terminology module are included in the termbases to maintain compatibility. Forbidden terms can also be stored in memoQ. However, unlike in Wordfast, they are stored within the termbases, and are distinguished on the term level by a "Forbidden" attribute. This apparently has certain implications for the fragment assembly and predictive typing features of memoQ.[8]

[7] https://www.wordfast.net/wiki/Fuzzy_Terminology_Recognition

[8] Thread "Feature request (or does this already happen?): no forbidden TB entries in predictive typing," memoQ@yahoogroups.com, March 2016

After trying out memoQ as a TM system I came to the conclusion that it offered a high degree of interoperability and was the best choice for our current operation in terms of capabilities and cost-effectiveness. However, the limitations of the built-in termbase made it necessary to purchase the QTerm terminology server extension for the standard memoQ Server.

This history has led us to define a standard terminological record that offers our ideal balance between the effort put into collecting terminological data and the scope for its current and future utilization.

1.19 The TransForm standard terminological record

The structure of the standard terminological record used at TransForm was originally defined for terminology collection via a form in the translation management database in the company intranet. It was derived from TBX-Basic and automatically associated metadata from the translation management database with source-target term pairs, thus building up account-specific glossaries that could be imported into other systems via TBX-Basic. Our move to QTerm termbases required modifications to the structure to accommodate memoQ-specific fields necessary to maintain compatibility.

The Concept (termEntry) level contains the standard transactional information on creation date and user and last modified date and user. It also contains the memoQ built-in fields for Domain, Subject, Client, and Project (metadata), and Image and Image caption fields. The memoQ termbase field Note and an ID field are also present.

The Language (langSet) level contains the memoQ built-in field Definition. This is directly equivalent to the descrip tag in TBX-Basic.

The term (tig) level contains the term itself and the fields Term source, Usage example, Usage example source, Note, Term type, Validation status and Validated by. It also contains three built-in QTerm fields: Case Sensitivity, Matching, and Forbidden. These are necessary to maintain compatibility with memoQ, in particular with the QA Check feature. The memoQ termbase fields Part of Speech, Number (grammar) and Gender (grammar) have also been included to retain compatibility with memoQ.

The key elements of this structure are the three-level concept-based structure itself and the use of specific fields. In particular, the compatibility with the TM system ensured by the memoQ built-in fields benefits term recording and recognition. However, one of the key factors behind the choice of QTerm instead of simply using the standard memoQ termbase was the need to define a term source field. This is because the source of a term is an extremely useful proxy for the term's reliability. If a term is used by the customer in the customer's documentation there is little scope for disagreement about the use of the term. Similarly, the documentation of both a usage example and the source of that example provides a known degree of confidence in the reliability of the term in context.

The Validation and Validated by fields have been brought over from the TransForm intranet terminology record, where they were intended for use as elements in an EN 15038-compatible terminology process.

1.20 To be continued…

We are currently have 20 years' worth of terminology collected and partly duplicated across four different systems. We are in the process of establishing which parts of this data are worth porting to QTerm and of unifying and porting the data selected, and of optimizing those parts of the data that have already been ported.

The second, practical part of the workshop will look at how some of these ideas and approaches are being implemented at TransForm GmbH.

Potential impact of QT21

Eleanor Cornelius
Department of Linguistics
University of Johannesburg,
South Africa
International Federation of
Translators (FIT)
eleanorc@uj.ac.za

Abstract

This paper describes the QT21 project from the perspective of the International Federation of Translators (FIT) in three main parts. Firstly, six of the ways that humans currently relate with machine translation (MT) systems will be outlined, leading up to a seventh way that will be discussed in more detail. Huge volumes of texts need to be translated in different sectors of the economy globally. A feasible approach to meeting this need is to employ both raw MT and humans, including translators, in addressing the world's translation needs. Secondly, analytic evaluation of MT quality by human translators will be introduced, focusing on the MQM framework. This seventh way involves annotation, by humans, of specific errors in the raw MT using standardized error categories, rather than only generating a single number indicating overall quality. Lastly, the potential impact of QT21 on MT and professional translators will be reflected on. Through FIT, human translators will be able to participate in the development of improved MT systems. This will help them give objective advice to clients and to guide the developers of next generation translation tools. FIT's position is there will be enough work for translators who do not feel threatened by MT.

1 Introduction

The primary aim of this presentation is to determine the potential impact of the QT21 project on human translators. In order to do so, I firstly provide some background information on the International Federation of Translators (FIT), and on the QT21 project, including FIT's involvement in this project. In the sections that follow, various ways in which humans interact and engage with machine translation are listed and described. A discussion of the QT21 project will not be complete without focusing on two important aspects of this project, namely research and evaluation. Lastly, I consider the impact of the QT21 project on human translators in the years to come.

2 Introduction to FIT and description of the QT21 project

FIT is an international federation of associations of translators, interpreters and terminologists. Through affiliation, more than 80 000 translators in 55 countries across the globe are represented in FIT. In short, FIT's goal is to promote professionalism in the disciplines it represents (http://www.fit-ift.org/).

FIT is a partner in the three-year Quality Translation 21 project (abbreviated as QT21 project) which runs from February 2015 to February 2018. QT21 is a machine translation project which forms part of the EU Horizon 2020 Framework. This project is managed by the German Research Center for Artificial Intelligence (in English abbreviated as DFKI). The main purpose of the QT21 project is to address language barriers in Europe that impede free flow of information. This purpose is in line with the EU's objective, to be achieved by 2020, for a European Digital Single Market that can operate without any barriers, linguistic or otherwise One goal of QT21 is to improve MT models and outcomes for language that (1) are

Proceedings of the 38th Conference Translating and the Computer, pages 10–18,
London, UK, November 17-18, 2016.

morphologically complex, (2) have free and diverse word order, and (3) are under-resourced. (See http://www.qt21.eu/.)

The explosive growth in data witnessed today has not seen an equal growth in the level of translation. MT can go a long way to address this imbalance between supply and demand, notably in cases where its quality is sufficient for the purpose at hand without taking away from the current work of translators. This relates to work not currently done by human translators.

Through investigation and analysis of innovative methods of machine translation, QT21 and FIT will engage translators in assessing the quality of machine translation, incorporating human judgement into the current data-driven development paradigm. Analytic metrics developed in the context of QT21 have already seen the harmonization of MQM and DQF into a single framework, an early deliverable of the project, to define benchmarks for translation quality. Indeed the project proposal points to the need for "metrics [that apply to both] human and machine translation." (See http://www.fit-ift.org/introduction-to-qt21/.) Through contracts between FIT and DFKI, FIT is also instrumental in the dissemination of the findings and advances of the QT21 project.

But how do human translators currently engage with MT?

3 Human engagement in machine translation

In this section, I list and discuss six of the various ways that humans, including translators and non-translators, currently relate with machine translation (MT) systems. At the end of section 4, I describe a seventh way.

3.1 Provision of input

Human translators provide input to MT, in the form of training material for data-driven systems. Pre-processing involves making source texts and their translations into bitexts and includes normalisation of those bitexts. A bitext, according to Harris (cited in Melby, Lommel & Vásquez, 2014: 409), "is a source text and its corresponding target text as they exist *in the mind of a translator*. […] Together, the translation units of the bitext constitute the entire source and target texts 'laminated' to each other."

Specifically, normal pre-processing includes the following: [1]

- *Sentence tokenization (segmentation):* This entails putting each sentence/segment on its own line.
- *Sentence alignment*: Ensuring that source and target sentences are on the same corresponding line numbers, and possibly using empty lines when there is a many-to-one or one-to-many sentence relationship.

At this point, a bitext has been created.

- Depending on the MT system, *removal of formatting annotations*, like italics, bold, hyperlinks, etc.
- *Character normalization*, so that orthographic variations are systematic. Examples include: replacing non-breaking spaces with normal spaces; opening and closing double quotes (",") with neutral ones ("); same for single quotes; replacing guillemets/angle quotes (« or »), German-style Anführungszeichen, and east Asian quotation marks with consistent ones (when a language uses multiple forms); normalizing combining characters with precomposed characters; some languages use multiple orthographic

[1] I am hugely indebted to Jon Dehdari from DFKI who provided the information contained in this section.

variants, like the use/non-use of the zero-width non-joiner in Persian, use of Eszett (ß) in Germany and Austria vs.double-s in Switzerland and Liechtenstein;

- *Tokenization*: separating punctuation away from words, so that the following sentence:

I'll take 3.5 of those, please.

becomes

I 'll take 3.5 of those , please .

This is a step that seems easy, but is annoyingly difficult to get right, especially across languages. There are currently more tokenization algorithms than there are pubs in Ireland!

- *Lowercasing or truecasing*: Truecasing is making a word lowercase if it normally is. For example, a truecaser would change the first word in an English sentence to lowercase for a word like "The", but not for a word like "Japan". Some words are tricky, like "University" or "Apple".

Data-driven MT heavily relies on human translation. In order words, the human translates the source text that is used as training material in MT. It thus follows that without human translation, there would be no data-driven MT.

3.2 Pre-editing of sources texts

Whereas pre-processing, as discussed above, involves creating bitexts from already translated source texts, pre-editing (PE) involves preparing source texts, that are not part of the training material, for MT. According to Martinez (2003: 16) the aim of pre-editing is "to achieve better human readability and clarity of the SL text, as well as better computational processing or translatability, especially by translation systems" (original emphasis). The distinction between "readability" and "translatability" is discussed further below.

The concept 'Controlled Language' (CL) during authoring is relevant to pre-editing as, according to O'Brien (2010: 143), there is overwhelming evidence that application of CL rules (that is, "constraints on lexicon, grammar and style with the objective of improving text translatability, comprehensibility, readability and usability") has a marked positive effect on MT quality.

Reuther (2003: 124-5) explains that CL can have two uses: (1) to enhance readability and understanding (i.e. cognition) by focusing on text linguistic aspects, and (2) to improve translatability by MT systems. In either case, the processing system may be human (a human reader, or a human translator) or it can be automatic (a monolingual automated language processing system, or an automated translation system, such as TMs or MT systems). Reuther (2003: 125) provides examples of constructions on lexical level, formatting level, and phrase and sentence level, that may pose problems for MT systems and, in some cases even humans as well, to process, regardless of use (to enhance readability and understanding, or to improve translatability). On lexical level, spelling, morphological and synonym variants may create processing problems. On formatting level, issues relating to punctuation, spacing and typography may pose problems for MT systems, but not for human processing. On phrase and sentence level, Reuther (2003: 126) indicates that some syntactical constructions affect readability and comprehension, but do not pose translation problems (regardless of whether the translation is done by a human or an MT system). In other cases, both comprehension and translatability may suffer. Readability CL rules and translatability CL rules are not vastly different, as readability rules are subsumed under translatability rules. This means translatability, according to Reuther (2003), facilitates readability, as there are only a few translatability rules that are not also readability rules (that is, at least, in the case of German).

Even with the use of TM and the resulting quality of the translated output, it is important to feed the TM with controlled output from the very beginning. If this is not done from the outset, a mismatch will occur between controlled input and uncontrolled reference material stored in the TM (Reuther 2003: 131). As (1) texts are written by humans, and (2) CL on all levels (lexical, formatting, and phrase and sentence level, and possibly also global text level) ensures both readability and translatability, the role of the human in the pre-editing process should not be under-estimated.

3.3 Gisting and triage

Humans also perform gisting and triage; that is, they assign a general meaning to raw MT output (gisting) and decide whether further processing is needed (triage). As early as 1979, Henisz-Dostert (1979: 153) cited Garvin (in Lehmann and Stachowitz 1971: 114) who said that MT output, for various purposes, "will be only casually scanned rather than carefully read". Although this source (Henisz-Dostert, 1979) is particularly old, it is interesting to note that not much has changed since that time (as far as gisting is concerned, at least).

This idea of scanning a text, translated by an MT system, is now also known as content gisting or browsing (see Martinez, 2003: 18). Gisting is a monolingual human activity in which the source text has no place or importance (this means the source text is not consulted during the gisting process). The purpose of gisting is to arrive at a general idea of what is conveyed in a translated text, i.e. the raw MT output. The following response of a respondent in Henisz-Dostert's study (1979), to the question "Why do you use MT?" sums up the purpose of gisting particularly well: "To determine if the publication contains material that is pertinent to my work." In such cases, the "speed of access could compensate for inadequacies of machine translation" (Henisz-Dostert, 1979: 155), as MT is faster than human translation (ibid, 166; 184). The person who does the gisting does not have to be a translator, nor does s/he have to be proficient in the source language. Gisting can be done for a number of personal reasons – and indeed Henisz-Dostert (ibid, 180) predicts that "under the conditions of a regular, rigorous service, requests for translations for scanning purposes only would become routine", and/or it can be a step leading to triage.

Triage is used by people who are not translators to determine whether human translation of a machine-translated text is warranted. Triage is thus not a form of translation, but much rather a decision-making process aimed at determining the best way to proceed in order to reach a particular goal. In relation to optimal use of resources, Melby (2016) makes the following statement: "[…] documents that are useful as raw machine translation or are never consulted do not use up valuable human resources for further post-editing or translation." Of course, in instances of incomprehensibility of a raw machine-translated text, the need for post-editing or improvement arises. This could entail requesting retranslation by a human translator, for instance.

Against this backdrop it is important for professional translators, having specialised knowledge and specific expertise, to advise their clients on whether MT is the best option or whether another approach would be better suited to fulfil the particular translation need.

3.4 Post-editing (PE) of MT output

Another way in which humans are involved in MT relates to PE of raw MT output: the correcting of mistakes in the raw machine-translated text.

According to Martinez (2003: 18), inbound translation to understand (assimilation) is not accompanied by PE (in the case of content gisting) or it is supplemented by rapid PE (RPE) (or light post-editing) in order to correct only the most serious errors in order to improve comprehensibility and accuracy. Texts edited in this way usually have a short life-span.

However, outbound translations to communicate (dissemination) require either (1) minimal post-editing (MPE) – in cases of technical texts, such as a set of instructions, with a longer life-span, in cases where cohesion is not all important, or (2) full post-editing (FPE), in cases where high-quality translation is required, or (3) only 10-20% of a document will be edited in cases where 80-90% accuracy is achieved, for instance the fully automated translation of weather reports.

PE, according to Martinez (2003: 20-2), is done by either by translators, revisers, non-linguists (technical experts) or trained specialists in the company, but the type of PE required will also be a determining factor. Martinez (2003: 22) explains: "(T)his new role where efficiency is a priority, could be successfully fulfilled by 'anyone' with ((very) good) bilingual and linguistic skills, involved in the field of communication of information". However, Martinez (2003: 21) also warns as follows when translators are used as post-editors:

> PE is completely different from translating and requires a different attitude to text production as well as certain "ideal" abilities. Sometimes, when MT software offers low quality, translators can become resentful of the fact that they could have produced a better translation from scratch. In most cases, translators find machine-translated texts irritating and rarely enjoy correcting bad translations.

Important in all of this, is the primary instruction to the human. Is the human instructed to translate or to edit? If the human is instructed to translate, the activity is human translation. If the human is instructed to edit machine-translated text, the activity is PE. Thus, PE is not human translation. The amount of time and cost expended, during PE, to achieve a product of high quality should be carefully considered. If excessive effort is required, then human translation – from scratch – should be advised.

3.5 Use of selected segments of raw MT

Human translation, typically, begins with a source text, accompanied by a set of instructions that can either be implicit or explicit. The end result of this activity is a target text. Humans can optimally use various resources while translating. If the instructions are appropriate and if the translated product meets these instructions, the product will be of high quality.

The professional translator consults various resources during the translation process. This typically includes terminology and translation-memory lookup. The translator, however, is free to either use or ignore suggested (real or fuzzy) matches. Likewise, when segments of MT are available, the translator is free to use them or ignore them.

3.6 No use of MT

Humans can also bypass MT; that is, they can translate from the source text using either no translation-specific tools at all or only terminology lookup and/or translation memory lookup, without consulting raw MT output.

3.7 Translators, bilinguals, and monolinguals

Some of the six translation-related human activities described above require the skills of a professional translator, some only require knowledge of both the source and target languages, while others can be performed by monolinguals. With the huge volume of texts that need to be translated in different sectors of the economy in the world today, the only feasible approach meeting this need is to employ both raw MT and humans, including translators, in addressing the world's translation needs. To this end, collaboration between professional translators and the buyers of translation is all important. FIT does not view MT as a threat to professional translators.

4 An introduction to two aspects of the QT21 project

4.1 Basic research

There are ten well-established universities that are partners in the QT21 project (see http://www.qt21.eu/consortium/). Some of the new approaches to MT that will be tried out are RNNs (Recurrent Neural Networks), novel syntactic and semantic translation models, and APE (automatic post-editing). FIT will not be involved in the basic-research aspect of QT21. These new approaches are mentioned only because they all require evaluation of the raw MT output to determine whether they are better than current approaches.

4.2 Analytic evaluation of MT quality by professional human translators

This section introduces the topic of analytic evaluation of MT quality by human translators, as a goal of the QT21 project is "improved evaluation and continuous learning from mistakes, guided by a systematic analysis of quality barriers, informed by human translators" (http://www.qt21.eu/) (own emphasis). This, then, represents the seventh way in which humans engage with MT. It involves annotation, by humans, of specific errors in the raw MT using standardized error categories, rather than only generating a single number indicating overall quality.

The focus here is on the MQM framework as a complement to, not a replacement for, reference-based translation evaluation methods, such as BLEU, that is widely used (Lommel, 2016: 63).

The most common approach to evaluation of an MT system under development is to select a source text and have it translated by a professional human translator. The raw MT output is then automatically compared with the human translation (the reference translation). Changes are made to the MT system, and the same source text is translated again and automatically compared with the reference translation in order to determine whether the change in the system made the output look closer or further away from the reference translation.

What then are the characteristics of translation quality metrics? A system can either be:

- holistic (focusing on the entire text) or analytic (focusing on specific portions of the text)
- reference-based (it requires a reference/sample/model of translation, previously done) or reference-free (no previously translated text is required)
- automatic, and thus fast, or manual, and thus slower.

Additionally, metrics can differ in terms of their validity. In relation to validity, the following question arises: "Does it measure what it is supposed to measure?" Lommel is critical of the validity of reference-based methods. A reference-based metric (such as BLEU) works on the underlying assumption that a particular reference translation is –

> […] a valid measure of quality and the tests designed to demonstrate that validity bias the results because they use a similar method with human evaluators who cannot independently evaluate the translations without the references that are under consideration (Lommel, 2016: 64).

Although BLEU is designed to cope with more than one reference translation, BLEU scores are typically measured by using only a single reference (Lommel, 2016: 64). Moreover, claims that BLEU matches human judgment may also be flawed, as it is not clear what these judgments are about. It is also debatable, according to Lommel (2016: 64), whether referenced-based methods indeed measure translation quality.

Metrics can also differ in terms of degree of *reliability*, begging the question: "Will the metric perform consistently when used by different evaluators during an actual application?" In terms of reliability, BLEU is reliable. Lommel (2016: 64) states the following:

> Because it is mechanical, for a given set of references and a hypothesis BLEU will always generate the exact same score. When the hypothesis changes the score will perfectly reflect the differences. BLEU does not depend on the judgment of an annotator.

Despite the reliability of BLEU as described in the quote above, MT engines are inherently inconsistent (Lommel, 2016: 65), as they may do very well with one part of text, but perform less well in another part.

In the QT21 project an additional method of evaluating the output of an MT system is used. Professional human translators apply a quality metric developed within the MQM (Multidimensional Quality Metrics) framework. The MQM metric will not be automatic but will be analytic; that is, specific errors in the raw MT are annotated by humans using standardized error categories, rather than only generating a single number (such as a BLEU score) indicating overall quality. Thus, the MQM-based metric in QT21 is manual (not automatic), analytic, and highly informative. Additionally, it does not require a reference translation as, in a typical production environment, there is no reference translation available. Why would there then be a need for another translation? Thus, automatic evaluation only makes sense in a MT development context, where a reference is used an evaluation tool.

Based on Lommel's (2016) assessment, it follows that reference-based methods will not always indicate whether the modified translation is better or worse than a previous translation, and for this very reason, it is therefore not as useful as it may seem.

In light of the above, improvements in quality must be meaningful in human terms. It is therefore important to incorporate judgements of human translators in translation quality evaluation. Both types of metrics (automatic and analytic) have a role to play in the assessment of translation quality. However, the strong and weak points of each system should be carefully weighed up. Whereas automatic metrics are fast and good for research and development, analytic metrics provides insight into specific problems and they can discriminate based on differing specifications (or instructions). The single score an automatic system allocates is not meaningful in human terms as it provides little insight into the problems in the translated product and the types of improvement required to enhance quality. Then again, analytic metrics (such as MQM), are slow and more expensive than automatic approaches, and they cannot be used for rapid development. Therefore, both BLEU-style and MQM-style metrics are needed.

MQM is a flexible system for defining metrics (either analytic or holistic), that allows for various specifications. Each general set of specifications will have its own metric (which may be identical to the metric for another set of specifications in some cases). MQM can be used to assess conformance to specifications for each type of translation:

- Raw MT: Does the translation output meet requirements for end-user usage?
- Triage is a downstream use, but we need to know if the translation is good enough for that use.
- PE: Is the translation fluent and accurate enough to support efficient PE? Does the human contribution bring the translation in line with its specifications?
- MT as an option and "classic" human translation: We can evaluate the text for its intended final use.

In light of the above, it is important to note that there can be no single set of specifications that applies to *all* translation. Quality depends on purpose, needs, and scenario. It is possible

to have a variety of measures of quality; however, not all measures will be appropriate for any given translation project. The metrics that are applied to assess translation quality should be in line with the particular specifications (instructions) relating to the translation project. Different metrics give *different* quality scores for the *same* text depending on the specifications, and thus: what is a good translation for one purpose may not be good for another.

For example: Consider a source text that is written in a very high and difficult register, but the text is being translated for use in educating twelve-year-old students. A metric that values absolute fidelity to the source will give a translation that meets specifications a bad score. A different metric that does not penalize changes in register will give a more appropriate score. Thus, changing what is measured produces a new metric.

MQM defines a family of metrics, as no single metric can ever apply to all translation projects.

Why is MQM good for professional translators? This metric provides a way to specify how translators will be judged that *respects* their ability to produce appropriate translations and their right to refuse inappropriate work. The metric is fair, as the criteria that are used for evaluation of quality is made available in advance. Moreover, MQM allows for direct comparison of different methods of translation and reproducible methods of assessing whether a translation meets the mutually agreed upon translation specifications. Lastly, MQM helps translators to understand the strengths and weaknesses of MT.

5. Potential impact of QT21 on MAT and on professional translators

FIT will invite human translators to participate in the QT21 project, from its substantial pool of translators that it represents through member associations. This will provide an opportunity for those translators to gain an insider view of the world of MT and thus better understand its current status. FIT is of the opinion, as stated in its Position Paper on MT (http://www.fit-ift.org/wp-content/uploads/2016/09/MT_pospaper_exit2.pdf), that "(T)ranslators should seek to respond to the new developments in good time and see how to derive benefits for themselves." Through their involvement and active participation in the QT21 project, translators will be able to see the strengths and weaknesses of MT, because reports of FIT's experience with evaluation will be disseminated to the entire FIT community. All of this, in turn, will help human translators give objective advice to those who need translation services and guide those who develop the next generation of translation tools. MT developers will look for ways to improve MT based on the annotations of human translators.

The position of FIT is that there will be more than enough well-paid work in the foreseeable future for translators who do not feel threatened by MT and who can advise others on a team that can use all seven ways of interacting with MT. In an evolving translation market, the volumes of translation work are increasing. This means the pie becomes bigger and bigger, and so the slices of the pie also grow proportionally in size. FIT's position, as expounded in its Position Paper on Machine Translation (http://www.fit-ift.org/wp-content/uploads/2016/09/MT_pospaper_exit2.pdf) is that there will be instances where raw MT output is completely acceptable. In such instances the user of a text simply wants to extract the gist of a text in its basic form (see the discussion of gisting in part 3.3 above). In other instances, there may be highly adverse consequences to raw MT output, for instance when businesses make available unedited MT texts to accompany their products. Such unedited machine-translated texts could damage the corporate image of the company and there could even be product liability implications.

In balancing the huge advances that are made in the field of MT, there can be little doubt that it is in the best interests of the translator community to actively engage with the entire translation industry on MT, in general, and the evaluation of translation quality, in particular.

Translators should become familiar with FIT's involvement in MQM and should acknowledge that both BLEU-style as well as analytic metrics have a role to play in quality evaluation. Those working in the field of MT people are most probably very familiar with BLEU, but may be less knowledgeable about MQM. Through its involvement in the QT21 project and the development of MQM, FIT plays an active role the translation industry.

Acknowledgements

The project QT21 leading to the above results has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645452.

References

Acrolinx. http://www.acrolinx.com/ Accessed: September 25, 2016.

Dehdari, Jon. DFKI. Electronic communication on 22 September 2016.

International Federation of Translators (FIT). 2016. Welcome to FIT. http://www.fit-ift.org/ Accessed: September 16, 2016.

International Federation of Translators (FIT). 2016. Introduction to QT21. http://www.fit-ift.org/introduction-to-qt21/ Accessed: September 14, 2016.

International Federation of Translators (FIT). 2016. Position Paper on Machine Translation. Available at: http://www.fit-ift.org/wp-content/uploads/2016/09/MT_pospaper_exit2.pdf Accessed: September 22, 2016.

Lommel, Arle. 2016. Blues for BLEU: Reconsidering the Validity of Reference-Based MT Evaluation. In Rehm, Georg & Burchardt, Aljoscha (eds). *Proceedings of the LREC 2016 Workshop "Translation Evaluation – From Fragmented Tools and Data Sets to an Integrated Ecosystem*", pp. 63-70.

Martinez, Lorena G. 2003. *Human Translation versus Machine Translation and Full Post-Editing of Raw Machine Translation Output*. Unpublished master's dissertation. Dublin City University.

Melby, Alan. 2016. Interaction between human translation and machine translation (MT) in translation activities. Available at: https://www.fbcinc.com/e/LEARN/e/Translation/presentations/Wednesday/MT-HT-Spectrum-OneSheet_V4b.pdf Accessed September 30, 2016.

Melby, Alan K., Lommel, Arle & Vásquez, Lucía M. 2014. Bitext. In: Chan, Sin-wai. *The Routledge Encyclopedia of Translation Technology*. Chapter 25, pp. 409-424.

O'Brien, Sharon. 2010. Controlled Language and Readability. In: Shreve, Gregory, M. & Angelone, Erik (eds). *Translation and Cognition*. American Translators Association Scholarly Monograph Series XV. John Benjamins Publishing Company, pp. 143-165.

QT21: Quality Translation 21. Available at: http://www.qt21.eu/. Accessed: September 25, 2016.

Reuther, Ursula. 2003. Two in One – Can It Work? Readability and Translatability by means of Controlled Language. In *Proceedings of the Joint Conference combining the 8th International Workshop of the European Association for Machine Translation and the 4th Controlled Language Applications Workshop (CLAW 2003)*, 15th-17th May, Dublin City University, Ireland, pp. 124-132.

From IATE to IATE 2
or when technologies are agents of change and means to improve users' satisfaction

Denis Dechandon
Head of the
Language and Technology Support Section
IATE Tool Manager
Translation Centre for the Bodies of the European Union
Luxembourg
Denis.Dechandon@cdt.europa.eu

Abstract

The migration to the revamped, modernised and upgraded InterActive Terminology for Europe, the EU's inter-institutional terminology database, is going through a thorough IT development process designed to produce a brand new tool built around some major requirements as detailed in this paper.

Keeping in mind all improvements required, as defined by a dedicated task force reporting to the IATE Management Group ('IMG'), or needed, due to the obsolescence of some technologies used over the last 12 years or to the availability of new technologies that could better serve users' needs. Taking into account the current state of a tool which had undergone corrective and evolutive maintenance over time with an increasing number of technical limitations, it was proposed and accepted to go for a brand new tool. The rebirth of IATE was announced.

Over the last two years the interinstitutional cooperation took a new rise: all Task Forces of the IMG brought ideas and expressed needs, while engaging in complementary activities, such as a vast cleaning of IATE entries.

The IATE 2 project deliverable will provide enhanced and new features to further improve terminology management. In conjunction with its collaborative platform, EurTerm, it will also strengthen collaborative working with stakeholders.

Keywords: Making life easier for users, responsive web design, integration with Computer Assisted Translation and Terminology (CATT) tools, improved collaborative working, improved return on investment.

1 Looking in the rear-view mirror

In the early 2000s, IATE was a very exciting and challenging project in the field of terminology. Originally born as a project of the Translation Centre for the Bodies of the European Union for its clients[1], it gained a much larger dimension, bringing together Eurodicautom, Euterpe and TIS, the former terminology databases of, respectively, the **European Commission**, the **European Parliament**, and the **Council of the EU**, together with further databases and terminology assets of the current partners of the IATE project, i.e.:

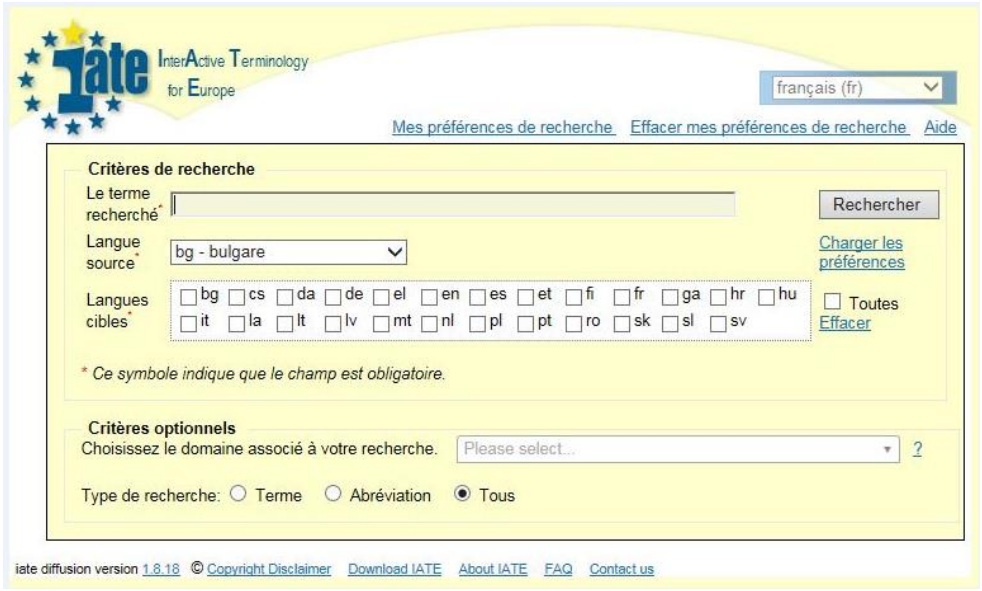
- Court of Justice
- Court of Auditors
- European Economic & Social Committee/Committee of the Regions

[1] Johnson, I. & Macphail, A. (2000). IATE – Inter-Agency Terminology Exchange: Development of a Single Central Terminology Database for the Institutions and Agencies of the European Union (http://mt-archive.info/LREC-2000-Johnson.pdf)

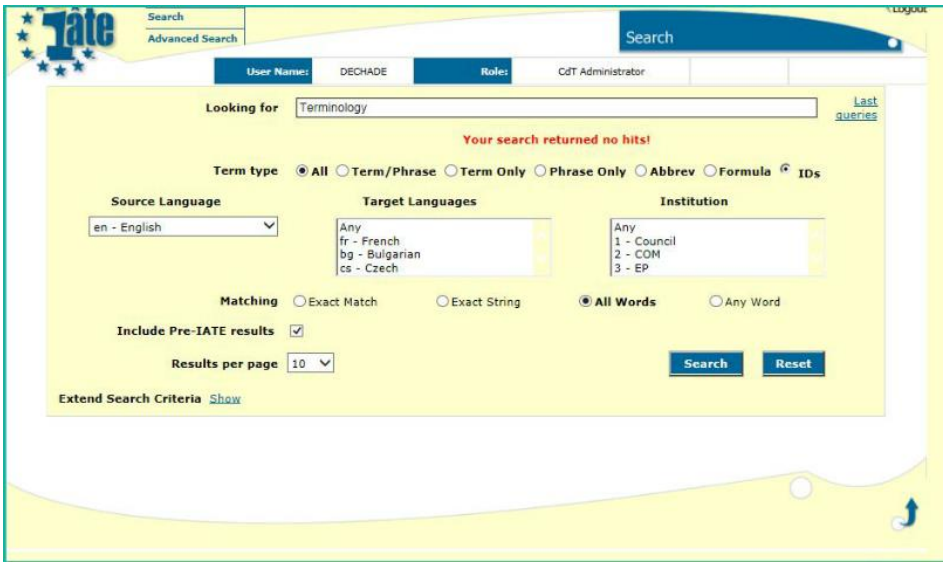
Proceedings of the 38th Conference Translating and the Computer, pages 19–32,
London, UK, November 17-18, 2016.

- European Central Bank
- European Investment Bank
- Translation Centre for the Bodies of the EU

The result of this collaborative working is well known, the biggest terminology database in the world. The public (read-only) interface as shown below was released in June 2007 and achieved over 41 million hits in 2015.



Prior to this, the internal interface as shown below was launched in the summer of 2004. It offers advanced features and different user rights for all EU linguists and drafters. Highly appreciated, IATE Internal had more than 17 million hits in 2015.



Where the public version of the tool provides read only access to external users, the version used by internal linguists enables read-and-write access and includes additional features (*inter*

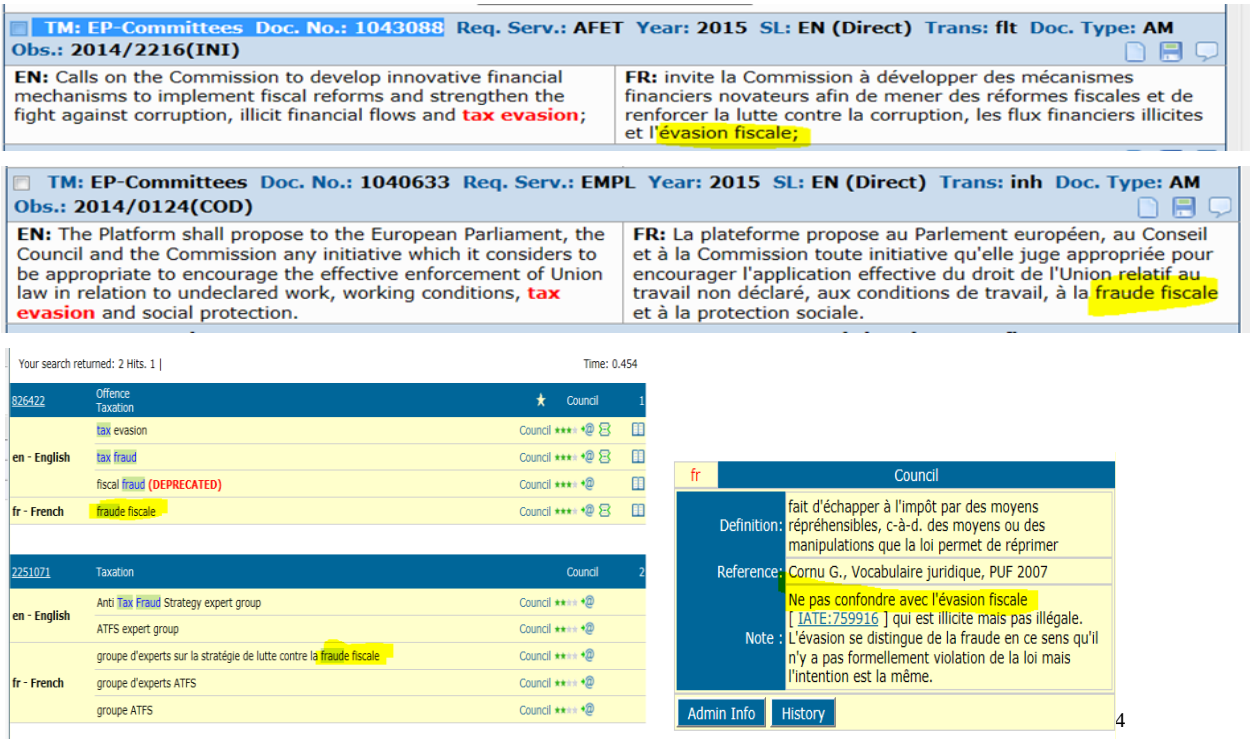
alia creation, consultation, editing, validation workflow[2], user management, content import and export), making IATE Internal a powerful and collaborative ('interactive') terminology management tool.

Hosted by the European Commission in Luxembourg, IATE is supported and developed by the Translation Centre for the Bodies of the European Union and managed by the IATE Management Group ('IMG'), a working party composed of one member per institution or body[3].

2 Technological evolutions in the translation world

IATE has undergone ad-hoc evolutive maintenance to satisfy emerging needs over time. In parallel, technologies implemented in the background went somehow old, if not obsolete for some of them or at least challenged by newer ones. Furthermore, translators', interpreters' and terminologists' needs had been evolving a lot since the early 2000s and some technologies contributed to the development and implementation of complementary and somehow competing tools. For example, the implementation of Computer Assisted Translation and Terminology (CATT) tools and Machine Translation (MT) engines, used in association with smaller tools helping translators to (meta)search terms and quotes, led to the redefinition of the processing of documents to be translated. The translation world was entering a new era.

Meanwhile it appeared that translators' reflexes were increasingly influenced by these new assisting technologies and tools, as they were helping them to win the race against shorter deadlines. Nevertheless as all linguistic assets used by such tools are not always of equal and constant quality, their concomitant use creates a perverse effect as some assets might be of different reliability or relevance.



4

[2] Johnson, I., Palos-Caravina, M.-J. (2000), 'Validation and Quality Control Issues in a new Web-Based, Interactive Terminology Database for the Institutions and Agencies of the European Union' in *Translating and the computer 22*, Aslib, London

[3] http://iate.europa.eu/faq/IATE_FAQ_EN.htm

[4] http://www.termcoord.eu/wp-content/uploads/2016/05/Vienna-LSP-2015.pdf

Alongside the implementation of improved and new (CATT) tools, a further step was taken with the introduction of a new mandatory stage into the translation workflow, i.e. the human / semi-automated / automated pre-processing of source documents, with i.a. referencing, pre-formatting, pre-translation with translation memories and MT.

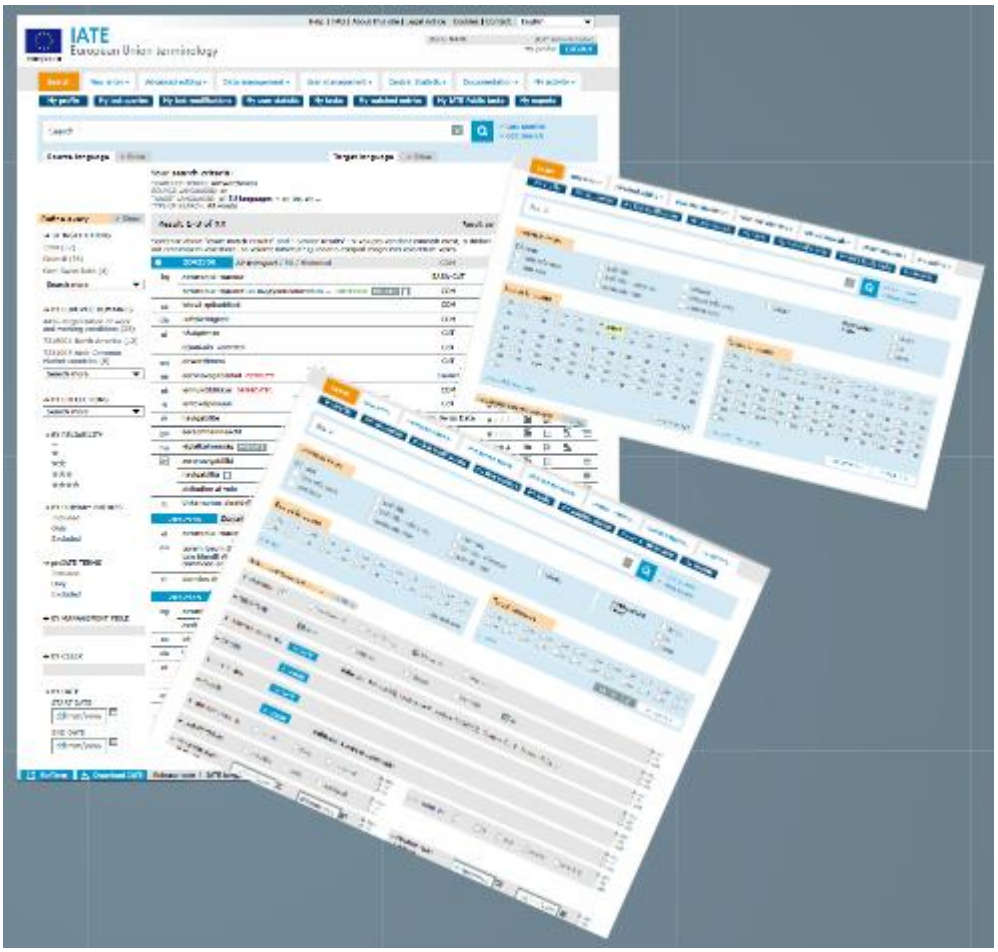
The next step will be the integration of the automatic term recognition module that can be used with various formats and by many user groups. Part of IATE, this module will be the very first one of IATE 2 and will contribute to its increased return on investment.

3 New technologies to satisfy IATE users' needs and to enable the development of new features and enhanced functionalities

New needs emerged over time and numerous requests for improvements were expressed and satisfied. As a consequence, IATE had already been developed and improved on several occasions.

Ultimately, all task forces of the IATE Management Group brought ideas and expressed further needs. Among them, the Data Entry Task Force, renamed IATE 2 Task Force, had the mandate to define the requirements for IATE 2 and adopted a common and harmonised position and approach. They came to the conclusion that, further to technical enhancements, major evolutions were needed:

- New improved layout and look & feel



- Full-text search and predictive text
- Easier data editing

- Advanced management of references (EUR-Lex)
- Early duplicate detection
- Communication tools (forum, tasks)
- Consolidation and project management tools
- My IATE panel (with user tasks, user statistics, i.a.)

Taking into account the further improvements required as defined above, and the additional ones needed due to the obsolescence of some aging technologies, it was proposed and subsequently agreed to build and create a new tool, IATE 2.

Following this, three main aspects had to be considered: the technologies to keep or to replace, the IT architecture and the database itself, i.e. the linguistic content.

In terms of technologies, several aspects had to be taken on-board, e.g.:

- Implementation of a responsive design (because of the use of laptops, tablets and smart phones)
- Interconnectivity
- More user-friendly interface
- Need for an improvement of the stemming feature

In view of the main needs expressed, the IATE Support & Development Team came to the conclusion that some technologies had to be replaced. For example, despite strong efforts in relation with one of the technologies used so far, no improvement could be achieved and a switch was decided, i.e. to adopt Elasticsearch and Lucene:

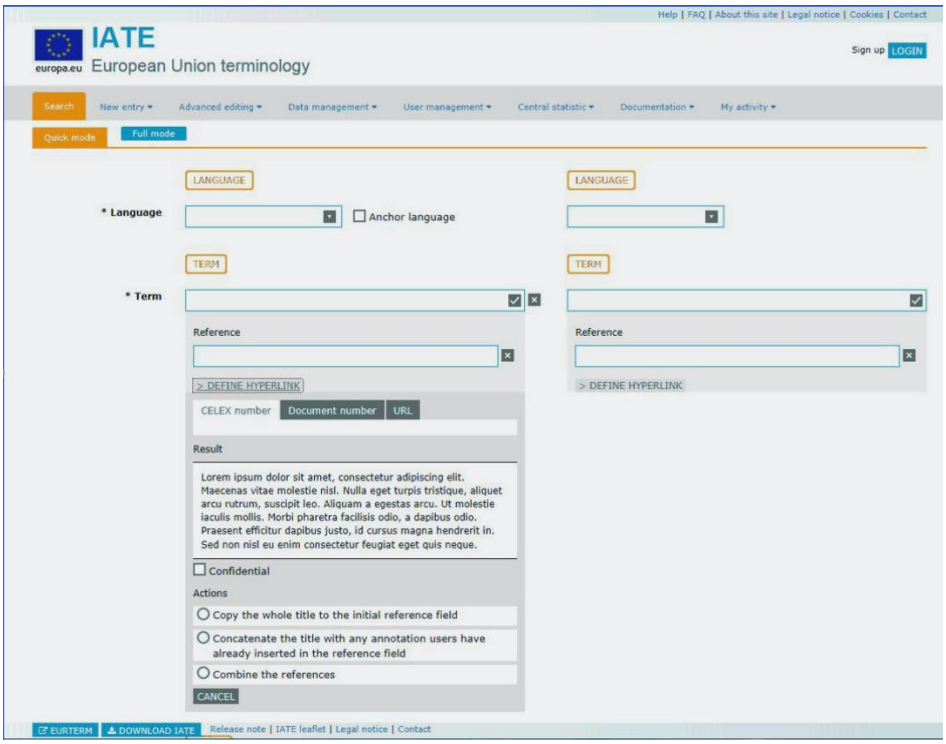
- The adaptability and transparency of these technologies, as well as the quality of the results achieved convinced the IATE Support & Development Team that these technologies are best suited to solve the issues detected by users and will allow for the improvement of the search function in IATE 2.
- A copy of the current Oracle database will be kept as back-up for contingency.

Consequently, the IT architecture and the infrastructure had to be redefined. Building on these results, the Support & Development IATE team and its Coordinator, hosted by the Translation Centre for the Bodies of the European Union in Luxembourg, have already laid the first cornerstones of this large IT development project:

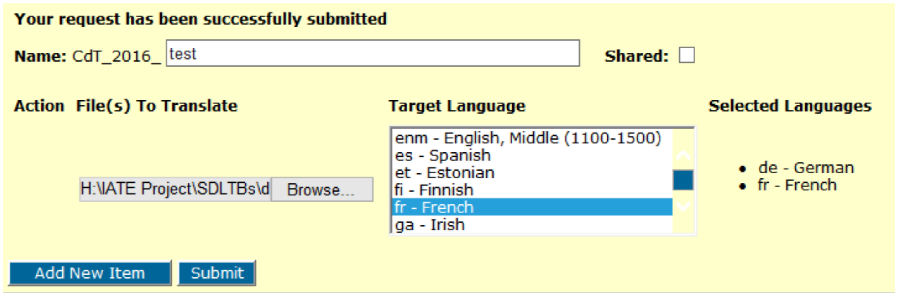
- Identifying which IATE parts can be re-used in IATE 2
- Setting-up development, test and user acceptance testing environments
- Selecting new technologies
- Adopting the Agile development practices
- Drafting prioritised user stories involving all stakeholders of the project

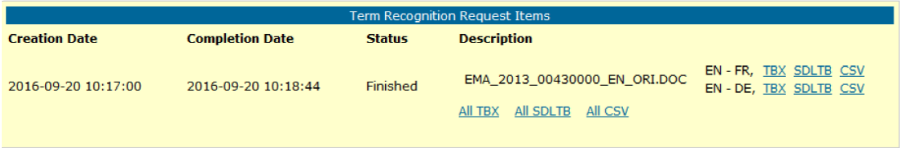
Whilst enhancement needs are defined, the foundations are now in place and the developments are on-going. Linguists can also expect over the next 18 months:

- For the current users of IATE Internal:
    - Early duplicate detection
    - Easier editing (creation/update modules)



- Easier management of references (EUR-Lex)
- Enhanced collaboration and interoperability (APIs, web services)
- Enhanced management of terminology projects within the terminology database
- Quick reporting of problematic content or entries for merging/delete/review/complete in other languages
- On-line statistics on searches and search results
- Watch entries mechanisms
- Term recognition module integrated to Institutions' CATT tool





- Improved feedback mechanism
- Communication tools (forum, tasks)
- Contextual help
- My IATE dashboard (with user tasks, user stats, etc.)

- For the current users of IATE Internal and IATE Public:
    - Full revamping of the user interfaces
    - Advanced search with combined filters
    - Full text search (search in identified database fields)
    - Enhanced indexation of data
    - Responsive design to respond to different target audiences and devices
    - Accessibility

The third aspect, the linguistic content, is in the hands of all linguists, i.e. translators, language terminologists and central terminologists, with some input from external users and partners.

4 IATE entries

At the same time to the activities described above, it appeared as well necessary to work on the quality of the linguistic content of the tool to better meet users' needs. This is where the interinstitutional cooperation further advanced.

If we first look back through history again, all EU institutions and other project partners already had their own terminology databases when the idea of creating a single central terminology database took root. With objectives, benefits and main features already defined at an early stage, the biggest challenge was the merging of the existing terminology resources and, as a consequence, the consolidation of legacy[5] terminology in this new tool.

Duplicates and incomplete entries were the 2 main issues identified at that time. Tools were used to detect problematic content but some 16 years after the initial launching of the project, it appears that only a part of the legacy data has been revised or updated.

Considering the growing availability of linguistic assets through various media, the obsolescence of some collections and the impact of the significant increase of the EU official languages, efforts have been (and are still) on the consolidation of duplicates, consolidation of the legacy data in all policy areas which are of relevance for translators at specific moments in time[6].

A few years after the most recent enlargements of the European Union (2004, 2007 and 2013), the challenges in the field of terminology remain significant within a multilingual framework.

[5] Rummel, D. & Ball, S. The IATE Project - Towards a Single Terminology Database for the European Union (http://www.mt-archive.info/Aslib-2001-Rummel.pdf)

[6] Zorrilla-Agut, P. (2014). When IATE met LISE: LISE clean-up and consolidation tools take on the IATE challenge. In G. Budin & V. Lušicky (eds.), *Languages for Special Purposes in a Multilingual, Transcultural World, Proceedings of the 19th European Symposium on Languages for Special Purposes, 8-10 July 2013, Vienna, Austria.* Vienna: University of Vienna, 536-545.

What are these challenges?

- Fill terminology gap between pre-2004 and post-2004 languages
  - Pre-2004: need for consolidation to reduce the "noise" in IATE
  - Post-2004: need for term creation to reduce the "silence" from IATE
  - "Noise" combined with "silence" means reduced efficiency
- Increase IATE quality
  - Increase number of IATE entries having definition, context and /or note ("added value")
  - Added value: IATE quality indicator

Or to present things differently:

Language	IATE entries (language level) 31.03.2015	Added value
FR	1 064 803	26,8%
EN	1 055 123	27,4%
DE	783 524	27,0%
IT	567 911	20,1%
NL	547 892	23,2%
ES	501 656	15,5%
DA	493 025	17,6%
EL	429 304	17,8%
PT	426 418	15,9%
FI	272 643	23,1%
SV	271 296	23,6%
GA	55 489	3,0%
PL	54 796	52,6%
LA	54 199	6,8%
LT	50 253	40,6%
ET	40 456	48,0%
SL	39 103	53,8%
MT	36 123	60,1%
CS	35 651	47,0%
SK	34 443	57,8%
RO	33 211	56,8%
BG	32 553	62,6%
HU	29 743	66,3%
LV	28 737	59,0%
HR	9 835	79,5%

Language	IATE entries (language level) 30.06.2016	Added value
FR	1 039 828	27,34%
EN	1 038 399	27,76%
DE	761 677	27,75%
IT	545 657	20,86%
NL	539 718	23,64%
DA	486 126	18,05%
ES	486 003	16,08%
EL	427 469	18,15%
PT	419 491	16,71%
FI	272 063	23,61%
SV	266 172	24,71%
PL	58 028	55,28%
GA	57 125	4,64%
LA	53 752	6,82%
LT	52 935	44,99%
ET	43 041	50,81%
SL	42 314	56,22%
MT	39 831	64,35%
SK	37 613	60,92%
CS	37 601	51,22%
RO	37 405	62,19%
BG	34 752	64,72%
HU	32 519	67,86%
LV	31 695	60,32%
HR	13 026	79,63%

Various features and methods have been defined and implemented to maintain and clean-up the database. This involves various stakeholders in all project partners, e.g. central terminology services, language terminologists, as well the IATE Support & Development Team. Each of them has a defined role and the combination of their activities ensures the sustainability of linguistic maintenance tasks and the steady enhancement of IATE entries.

All these background activities can be tracked, as the IATE history feature works reasonably well and keeps track of changes to individual fields. This will be retained in IATE 2, considering that this covers internal users' needs, as it makes it possible for them to check changes (who did what and when).

5 Improvement of the return on investment and linguistic content enhancement

The introduction of an automated pre-processing step for documents to be translated, similar to what is done with the translation memories technology, and the growing use of machine translation imply that efforts must be dedicated to the quality of the linguistic content re-used through such automation.

If we go back to the past and analyse previous comments, communications, activities and projects in the field of terminology quality management, it appears that it has been a recurring topic and concern[7].

As a matter of fact, various strategies and punctual actions were developed and launched over the last 12 years, starting with the introduction and implementation of a validation workflow in IATE and the identification and merging or deletion of duplicates[8].

As a basic principle, translators, language terminologists and central terminologists respectively have the following specific roles:

- feed the database on the fly,
- check and, where appropriate, validate the data inserted by translators, focus on added value data (e.g. definitions and contexts) and liaise with other terminologists of their language communities,
- coordinate work for all EU official languages, having a multilingual overview of the entries, and request work from language terminologists.

Even if they all have always been involved in the quality enhancement of the IATE database, the scope of the need has never been totally defined as such. Nevertheless, the necessity to improve translation quality thanks to the terminology quality enhancement has been explained and targeted in various publications and through some visible activities[9].

Further to this, the interinstitutional cooperation in the field of terminology is a powerful driver of the IATE linguistic content enhancement. Central terminology services launch and coordinate multilingual terminology projects and ensure that entries are properly documented in IATE for consistency and precision purposes.

To ensure quality, terminology working is governed by a framework, is carried out according to work programmes and builds on consolidation work. Additionally, feedback received from users is of utmost importance. To broaden the scope of the consolidation work, strategies are defined to identify poor data.

In this context, the IATE Management Group and the central terminologists in most IATE partners play an important role in the quality efforts. IATE falls indeed under the shared responsibility of all EU institutions and bodies involved in the project.

The IMG deals mostly with technical and data management issues. It also handles other issues of interest for interinstitutional cooperation, like training, legal aspects of terminology work and terminology-related tools. The IMG established task forces and working groups to

[7] Ball, S. (2003). Joined-up Terminology – The IATE system enters production. Translating and the Computer 25, November 2003. [London: ASLIB, 2003]

[8] Zorrilla-Agut, P. (2013). When IATE met LISE. LSP Symposium - Vienna

Lušicky V. & Wissik T. (2012). Terminology: Don't only collect it, use it! Translating and the Computer Conference, London, 29-30 November 2012

[9] Zorrilla-Agut, P. (2013). When IATE met LISE. LSP Symposium - Vienna

Lušicky V. & Wissik T. (2012). Terminology: Don't only collect it, use it! Translating and the Computer Conference, London, 29-30 November 2012v

consistently improve IATE, i.e. the interface and its features (through developments done by the IATE Support & Development Team), and the content of the database through various activities and channels:

- **IATE Handbook Task Force**: created to draft the IATE Handbook, a manual for terminologists working in IATE.
- **Data Entry Task Force**: initially created to look for ways to simplify the insertion of new data in IATE, it ended up with the task of drafting the specifications for a future version of IATE (IATE 2.0).
- **EurTerm Task Force**: ad hoc group discussing the management of the EurTerm collaborative platform.
- **Data Clean-Up Task Force**: created to look into ways of cleaning up the low-quality data in the IATE database to facilitate its integration with CATT tools and MT engines.
- **IATE/Studio Integration Task Force**: created to test and monitor the integration of IATE data into the Term Recognition module.
- **Normative Terminology Task Force**: created to look into better ways to identify and create sets of reliable data that can be made available through IATE for different purposes. Among them we find the creation of "authorities' tables" (used by the Publications Office) and the integration of IATE data into the CATT tool used by translators.
- **Task force for Interinstitutional Cooperation in the field of Terminology**: created by the Coordination Committee on Translation (CCT) to look into ways to increase cooperation in the field of terminology. It was not specifically speaking an IMG group, but most of its members were also members of the IMG.
- **Group of interinstitutional coordinators**: ad hoc group composed of the central terminology coordinators of the most active institutions in the field of terminology. They concentrate on avoiding duplication of efforts and on sharing information about projects underway in each institution. They also have the responsibility for submitting proposals to the IMG concerning data management or technical issues.
- **Interinstitutional Taxonomy Group**: created to set writing rules for updating Latin and MUL terms (abbreviations) on biological species (fish, animals, plants, micro-organisms, etc., and they also update these rules whenever necessary. This Group also initiates interinstitutional projects to further consolidate the IATE database with the ultimate aim of having a single entry for each taxon.
- **Interinstitutional Toponymy Group**: informal group created to deal with toponymic entries in IATE.

In the history of IATE, EU terminology services have continuously carried out recurring maintenance tasks, such as consolidation projects, targeted actions according to identified areas of interest, batch clean-up and updates of set of data spotted according to various criteria (e.g. low reliability, missing fields), duplicates, processing of feedback received from internal

and external user). Furthermore, they sometimes commit to broader clean-up projects (e.g. LISE[10]).

The updating/cleaning of IATE entries remains an on-going activity. The overall amount of entries is around 1.4 million multilingual entries, which corresponds to some 8.7 million terms in more than 24 languages.

As mentioned previously, the Data Clean-up Task Force committed to the vast cleaning of IATE entries, whilst the requirements for IATE 2 were being defined. Since its creation at the end of 2014, the Data Clean-Up Task Force has been working intensively.

Several activities were launched by this task force, but even more options to explore were and are proposed, such as the:

- clean-up of terms with brackets and other weird characters,
- building of lists of multilingual duplicates (where the term is duplicated in several languages),
- extraction of entries from other institutions than the Commission with references pointing to Eurodicautom,
- identification and deletion of low quality collections,
- deletion of:
    - entries with "hyperterms" (i.e. where the main term contained as synonym its hyperterm (e.g. "crusher block" + "block")),
    - duplicated sets of entries (i.e. similar sets of entries (glossaries) that had been entered in IATE several times by different institutions).

In a theoretical perspective, various ISO standards of interest and established strategies to clean-up and maintain databases provide a valuable reference framework. However, it appears that IATE is quite unique considering the volume of its contents, with some collections dating back to the 1970s, and the number of languages covered. This particular clean-up effort is mostly driven by the need to eliminate noise from hitlists. This is of particular relevance when it comes to the implementation and systematic use of the Term Recognition Module as part of the automated linguistic pre-processing of files to be translated with a CATT tool.

In this context the Data Clean-up Task Force has identified several ways of selecting data of poor quality and has been deleting and merging content for more than a year now with the participation of all EU institutions. Tens of thousands of low-quality entries have already been deleted or merged and this is only a beginning in this framework. Its members recently focused their work on the clean-up of monolingual entries. Notably, they cause noise in the hitlists obtained by terminologists but this is not the case when it comes to the Term Recognition Module. The focus is now on bilingual entries in all EU pre-2004 official languages. To date, bilingual entries amount some 20% of all entries in IATE.

The clean-up of legacy data has been a recurring topic since the creation of the database[11], and it seems that it will remain as such, as the volume of the whole content proves to be an obstacle to the completion of this activity.

[10] Lušicky V. & Wissik T. (2012). Terminology: Don't only collect it, use it! Translating and the Computer Conference, London, 29-30 November 2012

[11] Ball, S. (2003). Joined-up Terminology – The IATE system enters production. Translating and the Computer 25, November 2003. [London: ASLIB, 2003]

Further to the possibilities mentioned and to the criteria defined above, some other strategies could and possibly should be investigated and defined, such as detecting:

- which entries never appears in hitlists results,
- internal inconsistencies (e.g. long terms containing terms already available in simpler entries but with different equivalents).

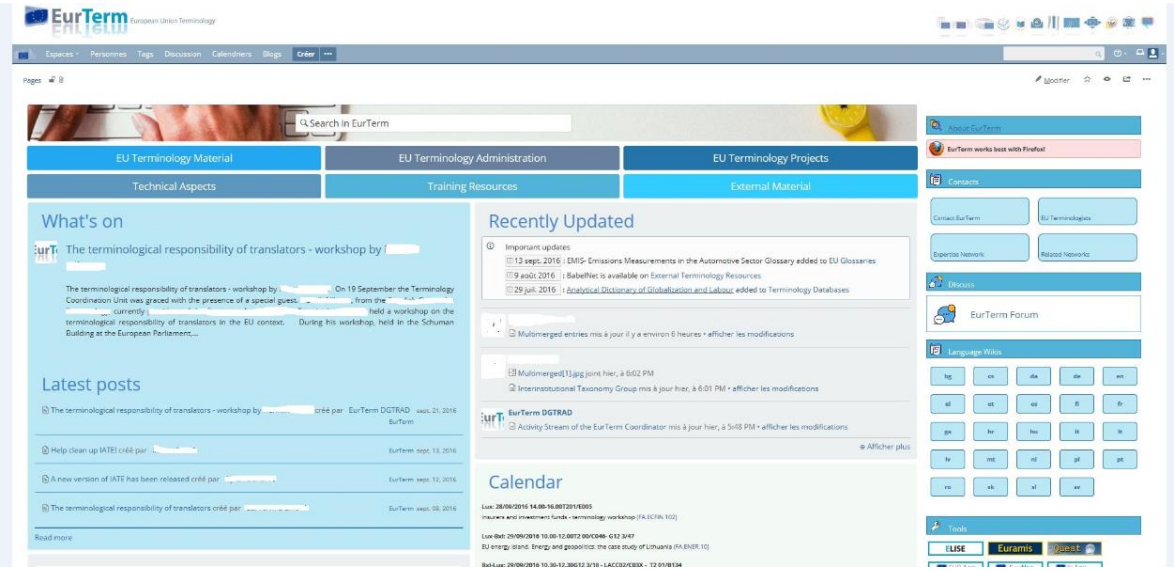
Nevertheless, considering respectively that:

- central terminologists concentrate on defined topics and areas that are most relevant for translators at specific times,
- the sheer volume of terms available for some languages (e.g. more than 1 million terms each for the English, French and German languages) and the number of staff that should be available to perform such an activity

Therefore priority is currently given to the strategies mentioned above.

6 Collaborative working

Beyond IATE and all the previous mentioned collaborative activities related to its use, feeding and maintenance with numerous stakeholders, another tool must be mentioned, EurTerm. This is the collaborative platform made available to internal translators and terminologists.

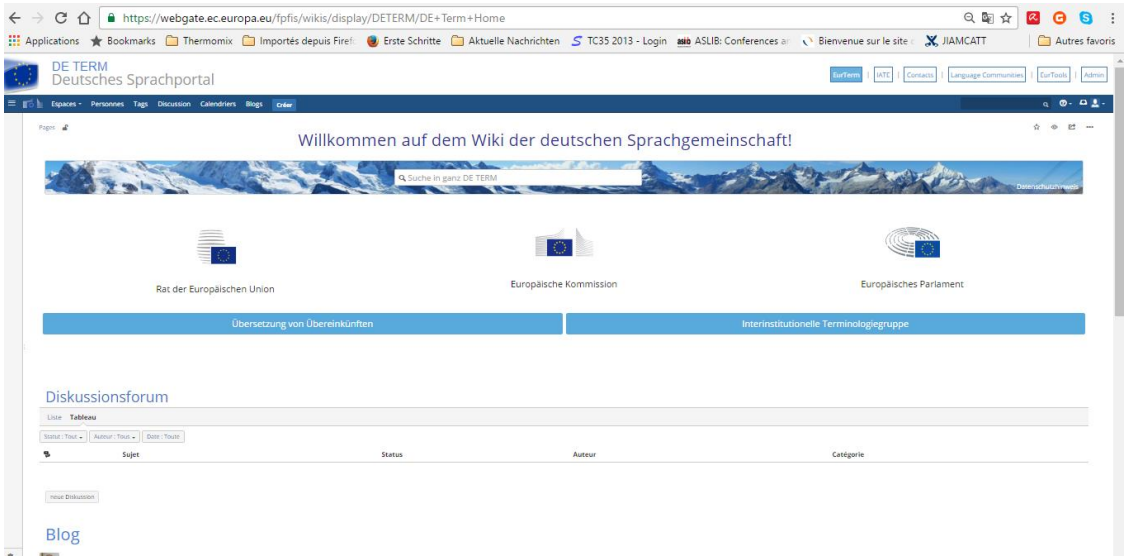


EurTerm is dedicated to the sharing of information, resources and of knowledge in the field of terminology between its users. It allows for synergies and collaborations, e.g. on terminology projects. Incidentally, it offers some features of interest for all internal linguists:

- A collection of institutional glossaries and external resources,
- Contact lists of EU terminologists and other experts,
- A calendar for all interinstitutional and international events concerning terminology.

Last but not least, it contains language wikis which can be used by all language communities in a very dynamic manner. Meant to facilitate communication and cooperation in terminology between institutions and with their national linguistic bodies, they contain collaborative

spaces for terminology working for each language community and provide a platform for communication, information and knowledge exchange.



Beyond internal collaborations, possibilities are increased by contacts and cooperation activities with:

- International Organisations for the sharing of terminology resources and to support some terminology projects,
- national authorities and experts for the access to specialised terminologies,
- universities, which make it possible for the academic community and the EU linguists' community to mutually benefit from each other (sharing of several terminology projects and recruitment of numerous trainees every year).

7 Conclusion

Perceived as a success of the interinstitutional cooperation, the IATE project fosters collaborations at various levels and enters a new era thanks to new technologies allowing for enhanced features which will better meet its users' needs.

From a practical, linguistic and terminology perspective, much of the work done in the background by highly motivated internal translators, language terminologists and central terminologists will contribute to the success of IATE 2, whose development is on-going.

IATE 2 in association with EurTerm, its collaborative platform, will continue to build on the achievements of IATE (Internal and Public). This next evolutive phase will sustain the contribution of the IATE project as a whole to the collaboration between linguists for the sake of translation quality. Beyond this, IATE 2 is anticipated to make it possible to take terminology management to a next step.

Acknowledgements

I would like to thank the IATE Management Group (and particularly its former chair, Mr. Dieter Rummel) and the IATE Support & Development Team and its current and former Coordinators for their patience and for all the knowledge passed on.

References

Ball, Sylvia. (2003). Joined-up Terminology – The IATE system enters production. Translating and the Computer 25, November 2003. [London: ASLIB, 2003]

Fontenelle, Thierry & Maslias, Rodolfo & Swinnen, Ingrid & Zacharis, Konstantinos IATE and interinstitutional cooperation in the field of terminology [Brussels, 2015]

Ghelf, Hether. (2013). Best Practice Strategies to Clean up and Maintain Your Database (http://fr.slideshare.net/BlackbaudPacific/best-practice-strategies-to-clean-up-and-maintain-your-database-hether-ghelf)

IATE Handbook (https://iate.cdt.europa.eu/iatenew/handbook.pdf)

ISO 12620:2009 – Terminology and other language and content resources -- Specification of data categories and management of a Data Category Registry for language resources

ISO 26162:2012(en) – Systems to manage terminology, knowledge and content — Design, implementation and maintenance of terminology management systems

ISO/TC 159 – Ergonomics

Johnson, Ian & Macphail, Alastair (2000). IATE – Inter-Agency Terminology Exchange: Development of a Single Central Terminology Database for the Institutions and Agencies of the European Union (http://mt-archive.info/LREC-2000-Johnson.pdf)

Johnson, Ian, Palos-Caravina, Maria-José (2000), 'Validation and Quality Control Issues in a new Web-Based, Interactive Terminology Database for the Institutions and Agencies of the European Union' in *Translating and the computer 22,* Aslib, London

Lušicky V. & Wissik T. (2012). Terminology: Don't only collect it, use it! Translating and the Computer Conference, London, 29-30 November 2012

Rahm, Erhard & Hai Do, Hong. Data Cleaning: Problems and Current Approaches

Rummel, Dieter & Ball, Sylvia. The IATE Project - Towards a Single Terminology Database for the European Union (http://www.mt-archive.info/Aslib-2001-Rummel.pdf)

Zorrilla-Agut, Paula. (2013). When IATE met LISE. LSP Symposium - Vienna

Zorrilla-Agut, P. (2014). When IATE met LISE: LISE clean-up and consolidation tools take on the IATE challenge. In G. Budin & V. Lušicky (eds.), *Languages for Special Purposes in a Multilingual, Transcultural World, Proceedings of the 19th European Symposium on Languages for Special Purposes, 8-10 July 2013, Vienna, Austria.* Vienna: University of Vienna, 536-545.

Translation Quality Evaluation of MWE from French into English using an SMT system

Emmanuelle Esperança-Rodier
Université Grenoble Alpes – LIG/GETALP
Emmanuelle.Esperanca-Rodier@univ-grenoble-alpes.fr

Johan Didier
Université Grenoble
Johan.Didier@univ-grenoble-alpes.fr

Abstract

Nowadays, Statistical Machine Translation (SMT) is widely available. Nevertheless, using Machine Translation at its best is not an easy task. Structures appearing sporadically trigger most of the regular mistakes of SMT systems. We work on one of those structures: the MultiWord Expressions (MWE). Our study aims at evaluating the quality of MWE translation obtained using SMT.

Firstly, we present the process of our quality evaluation of the English translation got via an SMT system created using Moses Toolkit (Koehn et al., 2007), of one French technical document. On the French document, MWE have been semi-automatically annotated according to their type (Tutin et al., 2015). Secondly, we describe the linguistic criteria of Vilar's classification of translation errors (Vilar et al. 2006) as well as the adaptation we had to perform to use Blast (Stymne, 2011). Thirdly, we analyse the global results of our quality evaluation before going into details, in our fourth part, on one particular type of MWE, which is the Full Phraseme one. We finally show that most of the French MWE are translated into English MWE, and that we need to implement in further work a collaborative error annotation tool.

1 Introduction

Machine Translation (MT) nowadays is widely available for a large set of people; professional translators, students, researchers, common people. Nevertheless being able of using MT at its best is not an easy task. Most of the MT systems make regular mistakes, which means that there are typical syntactic structures, lexical items they cannot deal properly with. Therefore, the end user has to be able to detect and to correct those mistakes. The ability of the user to detect and correct the mistakes relies on his/her language skills, which means that some people would have the necessary skill to detect that the sentence is not grammatically correct and would correct it, but some others would not know how to correct it. The language skill level related to the post edition of MT is an interesting topic, which we won't discuss in this paper for a matter of length.

If we go back to the mistakes themselves, some of them are due to the fact that as MT systems use probabilistic algorithm they cannot focus on structures that does not appear very often. In this article, we are going to take this problem into account and thus deal with one of those structures that are the MultiWord Expressions (MWE). MWE are very common but not always in proportions that are sufficient for MT to give successful translations. However, as MWE are typical of the language, if they are mistranslated, the end user would consider the whole translation as a poor one even if it is not the case. Among others, previous work from Ramisch et al. (2013) on one type of MWE has already depicted the complexity of MWE translation.

We have to keep in mind that, in this work, only the quality of the MWE translation has been addressed. Consequently, the information of a sentence not being translated correctly is not given.

Proceedings of the 38th Conference Translating and the Computer, pages 33–41,
London, UK, November 17-18, 2016.

In order to study the quality of MWE translation, we based our work on the analysis of a corpus.

Our corpus is made of one technical document (12,566 words) written in French on which MWE have been semi-automatically annotated according to their type. As we based our work on a linguistic approach, we used a script to locate the more obvious MWEs, which we validated or not, and then we manually annotated those which were not found automatically or for which the type was not so clear-cut. This way we were able to refine the MWE type definitions as well as work on the inter annotator agreement. We have thus decided on purpose not to use automatic tools such as Ramisch et al. (2010) proposed.

2 Methods

2.1 MWE annotation

We semi-automatically annotated, as we said, the MWE present in our corpus according to nine types described in (Tutin et al., 2015). In this previous paper, MWE were addressed as "multiword elements that includes several graphical units, separated by blanks or hyphens, or separated by several other words not included within the MWE".

Out of those nine types, we decided to focus on five of them, which are the following ones:

- Function words (F),
- Full Phrasemes (PH),
- Collocations (C),
- Technical Terms (T) and
- Named Entities (EN).

Tutin (2015) defined those five types in a draft of Annotation guidelines for multi-word expressions. In the interests of brevity, we will just give a rough definition for each of the five types here.

First, Function words are characterized by a vague, and mainly functional, meaning. They include grammatical words such as conjunctions e.g. even if, or among others, prepositions e.g. in front of.

Second, Full Phrasemes include MWEs which are not compositional, e.g. couch potato, and/or are words, mainly nouns, which refer to specific referent, e.g. death penalty.

Collocations include frequent compositional expressions, e.g. heavy smoker.

Technical Terms, a subtype of full phrasemes, are mainly nominal full phrasemes typical of specialized corpora.

On top of the type, the MWE annotation also consists of the part of speech of the MWE, the part of speech of the elements of the MWE, and the overlapping of MWE is also annotated…

The French annotated document has been translated into English by a MT system created using Moses Toolkit (Koehn et al., 2007).

Furthermore, we have decided that the quality evaluation of the obtained translation, would not be done via automatic metrics but using more linguistics criteria such as those defined in (Vilar et al., 2005), which we are going to describe in the following section.

2.2 Error type classification

Actually, the criteria used in Vilar's classification of translation errors, as described in Figure 1, suit pretty well the linguistic evaluation we want to perform.

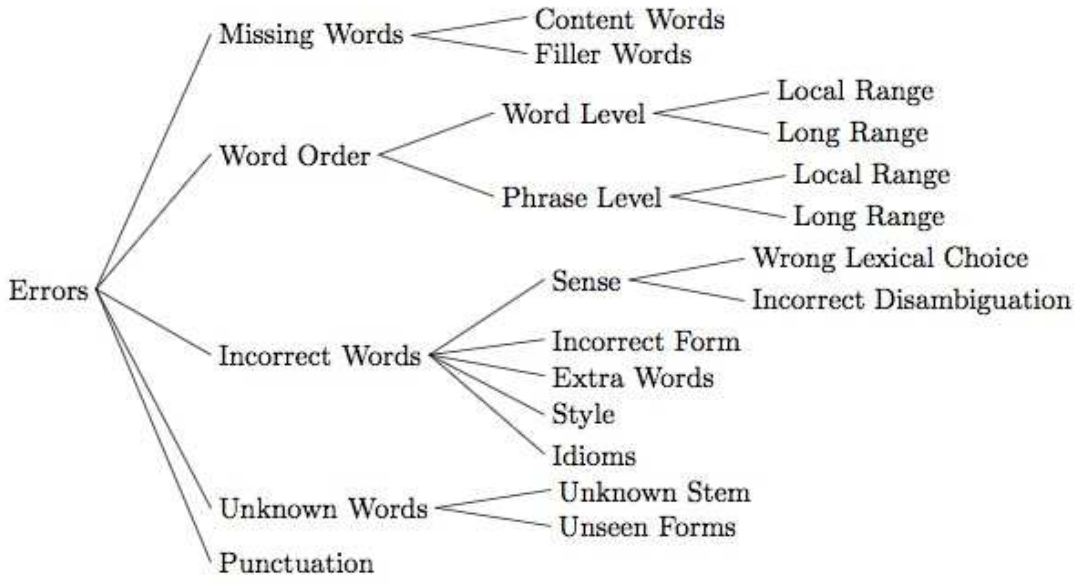


Figure 1. Classification of translation errors.

Five main translation error categories have been identified, namely:

- Missing Words, for words which have not been translated,
- Word Order, for a wrong order of the words in the translated sequence,
- Incorrect Words, for a mistranslation,
- Unknown Words, for words that were not known by the system and thus left in the source language,
- Punctuation, when punctuation rules of the target language were not respected.

As regards to the first two translation-error categories, Missing Words and Word Order, subcategories have been created to refine the error class. For the Missing Words error category, the distinction between Content and Filler words allows to see whether the missing word was meaningful or not. This subcategory illustrates the fact that the full meaning of the sentence was kept or not, which is obviously the aim of a quality translation evaluation. As regards to the Word Order error category, the Word or Phrase level subcategory shows if the translation error entails a reordering of the words themselves, or a reordering of phrases. It is well addressed to SMT evaluation as it permits to locate at which level the system failed, lexical level or syntactic level.

Looking at the third translation error category, Incorrect words, we can see that there are several subcategories aiming at distinguishing the reason of the mistranslation, which can be due to the fact that the system was not able to disambiguate properly the meaning of a source word nor to produce the right form of the word, although the base form of the word was well translated.

For the fourth translation error category, Unknown words, we can distinguish whether the stem of the words was known by the system or not.

And finally, the fifth translation error category, which is Punctuation, did not receive full attention from our part.

In addition, the part of speech of the word linked to the translation error category or subcategories, is also given. Thanks to Vilar's classification we were able to undergo a rich translation error annotation.

However, we wanted to track some more pieces of information. We have therefore added to the Vilar's classification the four following features. Hereafter, we will describe those additional features by defining them and giving their annotation abbreviation.

Firstly, we added the type of the MWE. The annotation of MWE in the source document was already giving that information, but due to technical difficulties we were not able to recover that information after translation. This is why we have decided to integrate the type of MWE into the translation error categories at the first level that is to say prior to any translation error category. We have used the abbreviation given in paragraph 2.1, e.g. EN for Named Entities…

Secondly, as we wanted to focus tightly to the translation quality evaluation of MWE, we wanted to refer to in-use translations of MWE. Consequently, for the most frequent MWE found in our source corpus, we have extracted from the bilingual concordancer Tradooit (https://www.tradooit.com), the related translations, considering them as the attested translations. We have thus integrated a category showing that the translation of the MWE was an attested one or not. We used the TA abbreviation when the translation was attested and the TNA abbreviation when it was not an attested translation.

Thirdly, we addressed the translation quality criteria by distinguishing four quality levels. When the source MWE was well translated, we used the BT abbreviation. When the source MWE was wrongly translated then we used the abbreviation MT for wrong translation. Then when the translation had to be edited but the meaning of the source sentence was kept, we used the abbreviation RevPres. And when the translation had to be edited but the meaning of the source sentence was not kept, we used the RevNPres abbreviation.

Finally, we wanted to annotate the fact that the translation was also a MWE in the target language, or not. We respectively used the abbreviations MULT and NONMULT. Consequently, Vilar's classification of translation errors has been extended to this scheme, Figure 2:

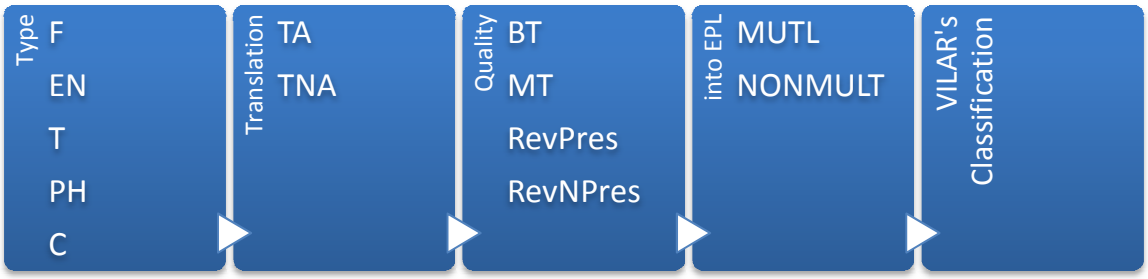


Figure 2. Extended Classification

2.3 Evaluation tool

The English translation has then been evaluated using BLAST (Stymne, 2011), an open source tool for the annotation of translation errors. We have chosen this tool as it has already been used for former MWE translation evaluation; and that it uses as a standard, and among others, the Vilar's Classification of translation errors, even if it can be used with any other hierarchical error classifications. Blast is also highly adapted to any evaluation purposes as it is not linked to the information provided by any specific MT. Furthermore, Blast is easy to

use because of its graphical interface. Among all the above, we have seen in Blast the way to add new annotations, edit existing annotations and also to search among the annotations.

Nevertheless we have experienced mainly two kinds of problems with the evaluation criteria. Firstly, we have encountered the borderline case of annotating several MWE in a same sentence. When several MWEs were present in a same sentence, it was impossible for us to link the translation error category respectively to the related MWE.

Secondly, as Blast ignores identical annotation, it seems impossible to have access to several error types at the same time for the same MWE. Also because of the hierarchical structure of the tool, we had to declare all the possible path of translation error annotation, thus having a linear path.

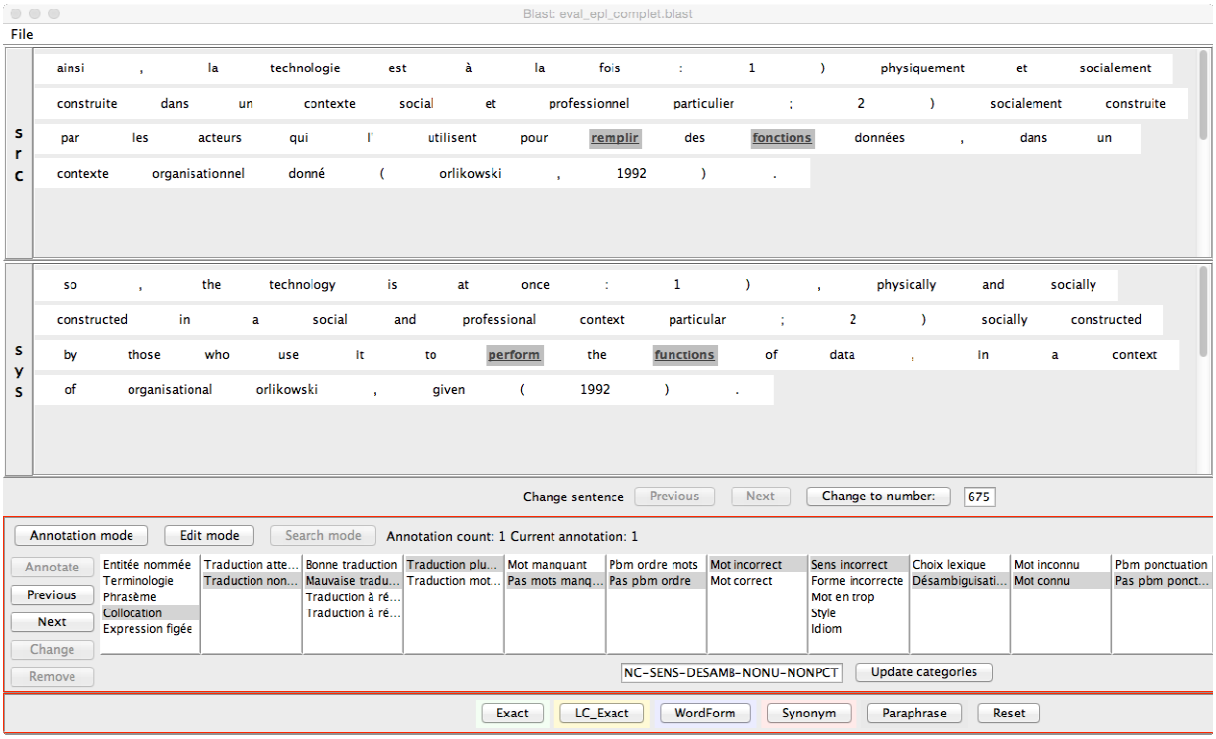


Figure 3. Blast Screenshot

The translation error annotation has thus been difficult. We could also mention that as the annotator has to make a decision, he cannot use the tool to trigger the attention on a difficult MWE to annotate thus asking for some help from another annotator. The possibility of having a collaborative tool would have been greatly appreciated for the MWE quality translation evaluation.

3 Results

3.1 Global results

Once the English translation evaluated, we have been looking at the global results which we have collected in Table 1 and Table 2 below.

As previously mentioned, out of the nine types of MWE, we have only worked on five of them. Those five types of MWE are represented thanks to their abbreviation, see above section 2.1, in the first column of Figure 4.

MWE type	Total number of occurrences	Attested Translation *TA*	Good Translation *BT*	Good Translation/ Attested Translation *BT/TA*	Good Translation/Non Attested Translation *BT/TNA*	Bad Translation/ Attested Translation *MT/TA*	Bad Translation/ Non Attested Translation *MT/TNA*
PH	135	64.4	64.4	96.6	6.3	0	66.7
C	308	63.3	72.1	96.9	29.2	0	22.1
F	202	78.2	81.2	98.1	20.5	0	36.4
EN	9	77.8	66.7	85.7	0.0	0	100.0
T	51	58.8	64.7	100.0	14.3	0	28.6

Table 1: Global results for 5 MWE types on Good Translation

Back to Table 1, if we look at the total number of occurrences per MWE type, it appears that we have mainly found in our corpus the Collocation type (C), with 308 occurrences. While the type of MWE we have found in the smallest quantity is the Named Entities (EN), 9 occurrences. Nonetheless, the ratio of bad translation over the not attested translation (MT/TNA), given in the eighth column, for EN is the highest with 100%. It means that when a MWE of the Named Entity type has been translated by a non attested translation it was also always a bad translation.

On the contrary, when a MWE of the Technical Term Type (T) was translated by an attested translation, it was always a good translation. The ratio good translation over attested translation, BT/TA, is 100% as mentioned in column number 5.

As a validity control, when an attested translation was given, it has never been a bad translation. The ratio MT/TA, appearing in the seventh column, is 0% for any type of MWE.

MWE type	Total number of occurrences	Attested Translation *TA*	Good Translation *BT*	Translation to be edited but source meaning kept/ Attested Translation *RevPres/TA*	Translation to be edited but source meaning kept /Non Attested Translation *RevPres/TNA*	Translation to be edited but source meaning not kept/Attested Translation *RevNPres/TA*	Translation to be edited but source meaning not kept/Non Attested Translation *RevNpres/TNA*
PH	135	64.4	64.4	1.1	18.8	0	8.3
C	308	63.3	72.1	1.5	32.7	0	13.3
F	202	78.2	81.2	1.3	40.9	0	2.3
EN	9	77.8	66.7	0.0	0.0	0	0.0
T	51	58.8	64.7	0.0	47.6	0	9.5

Table 2: Global results for 5 MWE types on Translation to be edited

In the same way, for validity control, we can notice on Table 2 that for an attested translation given, no translation evaluated to be edited with the meaning not being kept were found. For all the MWE types, the ratio RevNPres/TA, given in the seventh column, equals 0.

4 Detailed results focusing on Full phrasemes

We have decided to focus on the Full Phraseme type as they are the most difficult MWE to understand for non-native speakers. In the following, we are going to cover specific columns of Table 1 and Table 2, explaining the results by giving five examples, made of the French source, and of the English SMT Translation, and also of the related error annotation path for each example.

Actually, referring to Table 1, we find that 64%, see third column, of the Full Phrasemes were correctly translated, as we can see in the example 1. We can notice in that first example that the whole translated sentence is not correctly translated. As we already said, we have only processed to the quality evaluation of the MWEs, not to the evaluation of the whole sentence.

Example 1
French: […] leur statut d'embauche plus précaire fait en sorte qu'ils sont soumis à une forte pression […]
English SMT Translation: […] their status of employment more precarious done in such a way that they are under pressure […]
Error Annotation path: ph-TA-BT-MULT

In the fifth column of Table 1, we can see that more than 96% of Full Phraseme MWEs have been well translated when an attested translation existed. This result corresponds to the trend of the whole study. Nevertheless, roughly 6%, as written in the sixth column, of the Full Phraseme MWEs were well translated while the translation was not one of the attested ones. If we look at the example 2 below, the Full Phraseme faire état has not been translated in one of its attested translation but even so, it has been well translated by to present.

Example 2
French: […] nous allons ensuite faire état des méthodes […]
English SMT Translation: […] then we are going to present the methods […]
Error Annotation path: ph-TNA-BT-MULT

Going further in Table 2, we notice in the fifth column, that 1% of the translations has been evaluated as needing to be edited while the meaning was kept when an attested translation existed. Actually, example 3 shows that the Full Phraseme mis en oeuvre has been correctly translated as regards to its attested translation and meaning, but that its form was incorrect as the passive voice to be implemented was used in the translation while it should not.

Example 3
French: […] et les principales adaptations requises mises en oeuvre […]
English SMT Translation: […] and the major adaptations required to be implemented […]
Error Annotation path: ph-TA-REV_PRES-MULT-NONM-NONO-INC-FORME-NONU-NONPCT

Going to the next column, it emerges that a bit less that 19% of the translations have been evaluated as needing to be edited while the meaning was kept when the translation was not an attested one. As shown in Example 4, the Full Phraseme MWE pris en charge has not been translated by one of its attested translation, and that the translation proposed, i.e. taken over has taken the wrong lexical choice.

Example 4
French: […] la mécanisation et l' automatisation des procédés de travail dans l' industrie manufacturière ont été prises en charge par la production à la chaîne […]
English SMT Translation: […] the mechanisation and automation of working processes in the manufacturing industry have been taken over by the production of the chain […]
Error Annotation path: ph-TNA-REV_PRES-MULT-NONM-NONO-INC-SENS-LEX-NONU-NONPCT

In the last example, and referring to the last column of Table 2, only 8% of the translations have to be edited when it was not an attested translation of the Full Phraseme MWE. The Full Phraseme MWE en bout de ligne has been translated into at the end of the line and thus evaluated as needing to be edited with the meaning not being respected because of the use of Incorrect Words due to an Incorrect Disambiguation related to the Sense subcategory.

Example 5
French: mais en bout de ligne […]
English SMT Translation: but at the end of the line […]
Error Annotation path: ph-TNA-REV_NONPRES-MULT-NONM-NONO-INC-SENS-DESAMB-NONU-NONPCT

It would have been useful to check if the same MWE appeared several time in the document to verify the translation consistency. Is the MWE always translated in the same way, according to its place in the sentence, or to the syntactic pattern in which the MWE has been found? As Blast only allows to search for error categories, and that it does not permit to search for words or patterns, we could not proceed to such an investigation.

5 Conclusion

As a first conclusion, we have found that 80% of the MWE found in the French text were translated into MWE in English. As regards to the studied MWE types, the good translation rate is acceptable, showing that work has to be done in order to improve it.
As our corpus is not really big, one text of roughly 12,500 words, we want to draw the reader attention to the fact that this work is a first investigation of the MWE translation quality. Also, for some translation error annotations, the error annotation path was really long and thus some inconsistencies could arise.

Our second conclusion is then, that we would need another tool specifically dedicated to translation error annotations, with the possibility of selecting the source text and its target translation and to assign a translation error type. It will help identifying patterns in which specific translation error categories occur more often.

An extended work will thus consist in specifying a new collaborative tool dedicated to translation error annotation. Finally, a further work will be dedicated to deeply look at the quality translation results of the different MWE types studied.

Acknowledgements

This work has benefited from the AIM-WEST project (http://aim-west.imag.fr), which deals with the analysis and integration of MultiWord Expressions (MWEs) in speech and translation.

References

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi,Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst, 2007. Moses: Open source toolkit for statistical machine translation. In Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions , pages 177–180, Prague, Czech Republic, June. Association for Computational Linguistics.

Carlos Ramisch, Aline Villavicencio, Christian Boitet, 2010. mwetoolkit: a Framework for Multiword Expression Identification Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010, 17-23 May 2010, Valletta, Malta

Carlos Ramisch, Laurent Besacier, Alexander Kobzar, 2013. "How hard is it to automatically translate phrasal verbs from English to French?", MT Summit 2013 Workshop on Multi-word Units in Machine Translation and Translation Technology, Nice, France, September 2013.

Sara Stymne, 2011. Blast: A tool for error analysis of ma- chine translation output. In Proc. of the ACL 2011 System Demonstrations, pages 56–61, Portland, OR, USA, Jun. ACL.

Agnès Tutin, Emmanuelle Esperança-Rodier, Manuel Iborra, Justine Reverdy, 2015, Annotation of multiword expressions in French. Malaga, Espagne, Actes de la conférence Europhras2015, Juin 2015.

David Vilar, Jia Xu, Luis Fernando D'Haro et al., 2006. Error analysis of statistical machine translation output. In : Proceedings of LREC. 2006. p. 697-702.

InterpretBank. Redefining computer-assisted interpreting tools

Claudio Fantinuoli
Johannes Gutenberg Universität Mainz/Germersheim
fantinuoli@uni-mainz.de

Abstract

This paper presents InterpretBank, a computer-assisted interpreting tool developed to support conference interpreters during all phases of the interpreting process. The overall aim of the tool is to create an interpreter's workstation which allows conference interpreters to optimize the workflow before, during and after the event they are called upon to interpret. The tool takes into consideration the specific needs of conference interpreters, such as the way they prepare for a conference, the modality of terminology access, and so forth. It also exploits the latest advances in computational linguistics, especially in the field of information retrieval and text mining, making use of the abundance of information available on the Web to provide interpreters with specialized information which can be used to increase the quality of interpreter performance. The paper also introduces some theoretical principles of the use of terminology tools in interpretation and the results of initial empirical experiments conducted with this software.

1 Introduction

The use of information and communication technology (ICT) is ever-present in almost any professional, language-related activity, such as translation or copy-writing. While conference interpreting, like any other human activity, has been influenced by general advances in ICT, such has e-mail or the World Wide Web, the impact of interpreting-specific ICT solutions has been marginal. Some initial interest in the field of ICT applied to interpreting has been devoted to *setting-oriented* technologies, such as remote and telephone interpreting. Over the last few years, however, the focus of attention of practitioners and scholars seems to have moved towards *process-oriented* tools, such as terminology management, information retrieval, and so forth[1].

In the context of the latter area of interest, this paper presents InterpretBank, a computer-aided interpreting tool which supports interpreters during the various phases of an assignment, offering functionalities to prepare linguistically and extra-linguistically for a conference, manage terminological information and access the relevant terminology in the booth. The rest of the paper is organized as follows: section 2 introduces issues of linguistic and extra-linguistic needs of interpreters working on specialized conferences; section 3 presents a brief overview of computer-assisted interpreting (CAI) tools developed to date; section 4 introduces the main solutions and features proposed by InterpretBank; section 5 presents some empirical research conducted on CAI tools and on InterpretBank in particular. Finally, the conclusion summarises the topics introduced in this paper and presents some future perspectives.

2 Theoretical background: Linguistic and domain knowledge

Simultaneous interpreting, in particular at technical and scientific conferences, poses particular challenges to linguistic and domain knowledge that must be acquired prior to the assignment and that must be accessible during the interpretation itself (Thrane, 2005; Fantinuoli, 2016a). At a typical conference, interpreters are called to work for specialists who share background knowledge that is totally or partially unknown to people who are not experts in that particular field (Gile, 2009; Kucharska, 2009). As a consequence, interpreter-mediated communication

[1] See Fantinuoli (2016a) for more information on the difference between setting and process-oriented technologies in conference interpreting.

Proceedings of the 38th Conference Translating and the Computer, pages 42–52,
London, UK, November 17-18, 2016.

is characterized by a *knowledge gap* between interpreters and conference participants (Will, 2009; Fantinuoli, 2016a). This gap concerns both *linguistic knowledge*, especially related to terminology, and *domain knowledge*, i.e. expertise in a specific topic, information about the speaker, the situational context, etc.

To fill this gap, interpreters are called to do preparatory work beforehand. In fact, if translators can acquire much of this knowledge while translating, interpreters need to acquire it in the time before the conference. Preparation is considered essential to cope with the numerous difficulties that may arise during interpretation and that may be the cause of errors[2] (Pöchhacker, 2000). For this reason, over recent years many scholars have focused much attention on the preparation phase, especially on the way to define, constitute and access the knowledge needed to perform well at a conference (Fantinuoli, 2006; Rütten, 2007; Will, 2009; Stoll, 2009; Díaz-Galaz et al., 2015).

In the context of highly specialized conferences, a central role, in terms of knowledge acquisition and access, is played by ontologies and term collections. They are required to create the knowledge system needed to achieve precise and shared comprehension (see Morelli and Errico, 2007). However, during the act of interpreting, the processing of specialized terminology may be a cause of saturation (section 4.4) and, together with deficiencies in the understanding of the subject, one of the main reasons for errors or imprecisions.

The knowledge system that interpreters constitute in the preparatory phase is typically recorded in multilingual glossaries (or to be more precise the terminological part is included while the domain knowledge – definitions, conceptual systems, example of usage, etc. – is generally omitted). Once in the booth, the terminological information has to be actively retrieved, with or without external support, because even if several strategies are adopted to avoid the use of terminology, such as paraphrasing, hyponyms, or even omissions, the use of precise, correct and shared terminology remains a prerequisite to achieve efficient communication and to ensure that interpreters are perceived as a competent actor in the communication setting[3].

Software developers and several scholars have suggested that CAI tools could support interpreters in better rationalising and organising the process of knowledge and terminology constitution and its deployment during the task of interpreting (see for example Rütten, 2007; Will, 2009; Stoll, 2009; Tripepi Winteringham, 2010). In the next two sections the tools developed to date will be briefly discussed (section 3) and the basic ideas of InterpetBank will be discussed (section 4).

3 An overview of computer-assisted interpreting tools

The number of software applications developed for computer-assisted interpreting is very limited. They are quite heterogeneous in terms of their architecture, scope and functionalities and reflect the ideas and habits of the respective developer, generally an interpreter himself, more than the actual needs of the interpreter community. As CAI tools are relatively new, there is still no evidence of the advantages or disadvantages of their use. Among the different software available, there is no shared view on which approach or feature can best meet interpreter requirements and expectations.

In recent years, some initial attempts at evaluating CAI tools have been made. Yet, no

[2] Stoll (2009) states, for example, that an insufficient preparation can cause an increasing cognitive load during interpretation. This leads to a poor text analysis, memory activation and text production. As a consequence, the interpreter needs to apply "repairing strategies" with negative consequences on the quality of interpretation.

[3] Surveys conducted among delegates of technical and medical conferences indicate that "correct terminology" – among other interpreter-related qualities – is considered critical for the perceived quality of interpretation services, see for example Kurz (2001).

sound methodology or golden standards have been advanced. Some initial studies propose an articulated way to asses the tools (see Costa et al., 2016), but the priorities set and the weighting schemes are somewhat questionable. So, for example, the number of possible working languages is given a elevated ranking, though it is arguable that the presence of a high number of language labels may have some influence on the software usability. Similarly, the number of exported formats is awarded elevated importance, even if the formats may only be for internal use and therefore irrelevant in terms of software usability. Somewhat surprisingly, the applied weighting scheme does not take into account the presence of a search mechanism designed to cope with the peculiarities of the interpreting process, possibly one of the most distinctive and peculiar features of CAI tools (section 4.4). A more interpreter-oriented assessing system has been proposed by Will (2015). Based on the DOT terminology model (Will, 2009), the author identifies six categories to asses CAI tools, from the presence of a simultaneous modality to the flexibility of the data viewing system. Notwithstanding, the applied weighting system seems to be somewhat arbitrary and the conclusions unmotivated.

Without sound criteria for the assessment of CAI tools at our disposal, a fairly broad categorization can be tentatively proposed on the basis of the architecture and functionality spectrum they offer. Accordingly, CAI tools can be divided into two groups: *first-generation* and *second-generation* tools. *First-generation* tools are programs designed to manage multilingual glossaries in an interpreter-friendly manner, but do not envisage any other specific supporting activity of the interpreting process, such as information retrieval, text management, etc. The list of first-generation software is comprised of Interplex, Terminus, Interpreters' Help, LookUp and DolTerm. Only Interplex[4] and Interpreters' Help[5] are actively maintained and are commercially available. They are graphical interfaces designed to store and retrieve terminological data from a database and differ from terminology management systems for terminologists and translators as they use simple entry structures and offer some form of dedicated functionality to lookup glossaries in the booth. All tools can store additional information to the terms in explicitly or implicitly dedicated fields and allow the categorization of entries through a one-tier or a multi-tier categorisation system. None of the first-generation tools implement any sort of advanced search algorithm that takes into account the time constraints of the interpreting task, such as misspelling correction, progressive search in one or more glossaries, etc.

Second-generation CAI tools address the goal of extending the limited scope of first-generation CAI software building on initial academic research and investigations on terminology and knowledge management, proposing a more holistic approach to the interpreting task. They offer advanced functionalities that go beyond basic terminology management, such as features to organise textual material, retrieve information from corpora or other resources, learn conceptualised domains, and advanced search functions. The second-generation tools developed to date are Intragloss[6] and InterpretBank[7]. Again, the two tools are very different in terms of approach and functionalities. Intragloss focuses on the preparatory phase of an assignment and presents a novel approach to glossary building, as it is based on the interaction between preparatory texts and the terminological database. Among other things, it supports creating glossaries directly from within the preparatory documents or websites by highlighting a term in the document and searching for its translation in online resources such as glossaries, databases, dictionaries, etc. As in all other tools, terminology can be organized by domain or assignment. For conference preparation, it automatically extracts all the terms from the

[4] www.fourwillows.com/interplex.html
[5] www.interpretershelp.com
[6] www.intragloss.com
[7] www.interpretbank.com

domain glossary that appear in the preparatory documents, thus directly linking the texts with the available terminology repository. As far as the glossary lookup is concerned, it offers basic functionality in line with first-generation CAI tools.

The second tool, InterpretBank, is the object of this paper and will be presented in the next section.

4 InterpretBank

InterpretBank is a software developed as part of a doctoral research project at the University of Mainz/Germersheim. The overall aim of the tool is to create an interpreter workstation which allows conference interpreters to optimize their workflow before, during and after the event they are called upon to interpret[8].

InterpretBank has a modular structure including a corpus-based preparation utility which comprises automatic text collection and terminological extraction (4.1), an editor designed to create and manage specialized glossaries (4.2), a memorization utility to support interpreters in learning conference related terms (4.3) and a dedicated conference modality to access terminology in the booth (4.4). The modules are independent pieces of software designed to cope with a particular task of the interpreting workflow. However, as a toolkit they interact seamlessly with each other.

In the next sections, the theoretical principles for the development of the modules and the most unique features implemented within the tool will be presented.

4.1 Collecting corpora and extracting linguistic information

Conference preparation is generally time-consuming and interpreters often experience the feeling of not knowing exactly how to perform this task efficiently. To help interpreters rationalise this activity, a computer-assisted approach based on corpus exploitation is proposed (Fantinuoli, 2006, 2011). Adapting the corpus-based approach originally developed for L2 acquisition (see for example Carter et al., 2007) and for translation tasks (see for example Zanettin, 2002), the author introduces the corpus-driven interpreter preparation (CDIP) as a means to make the process of linguistic and domain knowledge acquisition "terminology driven", i.e. from the terminology to the conceptual structure of a particular domain.

This approach attempts to solve the dichotomy between terminology-oriented preparation and domain-oriented preparation which has been described by Gile (2009):

> [...] interpreters experience very concretely the deleterious effects of insufficient familiarity with technical terms that are used in conference. Since very little time is available for advanced preparation, they generally have to choose between primarily extralinguistic preparation and primarily terminological preparation. Most of them give preference to terminology [...].

The CDIP involves the idea that corpora can be the source of a potentially endless "serendipity process" (Bernardini, 2001), as one term can lead to another, depending on the interpreters' intuition and requirements. In this approach, interpreters can "explore" the corpus starting from a list of specialized terms and learn them in real context, discovering their meaning and the way they are used by domain experts.

On more practical terms, the CDIP can be seen as an improved way to read preparatory documents as concordancers grant the possibility to read textual material in a dynamic and linguistically-motivated way. For example, the visualisation of word patterns, which is typical

[8] For a detailed overview of the interpreting phases, see for example Kalina (2007) and Will (2015).

of concordancer tools[9], can help users infer meaning and usage (in context) of a term and discover relevant collocations, in this way supporting them in extending both their passive and active linguistic knowledge.

The starting point of this kind of preparation is a list of terms and a corpus of specialized texts. The corpus-based module has been designed to automatically build specialized comparable corpora from the Web using a small set of terms, e.g. the titles of the conference speeches[10], and to extract the most important terminology and phraseology of the domain[11].

The terminology extraction algorithm extracts the relevant (monolingual) terminology using a hybrid method which combines morphosyntactic rules and statistical measures. Since the extracted terms are statistically motivated, i.e. their status as candidate terms is based on the relevance for the domain, they can be considered important for the assignment and included in the conference glossary. Once imported in the glossary editor, this list of terms can be processed to find suitable translations using the features described in section 4.2.

4.2 Creating and managing glossaries

The glossary editor is a module designed to create and manage assignment-based glossaries. Besides common database functionalities, such as data filtering, merging, etc., the tool integrates a series of features to support the user in the compilation of new glossaries, such as automatic translation, lookup in online terminology databases (e.g. the terminology database of the European Union[12]), a definition retrieval system for finding information on specialized words as well as a concordancer to integrate the preparatory documents and find examples of term use in real contexts. These functions are designed to integrate seamlessly with the first step described in section 4.1 as they allow to find translations suitable for the initial monolingual list of terms.

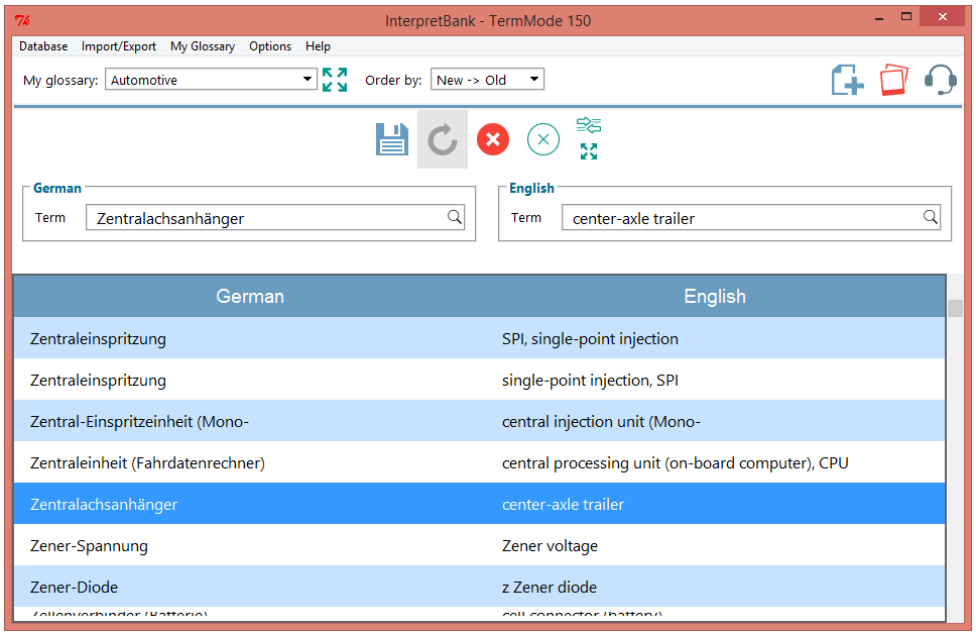


Figure 1: Glossary editor with simple visualisation mode.

According to Gile (2009), interpreters tend to add very little information to their glossaries.

[9]For a translator-oriented concordancer, see for example Fantinuoli (2016b).
[10]The corpus creation procedure is described in more detail in Fantinuoli (2012).
[11]The term extraction procedure is described in more detail in Fantinuoli (2006).
[12]www.iate.eu

For this reason the predefined data structure is simple and setup procedures can be avoided. Besides the term and its translation, it includes only a few general fields for storing extra information at concept and term levels, such as definitions, personal notes, etc. The user can choose between a simplified view, showing only terms and their translations, and an advanced view containing extra information too. In order to foster terms' re-usability, all glossaries are saved in a single database and can be categorized for disambiguation with a two-tired classification system, for example *domain+subdomain* or *conference name+domain*, etc.

The user interface differs from traditional solutions used in the language industry, as it integrates a terminological card view with a tabular treatment of data which "reflects the retention of a paper-oriented presentational view of terminological information" (Wright and Budin, 2001, p. 575). This seems to be the visualisation structure preferred by interpreters to organize their terminology.

The glossary can be accessed in memorization modality (section 4.3) or in conference modality (section 4.4).

4.3 Memorizing terminology

The peculiarities and time-constraints of simultaneous interpreting requires the ability to quickly process source text information, the terminological part, among other things. It goes without saying that in order to make simultaneous interpretation possible, terminology equivalents in the two working languages need to be at the interpreter's immediate disposal. Since it does not feasible to rely exclusively on external glossaries to lookup terminology in the booth (section 4.4), it is necessary to memorize such equivalents, at least for the most frequent technical and scientific terms. To help the interpreter achieve this goal, a memorization module (MemoryMode) has been implemented within the software. Its function is quite trivial: single glossaries can be visualized in a flash card interface which alternates the term and its translation.

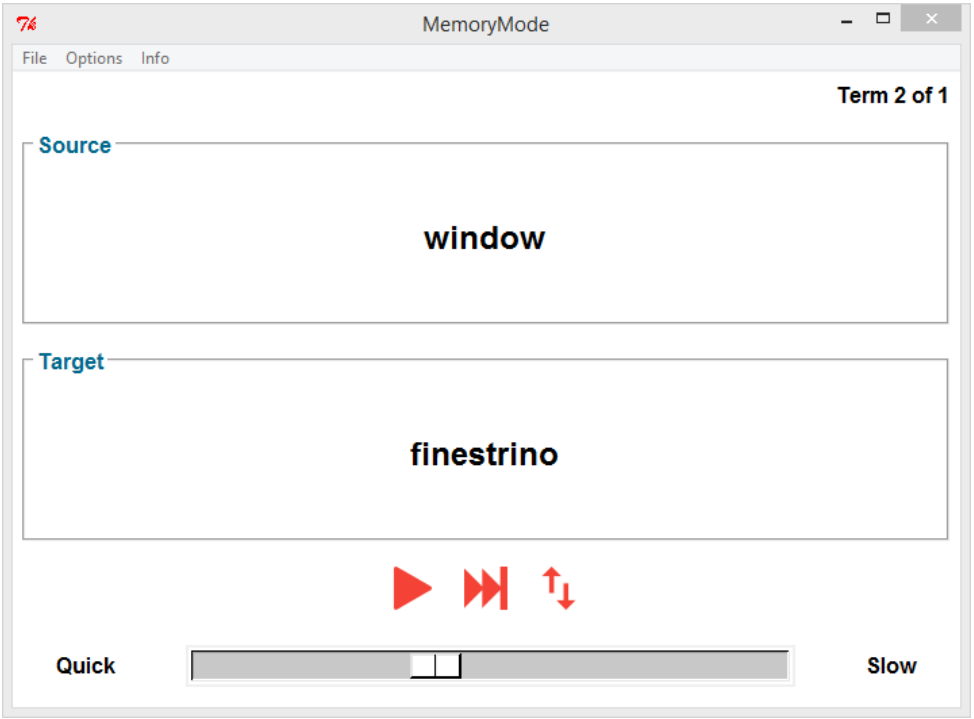


Figure 2: Memorization modality.

This system supports interpreters in visually drilling and automatizing the translation of the

conference terms. With the goal of making the tool as flexible as possible, speed, language direction and term order can be set by the users. It goes without saying that this basic way of memorizing terms is suitable only when exact equivalences for the same concepts are possible.

4.4 Accessing terminology in the booth

During simultaneous interpreting, listening to the oral text and producing the translation are time-delayed. In this time span, called *décalage*, many co-occurring processes take place, such as text analysis, activation of background knowledge, self-monitoring, etc. (see Gile, 2009). In this phase, the interpreter is also called to find translation equivalents for specialized terminology used by the speaker (section 2). All processes taking place during interpretation are interconnected and depend on the availability of sufficient cognitive processing capacity which is divided among the co-occurring processes (see Gile, 2009).

The difficulties connected with the use of specific terminology, especially if the text is terminology-dense, may result in an overall saturation or a saturation in one of the efforts. This can be the cause of errors, omissions and, in most broad terms, can determine a poorer quality of interpreter performance. In order to avoid a cognitive overload of the interpretation task, interpreters need to have at hand the right term at the right moment without interfering with the balance of the other various processes. At best, they should have "interiorized" the bilingual terminology so that it can be at their disposal with minimal effort. Yet, it is virtually impossible to memorize all terms used at a specialized conference. Furthermore, the spontaneity of most speeches makes it difficult to predict beforehand which terms will be used. Due to the above reasons, interpreters tend to learn only the terms they assume to be the most important ones, omitting less frequent or highly specialized words.

In order to cope with this shortcoming interpreters may employ terminology tools during simultaneous interpretation. Since any activity or disturbing factor that does not directly focus on the traditional "translational process" (Pöchhacker, 2009, p. 97) should be reduced to a minimum, at least three conditions should be met to allow the use of a terminology tool:

- by means of anticipating part of the cognitive load from the interpretation to the preparation, for example with the approach in section 4.1, interpreters release resources that can be used for other tasks while in the booth (Kalina, 1998; Stoll, 2009);
- the looking up activity should be selective and focused;
- the tool should be designed to minimize the cognitive load added to the interpreting process.

The last point plays a central role in the design of a CAI tool. In order to be booth-friendly, the searching mechanism needs to behave quite differently from the established terminology tools used by translators because it needs to take into account the time constraints and the complex cognitive processes governing simultaneous interpretation. In broad terms, the interface should be designed to simplify the interaction between user and machine, reducing the cognitive effort necessary to use it, speed up the search process and produce a suitable output in terms of visualisation and number of results.

Despite the lack of empirical evidence (section 5), it can be hypothesized that in order to minimize the cognitive load of the querying process, i.e. to make the tool truly booth-friendly, a search tool should achieve at least the following goals:

- have a user interface which is clear and unobtrusive
- require a short user's input

- produce pertinent results
- offer a clear visualisation of results
- not be influenced by spelling errors

Compared to traditional dictionary interfaces, for which the user has to enter the entire word or to make a choice among suggested matches, ConferenceMode is designed to accept partial words, possibly containing spelling errors, without affecting the reliability of the results. It implements two methods to start a query: the traditional method requires entering a series of characters into the search field and starting the query with the Enter key. In order to reduce the number of keyboard strokes, and consequently minimize the cognitive load, the second method performs the query progressively without the need to press the Enter key. The main idea is straightforward: every keyboard event triggers a new query. The query is run recursively with any new character until a small number of possible translations is retrieved for the user to comfortably choose from the results. At this point the search operation is concluded and the software is automatically ready for a new query.

In order to further reduce the cognitive load needed to query the database, the tool uses fuzzy search (which acts as an error correction mechanism for misspellings typed by the interpreter or saved in the glossary), stopword exclusion to reduce the number of matches displayed as well as the use of diacritic and accent-insensitive searches. This function avoids the need to repeat a query, which is not possible in the context of interpreting, if the original query has not produced the desired result because of a spelling error or the like.

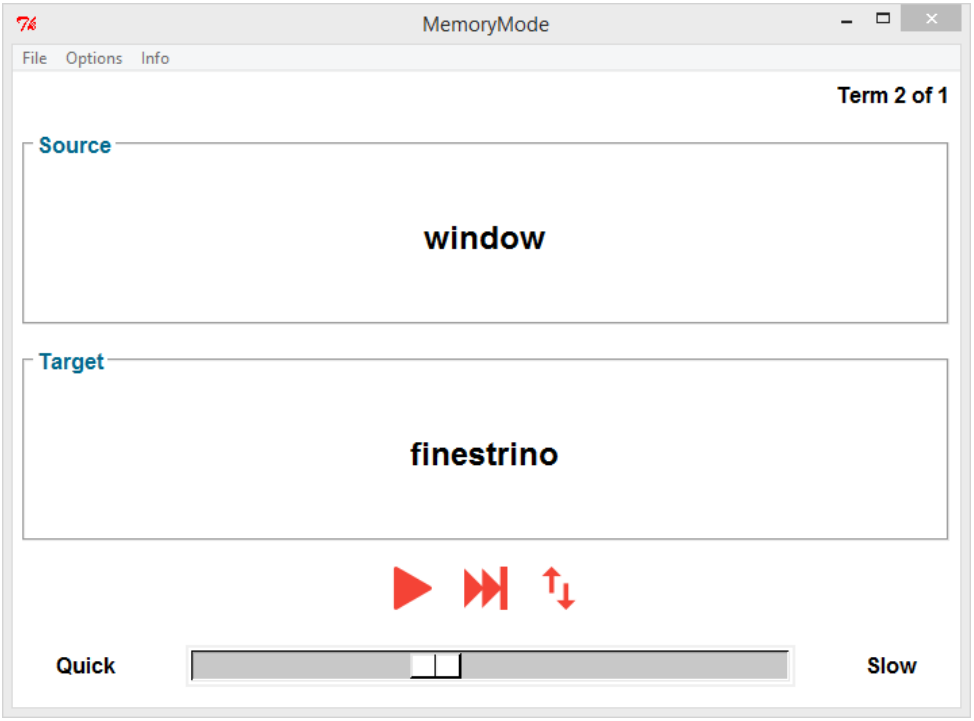


Figure 3: Conference modality.

As the interpreter at work may not have the time or the cognitive ability to detect and choose the correct translation of a specific term among the myriad of possible solutions that are generally offered by dictionaries (Donovan, 2006; Tripepi Winteringham, 2010), InterpretBank works on the basis of conference-related glossaries. In order to discriminate results according

to the conference topics and their relevance for the assignment, ConferenceMode performes a query progressively in the hierarchical structure of the database. The user can set a glossary priority in order to have one or more glossaries as the default assignment glossary. This will be the primary set of data searched by the tool, as it is considered the most relevant for the event the interpreter is called upon to interpret. If no result is produced, the module automatically extends the query to the entire database. In this way, the tool provides a means for term decontextualisation based on the domain or the conference subject, but allows for taking advantage of the entire terminological repository (which is conference or domain independent) collected by the interpreter.

Too much information can be an obstacle for the interpreting process in the booth. For this reason, the additional terminological fields which may be part of an entry (context, grammar, examples, etc.) are not shown by default, and only the terms and their translations are seen. If needed, however, the interpreter may choose to view this extra information too.

5 Empirical research

Even if the interest among practitioners seems to have increased during the last few years, CAI tools have remained fairly marginal in the growing body of interpreting studies, as confirmed by the small number and scope of publications dedicated to this topic. This is particularly true for experimental studies. However, the first papers and final theses on this topic have been published. They are particularly important as more knowledge about the use of CAI tools is needed to evaluate the tools and to pave the way to their further development.

In the context of tools dedicated to interpreter preparation, Xu (2015) experimentally investigated how the Corpus Driven Interpreter Preparation, as described in section 4.1, can improve trainee interpreters' performances. In the experiment, the tasks involved were the building of small comparable corpora as well as the use of automatic term extractors and concordance tools to discover terminological and factual information about a specific topic. The results show that the test groups consistently had better terminology performance during simultaneous interpretation, interpreting more terms correctly, having higher terminology accuracy scores and making fewer terminological omissions. This had an impact on the holistic interpreting performance scores which were higher than in the control groups.

As far as the use of InterpretBank in the booth is concerned, Gacek (2015) empirically analysed if its use improved interpreter performance in terms of terminological quality. Based on the experimental data and the participants' comments, the study shows that the use of a booth-friendly search tool is more efficient in improving the terminology rendition in terms of correctness and completeness than other more traditional solutions. Similar conclusions were drawn by Biagini (2016) who correlated the empirical findings of his experiment with the responses provided by the participants in a survey. In his empirical study, conducted with stringent scientific criteria and with the use of advanced statistical measures, the author aimed at answering the question if the use of an electronic glossary in the booth could be seen as a disturbing factor in the interpreting process or if it could provide support, even for novice interpreters. He compared the interpreting performance of a terminology-dense text of two groups of testers, the first using InterpretBank and the second a traditional glossary on paper. The results of the experiment show that all testers had a better performance when using the software. They were able to search and correctly translate a larger amount of technical terms, reducing term omissions at the same time. The author suggests that the improved terminology performance could be due to the fact that CAI tools reduce the cognitive load needed to look up terms when compared to other traditional methods.

Another area of interest for empirical research is related to the didactics of CAI tools.

As some universities recognise the need to adapt their curricula to the emerging use of new technologies in interpretation, a pilot study was conducted at the University of Bologna to understand if it is recommendable to integrate CAI tools in their curriculum (Prandi, 2015). The aim of the experimental study was to collect information on the students' approach to InterpretBank in the booth while interpreting terminology-dense texts. The experiment showed that most testers were able to conduct effective terminology searches (with an average 90% rate of terms correctly identified), even if they still were novice interpreters. The analysis of audio/video as well as keylogging data showed that the amount of experience in the use of the tool plays a key role in helping students integrate it in their workflow. As a drawback, the author stressed the tendency of some testers to rely too much on the software, with obvious negative consequences on the overall performance. The author concludes that CAI tools can be successfully integrated into the curricula of future interpreters, provided they already have ample experience in interpreting (for example at the level preceding the final exams, as the texts need to be of a rather specialised nature) and enough time to understand how to adapt their interpreting strategies to the use of the tool.

6 Conclusion

In this paper we proposed some of the ideas and motivations for the development of InterpretBank and described some of its functionalities. Our software supports professional interpreters during the linguistic and domain-oriented preparation of an assignment as well as during interpretation in the booth with the overall goal of increasing interpreter performance. Initial empirical studies have been conducted on this tool to analyse its impact on the didactics of interpreting as well as on the interpreter's performance. Though initial results seem to be encouraging, it is still premature to draw any definitive conclusions about the advantages and disadvantages of the use of similar tools. More process and product-oriented research in this area is required. ITC is advancing quickly and is opening new perspectives in the area of CAI tools. Speech recognition, for example, could represent the next step in the evolution of CAI tools. It could be used to automatically extract terminology in real-time from the interpreter's database or to show name entities, numbers and the like on the interpreter's monitor. Results from empirical experiments would not only help us to better understand the way CAI tools influence the interpreting process but also give us suggestions on how to improve the available tools.

References

Bernardini, S. (2001). Spoilt for choice: A learner explores general language. In Aston, G., editor, *Learning with Corpora*, pages 220–249. CLUEB, Bologna.

Biagini, G. (2016). *Printed Glossary and Electronic Glossary in Simultaneous Interpretation: A Comparative Study*. PhD thesis, Università degli studi di Trieste.

Carter, R., McCarthy, M. M., and O'Keeffe, A. (2007). *From Corpus to Classroom.* Cambridge University Press.

Costa, H., Pastor, G. C., and Durán Muñoz, I. (2016). An Interpreters' Guide to Selecting Terminology Management Tools.

Díaz-Galaz, S., Padilla, P., and Bajo, M. T. (2015). The role of advance preparation in simultaneous interpreting: A comparison of professional interpreters and interpreting students. *Interpreting*, 17(1):1–25.

Donovan, C. (2006). Where is Interpreting heading and how can training courses keep up? Technical report, emcinterpreting.org.

Fantinuoli, C. (2006). Specialized Corpora from the Web for Simultaneous Interpreters. In Baroni, M. and Bernardini, S., editors, *Wacky! Working Papers on the Web as Corpus.*, pages 173–190. GEDIT, Bologna.

Fantinuoli, C. (2011). Computerlinguistik in der Dolmetschpraxis unter besonderer Berücksichtigung der Korpusanalyse. *Translation: Corpora, Computation, Cognition. Special Issue on Parallel Corpora: Annotation, Exploitation, Evaluation*, 1(1):45–74.

Fantinuoli, C. (2012). *InterpretBank - Design and Implementation of a Terminology and Knowledge Management Software for Conference Interpreters*. Phd thesis, University of Mainz.

Fantinuoli, C. (2016a). Computer-assisted interpreting: Challenges and future perspectives. In Durán Muñoz, I. and Corpas Pastor, G., editors, *Trends in E-Tools and Resources for Translators and Interpreters*. Brill, Leiden.

Fantinuoli, C. (2016b). Revisiting corpus creation and analysis tools for translation tasks. *Cadernos de Tradução*, 36(1):62–87.

Gacek, M. (2015). *Softwarelösungen Für DolmetscherInnen*. Master thesis, University of Vienna, Vienna.

Gile, D. (2009). *Basic Concepts and Models for Interpreter and Translator Training: Revised Edition*, volume 8 of *Benjamins Translation Library*. John Benjamins Publishing Company, Amsterdam, 2 edition.

Kalina, S. (1998). *Strategische Prozesse Beim Dolmetschen*. Gunter Narr Verlag, Tübingen.

Kalina, S. (2007). "Microphone Off" –Application of the Process Model of Interpreting to the Classroom. *Kalbotyra*, 57(3):111–121.

Kucharska, A. (2009). *Simultandolmetschen in Defizitären Situationen. Strategien Der Translatorischen Optimierung*. Frank & Timme, Leipzig.

Kurz, I. (2001). Conference Interpreting: Quality in the Ears of the User. *Meta : journal des traducteurs / Meta: Translators' Journal*, 46(2):394–409.

Morelli, M. and Errico, E. (2007). Le microlingue nell'interpretazione: esperienze professionali e didattiche. *Tradurre le microlingue scientifico-professionali*, pages 347–372.

Pöchhacker, F. (2000). *Dolmetschen - Konzeptuelle Grundlagen Und Deskriptive Untersuchungen*. Stauffenburg Verlag, Tübingen.

Pöchhacker, F. (2009). *Introducing Interpreting Studies*. Routledge, London.

Prandi, B. (2015). The use of CAI tools in interpreters' training: A pilot study. In *Proceedings of the 37 Conference Translating and the Computer*, London.

Rütten, A. (2007). *Informations- Und Wissensmanagement Im Konferenzdolmetschen*. Frankfurt am Main: Peter Lang.

Stoll, C. (2009). *Jenseits Simultanfähiger Terminologiesysteme*. Trier: Wvt Wissenschaftlicher Verlag.

Thrane, T. (2005). 'Representing Interpreters' Knowledge: Why, What, and How? In V. Dam, H., Engberg, J., and Gerzymisch-Arbogast, H., editors, *Systems and Translation*. de Gruyter, Berlin.

Tripepi Winteringham, S. (2010). The usefulness of ICTs in interpreting practice. *The Interpreters' Newsletter*, 15.

Will, M. (2009). *Dolmetschorientierte Terminologiearbeit. Modell Und Methode*. Gunter Narr Verlag.

Will, M. (2015). Zur Eignung simultanfähiger Terminologiesysteme für das Konferenzdolmetschen. *trans-kom*, 8(1):179–201.

Wright, S. E. and Budin, G. (2001). *Handbook of Terminolgy Management*. John Benjamins, Amsterdam & Philadelphia.

Xu, R. (2015). *Terminology Preparation for Simultaneous Interpreters*. Phd thesis, University of Leeds.

Zanettin, F. (2002). Corpora in translation practice. In *Proceedings of the Workshop Language Resources for Translation Work and Research*.

Why XLIFF and Why XLIFF 2?

David Filip
ADAPT Centre, O'Reilly Institute
Trinity College Dublin, University of Dublin
Dublin, Ireland
david.filip@adaptcentre.ie

Abstract

This is to inform the business and decision making communities among the ASLING audience about the high level benefits of bitext and XLIFF 2. Translator and Engineering communities will also benefit, as they need the high level arguments to make the call for XLIFF 2 adoption in their organizations.
We start with a conceptual outline what bitext is, what different sorts of bitext exist and how they are useful at various stages in various industry processes, such as translation, localisation, terminology management, quality and sanity assurance projects etc. Examples of projects NOT based on bitext are given, benefits and drawbacks compared on a practical level of tasks performed. The following is demonstrated: That bitext management is a core process for efficient multilingual content value chains; That usage of an open standard bitext creates a greater sum of good than usage of proprietary bitext formats; and finally: That XLIFF 2 is the core format and data model to base bitext management on.

1 Introduction

XLIFF can be expanded as XML Localiz**z**ation Interchange Format, as the specifications are being developed in US English and the XLIFF TC should expand to (OASIS) XML Localisation Interchange File Format, as the Committee was originally convened and named by a group of Ireland based translation buyers and tool makers (XML can be expanded as eXtensible markup language).
Let us start with the statement that ***XLIFF is the only open standard bitext format*** and actually this whole paper will be just an explanation of that statement and why that matters. To fully explain the above statement we must define and discuss the notions of 1) a bitext format and 2) its management, 3) open standard and finally 4) that there is no other such format and that 5) this is good. In the following, XLIFF means XLIFF Version 2.0, the published OASIS Standard (Comerford et al., 2014a), unless XLIFF 1.2 (Savourel et al., 2008), the OASIS standard superseded by XLIFF Version 2.0, is specifically mentioned. Whenever XLIFF 2 is mentioned, it means any of the XLIFF 2.x Versions, current (Comerford et al., 2014a) (Filip et al., 2016, the XLIFF 2.1, first public review draft) or future, as all those subscribe or will subscribe to the same core object model and are backwards compatible (Saadatfar and Filip, 2015, 2016).

2 Bitext

Let us first define bitext:

> *A structured (usually mark up language based) artefact that contains aligned source (natural language) and target (natural language) sentences. We consider bitex to be ordered by default (such as in an XLIFF file – defined below, an "unclean" rtf file, or a proprietary database representation). Nevertheless, unordered bitext artefacts like translation memories (TMs) or terminology bases (TBs) can be considered special cases of*

Proceedings of the 38th Conference Translating and the Computer, pages 53–68,
London, UK, November 17-18, 2016.

> *bitext or bitext aggregates, since the only purpose of TM as an unordered bitext is to enrich ordered bitext, either directly or through training a Machine Translation engine.*

(Filip, 2012) (Filip and Ó Conchúir, 2011)

This is clearly not the classical definition but a (natural) development of the original "bi-text" concept that was first introduced by (Harris, 1988). Melby et al. (2015) explain how Harris – although he first defined bi-text in a psycholinguistic sense, as an alignment of source and target segments in translator's mind – intended that this notion inform the development of translation technology, and so it did. Already Harris himself postulated that a bi-text manifestation can be the authentic result of an actual translation process or a result of a later alignment of the source and target documents – again in the mind of a reviewer or reviser or as an actual interlinear source and target text rendering. Back in 1980s, there were obviously no bitext formats or artefacts (in the sense of the Filip, 2012 definition) around and Harris considered bitext manifestations simply as interlinearly rendered segments of source and target. In translation technology, the pre-segmentation usually is rules (and exceptions) driven, but there can be also statistically or deep learning driven segmenters. Nevertheless, most of the current CAT (Computer Aided/Assisted Translation) Tools do allow for manual re-segmentation of the working bitext by the translator. Also Harris considers bi-text to be primarily ordered and (Melby et al., 2015) explain how in translation technology past and current the unordered translation memory is actually secondary to the ordered bitext or source and target document alignment. Clearly Translation Memories (TM) and bilingual (as a special case of multilingual) Term Bases (TB) or Glossaries are derived and special forms of bitext. These were created from bitext and serve to make better and more consistent bitext and target text in the future.

After this short discussion of the bitext notion, we should be happy enough to use the (Filip, 2012) (Filip and Ó Conchúir, 2011) definition of bitext as an up to date explication of the original Harris notion. Nevertheless, the fact that the notion of bitext is for all practical purposes older than translation technology itself shows how profoundly important bitext is for actually creating natural language translations. We can argue that bi-text (as the psycholinguistic paragon and the potential post hoc re-alignment) exists and plays an important role even in low tech translation workflows that don't make use of bitext in the sense of an actual translation processing and exchange format. It doesn't really matter how bitext is represented (interlinearly, side by side, serialized as rtf, XML or JSON or what have you), since bitext really is just an abstract data or knowledge structure. That's right, the representations don't matter BUT if they do represent the same things (data/knowledge objects) and if they have an effective and efficient way to exchange them (semantic interoperability). Both – the machine readable bitext artefacts and the psycholinguistic instances of bi-text in the mind of translators and revisers – are just manifestations of the same abstract knowledge or data structure that captures the source and target texts as mutually corresponding segments.

3 Bitext Management

Let's look now at Bitext Management:

> *[Bitext Management is a g]roup of processes that consist of high and low level manipulation of ordered and/or unordered bitext artefacts. Agents can be both human and machine. Usually the end purpose of Bitext Management is to create target (natural language) content from source (natural language) content, typically via other enriching Bitext Transforms, so that Bitext Management Processes are usually enclosed within a bracket of source content extraction and target content re-import.*

(Filip, 2012) (Filip and Ó Conchúir, 2011)

Translation industry is highly fragmented and the industrial workflow that produces fit for purpose translations for various translation buyers and target groups is highly distributed. One of the chief reasons for worldwide distribution of typical corporate, public sector or pro bono translation workflows is the best practice of using native speakers of the target language living within the specifically targeted locale. Other reasons include that translation needs tend to have discrete peaks and typical corporate organizations cannot create permanent capacities to translate content. Typically, an organization creates the necessary infrastructure and capacity to interface with external language provision providers. There are some organizations with a statutory flow of translation needs and those normally develop an in-house capacity that can handle the statutory flow. Organizations – such as European Parliament or European Commission, Canadian government et al. – with stable and high statutory flows produce more translations in house than via outsourcing. Nevertheless, even in organizations with low level of inter-organizational outsourcing, a good professional translation is a result of a concerted effort of at least two people, the translator and a reviser. Additional Quality Assurance (QA) steps will occur as spot-checks. It has become a best practice to inform the selection of and ratio of spot-checked translated content by a mixture of automated QA tools reports and human decision making given the relative importance, (lack of) usefulness of a given translated content etc. In any case, the bilingual reviewer performing the spot-check would be extremely inefficient if not presented with an actual bitext artefact that has been the medium and the result of the translation and revision process.

If we review current translation practices, from niche literary and art theoretical translations, over highly customized marketing transcreations, through product documentation in various areas, print manuals and user assistance, high volume knowledge bases, user generated and social media content, to crowdsourced and wiki based translation efforts, the need for bitext is ubiquitous; albeit it is not present in all cases in the form of an articulate exchangeable artefact that can be instantiated more than once without the loss of identity or integrity. For instance in a literary translation workflow there is rarely a tangible manifestation of bitext. Bi-text exists in the original Harris sense in the mind of the Translator as she works through the bulk of the book, treatise or what have you. Then this unique authentic alignment is lost and the Translator herself often struggles to recreate that unique initial alignment that she had in mind when originally creating the segment translation, when doing (multiple rounds) of self-correction or self-revision. In the next step, in good publishing houses, there comes a Translation Editor who performs a bilingual review and that person too needs to perform the task of re-aligning the source and the target as created by the Translator. Even in this simplest editorial workflow, there is arguably a lot of time waste, as even a single person workflow involves repeated re-creation of bi-text. Even tech hostile partisans of artistic translation or transcreation should see that the Translator would benefit from fixing their first bitext rendition of the source and target and that existence of such a fixed rendition would greatly facilitate the self-revision process. The waste entering with the second person in the process, the Translation Reviewer or Editor who is supposed to discuss choices and options of the translation with the Translator is beyond argument.

Some may argue (Melby et al., 2015) that pre-segmentation imposes a certain interpretation onto the Translator and may limit or preclude creativity in the process. In the workflows of industrial localisation, the weight of this argument is very limited and it is well addressed by the option provided by most current CAT tools to change the pre-segmentation at their working time. Workflows differ in how this is handled in the subsequent steps and obviously there is value and economy in actually preserving the segmentation choices and possible reordering of segments that the Translator performed during the translation. In the section

6 Overview of the XLIFF data structur, it will be shown how XLIFF supports re-segmentation and reordering of segments within higher level logical text units and exchanging of these modifications within an ecosystem based XLIFF roundtrip while maintaining alignment of corresponding source and target segments.

In industrial localisation, there is a non-negotiable requirement to exchange bitext between and among workflow agents, human or machine; the time-loss when distributing source text and translation suggestions to pools of single target language translators, the need for industrial revisers to instantly see the corresponding segments between source and target content simply cannot be served in any other way than through persistent bitext.

Historically, this requirement had been fulfilled by proprietary (albeit ad hoc or *de facto* standardized) formats such as the "unclean" rtf (rich text format). The early *de facto* standard bitext interchange formats had been driven by Trados, as the leading CAT Tools vendor. This is discussed with somewhat more detail in 5 Evolution and Adoption of Bitext formats.

Various types of organizations based on their priorities and availability of resources choose one of three possible approaches to bitext management. This partially depends on their current level of Localization Maturity (DePalma et al., 2006).

At very low levels of maturity, customers – and often also translators – don't know that they could benefit from working with an actual bitext representation in their translation process (Fig 1).

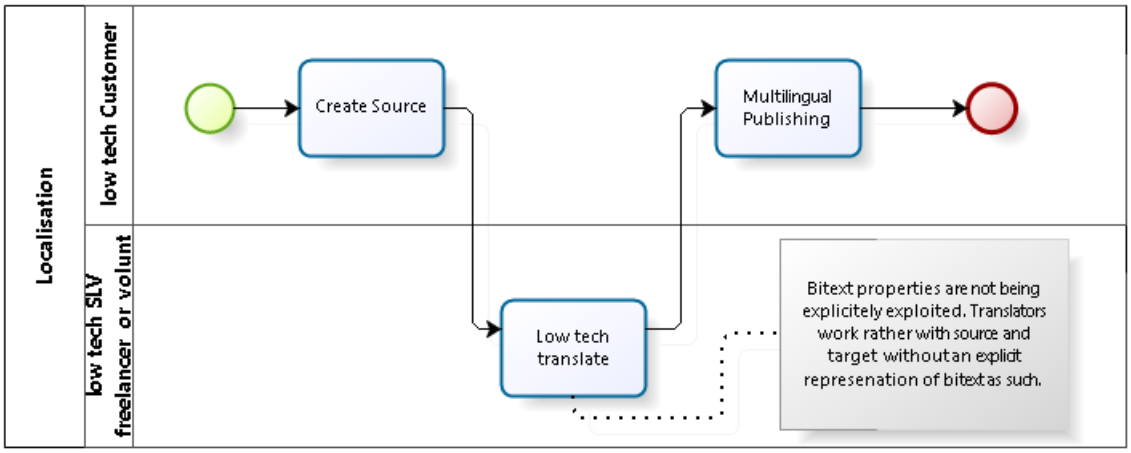


Fig 1: No bitext (just bi-text)

In case customers know about bitext they don't know language service providers or freelancers that could manage bitext for them. At low levels of maturity, the return on investment into having stable bitext representation isn't clear because it isn't measured or otherwise quantified in the organizations. Translators are either afraid to invest into CAT Tools that they could use to manage bitext or are not technically savvy enough to use free open source or cloud offerings.

In case of translators who use bitext privately (without telling their customers), we are drifting towards the second model (Fig 2). To be fair to the translator, the low tech customer usually doesn't care enough to know what the translator did or had to do, so that they could afford to provide their documentation translation at 4 cents per word or less and not perish in the process.

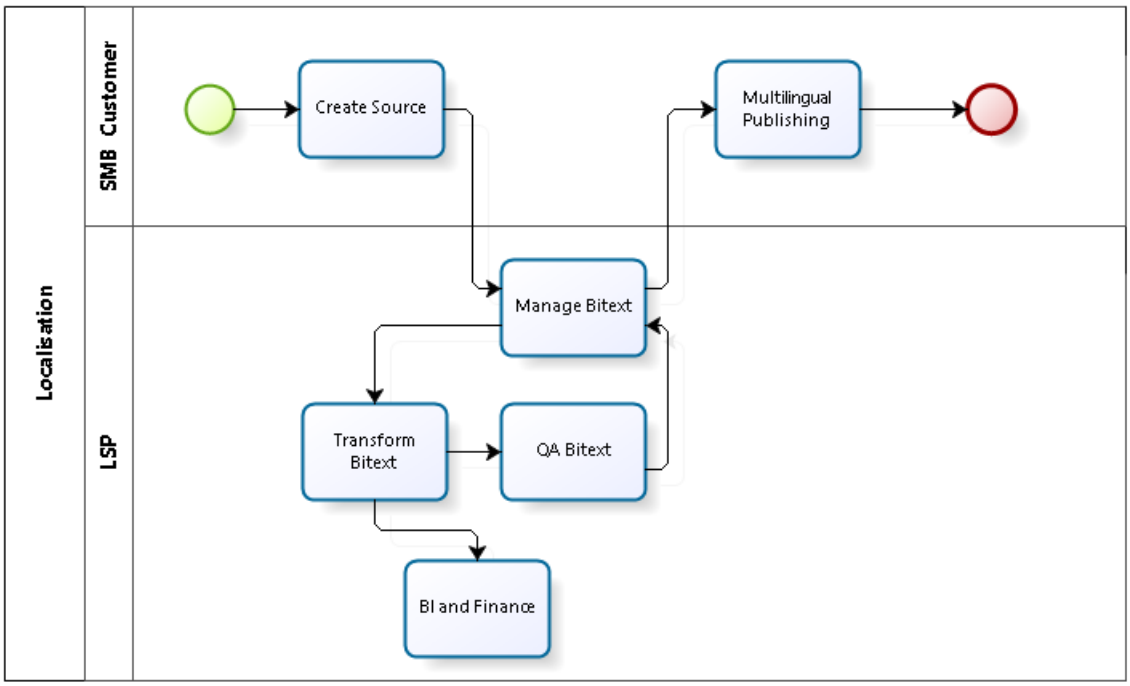


Fig 2: SMBs don’t manage bitext

Small and medium businesses (SMB) or other customers at average levels of maturity are usually aware that their language service provider benefits from explicit bitext management and they want to share the benefit. However they usually don’t care enough or don’t have enough resources to be able to manage the bitext themselves and they rely on their Language Service Providers (LSP) to do that for them.

Technology agnostic service providers in this model tend to use standard based interoperable solutions since they don’t want to manage multiple tools stacks for different customers and want to be able to effectively exchange work packages among their in-house staff (usually experienced revisers and proof readers) and freelancers (usually the translator performing the bulk of the jobs).

However, large service providers with their own technology and tool stacks are tempted to secure their revenue streams by vendor lock-in. Therefore, it is difficult to make the case for large service providers to use the standard bitext management format. This is a complex argument based on enlightened self-interest that brings us to the third model (Fig 3).

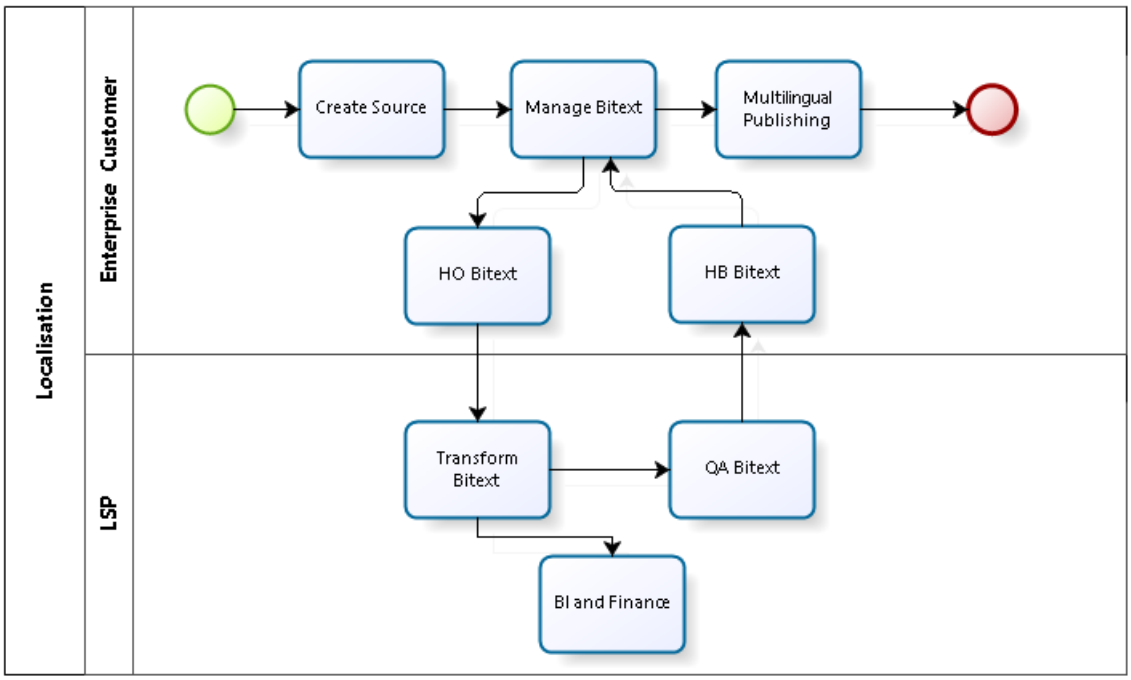


Fig 3: Enterprises do manage bitext

Large service providers usually crave for huge orders from large localisation services buyers. However, the very largest buyers are usually at fairly high levels of localisation maturity and definitely do understand the value of managing bitext and the role of managing it themselves in prevention of vendor lock-in. These very large buyers actually have the power to participate in standards development and the weight required to effectively request that their providers – no matter how big – play nice and along the standard bitext format. However, even large buyers have had and used to employ other options than employing an open standard bitext format. Obviously, automation is easiest in controlled environments and this led Microsoft to actually developing and maintaining a proprietary CAT Tool and bitext management solution for over 20 years. Only in 2011/2012, Microsoft decided against the ongoing maintenance cost of a technology tool stack outside of their core business that they would prescriptively enforce within their vendor base and decided to immediately employ XLIFF 1.2 in Windows user assistance localisation, to participate in the XLIFF 2 standardisation project and in general to require compliance within their vendor base in a technology agnostic and descriptive way based on the open standard.
Microsoft by the way donated their XLIFF 2.0 open source implementation, the so called XLIFF Object Model (King, 2015; Microsoft News, 2015; Petersen, 2015) to the localisation community and that code has been reused since by toolmakers building XLIFF 2 compliance in .NET environments.

4 Open Standards

What does that mean for a standard to be open? Please note that in standards, “open” is not the antonym to “closed” as in case of Software; open in standards is opposed to proprietary. Openness in standards doesn’t come as a binary notion, either open or proprietary; there are shades of grey with regards to standards openness. There are two major conceptual components to openness in standards: i) licensing and related cost ii) transparency of creation and availability. Rosen (e.g. Rosen, 2004, aka oslbook), Krechmer (2006), (Cargill, 2011) and

others provide rather clear criteria what a standard needs to be like to be considered open. However, the thinking on standardization develops over time and is informed by ongoing IPR (Intellectual Property Rights) litigation. Also the proprietary standardization has a substantially longer tradition than open standardization; the very notion of open standardization actually started to form as late as 2004 and actually no SSO (Standards Setting Organization) with truly open policies in both aspects existed before 2004. The thinking on openness in interoperability standards has been linked with the related but distinct notion of Open Source development.
Let us review what Openness Requirements are put forward by OSI (Open Source Initiative, 2012), the most influential legal framework around Open Source and Permissive Licensing of Open Source Software (OSS)

> ***The Requirement***
>
> *An "open standard" must not prohibit conforming implementations in open source software.*
>
> ***The Criteria***
>
> *To comply with the Open Standards Requirement, an "open standard" must satisfy the following criteria. If an "open standard" does not meet these criteria, it will be discriminating against open source developers.*
>
> 1. ***No Intentional Secrets:*** *The standard MUST NOT withhold any detail necessary for interoperable implementation. As flaws are inevitable, the standard MUST define a process for fixing flaws identified during implementation and interoperability testing and to incorporate said changes into a revised version or superseding version of the standard to be released under terms that do not violate the OSR.*
> 2. ***Availability:*** *The standard MUST be freely and publicly available (e.g., from a stable web site) under royalty-free terms at reasonable and non-discriminatory cost.*
> 3. ***Patents:*** *All patents essential to implementation of the standard MUST:*
>   1. *be licensed under royalty-free terms for unrestricted use, or*
>   2. *be covered by a promise of non-assertion when practiced by open source software*
> 4. ***No Agreements:*** *There MUST NOT be any requirement for execution of a license agreement, NDA, grant, click-through, or any other form of paperwork to deploy conforming implementations of the standard.*
> 5. ***No OSR-Incompatible Dependencies:*** *Implementation of the standard MUST NOT require any other technology that fails to meet the criteria of this Requirement.*

The above requirements are strictly speaking only met by standards developed under the Non-Assertion IPR mode. Even RF (Royalty Free) standards developed at W3C or OASIS might have theoretically an issue with the requirement #4, as traditional obligations of participants in committees and working groups chartered with the RF IPR mode don’t preclude license agreement conclusion or even negotiation. In practice however, such negotiations rarely happen or are replaced by granting of a Non-Assertion covenant even in case the IP was originally released for the scope of the standardization project chartered as a classical RF committee. The simple reason for that is that no one cares to spend money on expensive lawyers negotiating a licence grant at no cost.

To summarize, the only absolutely save IPR mode from the OSI point of view is the Non-Assertion that is being used only by OASIS and that from 2008, so that only the more recently formed Technical Committees could have actually been chartered under this IPR mode and it would be very difficult to impossible for older initiatives to re-charter with this progressive IPR mode. Nevertheless, the RF IPR mode – that is the only option at W3C and the most widely adopted option at OASIS – is generally safe in practice. But in case of standards from all other traditional standardization bodies such as ISO, Unicode, ETSI, IEEE etc. the IPR mode is invariably (F)RAND and thus openness of the standard for the practical purposes of Open Source development needs to be secured by legal frameworks extraneous to the original standardization framework, which constitutes a non-negligible legal risk for open source implementers and adopters.
Importantly, all of the above openness requirements are covered by the two aforementioned conceptual components. The requirement #1 *No Intentional Secrets* is in its description a requirement of Transparency of development and maintenance. 2# is a requirement of transparent publishing mixed with the licensing cost requirement. Interestingly, OSI does not preclude the potential cost of buying a copy of a standard, which is different from paying royalties dependent on the usage of IP used essentially in the standard. 3# is the requirement of no royalties in somewhat more detail. #4 is the most problematic and in essence it is an extension of the transparency of publishing requirement. #5 is again an extension of the development transparency. Secrets or hidden cost must not occur through dependencies. Hence we are going to review in a little bit more detail how XLIFF and related standardisation fares with respect to transparency of creation and availability as well as licensing and related cost.

4.1 Transparency of Standards Development and Publishing

This paper does not intend to provide an exhaustive overview how standards happen to be developed at a range of SDOs (Standards Developing Organizations) or SSOs (Standards Setting Organizations); we are going to use SSO onwards as a simple short term deemed synonymous with SDO, Standards Organizations, Standardization organizations and other similar terms in currency. Rather we want to demonstrate that the producer of XLIFF, OASIS XLIFF TC can serve as a paragon of transparency and persistent availability of open standards among SSOs in general. Transparency of the OASIS Technical Committee Process can be only rivalled with that of W3C. Not only all public review stages of an OASIS Standard Track Work product have persistent public URIs, but the whole standards creation process is fully publicly visible and that since the very inception of standardization projects at the stage of chartering new Technical Committees. All OASIS working lists including the ones dedicated to committee formation are publicly and persistently archived and every message ever sent to those lists has a persistent public OASIS URI; moreover, the mailing lists are effectively searchable through indexing services such as http://markmail.org/. All OASIS SVN repositories are publicly visible and accessible at https://tools.oasis-open.org/version-control/browse/, all OASIS GitHub repositories (Cover, 2016) that will be used to conduct Technical Committees' chartered work will be available at https://github.com/oasis-tcs/. This location will soon include OASIS XLIFF OMOS repositories for the work on the Abstract XLIFF Object Model (https://github.com/oasis-tcs/xliff-omos-om) and for the work on JLIFF (https://github.com/oasis-tcs/xliff-omos-jliff), the JSON serialization of the said Abstract Object Model. No OASIS TCs were conducting chartered work on GitHub at the time of writing this. Everyone can perform a full depth checkout of any of the SVN or GitHub repositories and inspect every line of specification, validation artefacts or sample code since the inception of each of the OASIS standardization projects.

Write access to OASIS working lists, wikies, and repositories (SVN or GitHub) is restricted to the TC members and active OASIS (TC) credentials are required to contribute. This restriction does not impede transparency; on the contrary, this restriction underpins the SSO's IPR Policy. OASIS TC members were authorized by the Primary Representatives of their employers (or are authorized as individual members) to contribute IP towards a chartered standardization effort as scoped in the TC Charter at the inception of the project. Charter, especially its parts defining scope and IPR are essential for the TC identity. If IP contributed to OASIS TCs' chartered work came from contributors who had not agreed to OASIS IPR policy or who had not been authorized to contribute IP towards standardization in the specific area scoped by the specific TC Charter, it would undermine the legal safety of the standardization deliverables and could lead to inability to publish work products due to copyright or patent infringement claims, or worse implementers of published work could be sued by copyright or patent owners for infringing on their IP by implementing the standard. This shows that transparency and IP management must strike a fine balance between flexibility and red tape to provide a legally safe framework for transparent development of Open Standard deliverables.

4.2 Availability of Standards under Royalty Free Conditions

Despite different schools of thought, there is a wide consensus among standardizers and IPR lawyers that standards purporting to be open need to give access to the embedded essential IPR (patented or not) at (F)RAND ((Fair) Reasonable and Non-Discriminatory) or more implementer friendly conditions. Rosen (2004), OSI and others argue that IT standards in the Internet age should be RF rather than just any sort of (F)RAND.

Again this paper does not aim to provide an overview on how SDOs manage Patent encumbrance of their standards. Rather we want to show that also in this aspect OASIS XLIFF TC and XLIFF OMOS TC are at the forefront of openness, ease and clarity of licensing conditions for potentially included patented IPR.

OASIS XLIFF TC Charter (XLIFF TC, 2014) was first put forward in late 2001 in the original Call For Participation (Best, 2001), the charter has been "clarified" four times since (2002, 2005, 2006 and 2014). Clarification here means performing editorial changes to the charter mainly to keep it up to date with specific developments. However, two things can never change in the process of subsequent clarifications, should the TC preserve identity: (1) The statement of purpose (including the high level technical scope) and (2) the IPR mode. Hence the original OASIS XLIFF TC needs to stay on the same IPR Policy, with which it had started back in 2001, i.e. the RF on RAND OASIS IPR Policy (OASIS Board of Directors, 2010: Sections 10.2.1 and 10.2.2).

Should one of these essential components of the Charter change, the original TC ceases to exist and a new one is being formed by the process of Rechartering. All IPR must be committed again to the newly formed TC if Rechartering took place. This is to make absolutely sure that the IP was provided and remains available under the same IPR Policy and for the same purpose. For instance the XLIFF TC in its Charter committed to conducting its work via development of XML Vocabularies, hence a "sister committee" – the XLIFF OMOS TC – had to be chartered when the wider stakeholder community agreed (Filip, 2015) that XLIFF needs an explicit abstract data model and an option to develop non-XML serializations, prominently a JSON serialization. XLIFF OMOS TC took the most progressive available RF IPR policy, the so called Non-Assertion IPR Policy (OASIS Board of Directors, 2010: Section 10.3) that had not been formulated at the time of the original XLIFF TC formation. Thanks to this progressive IPR mode, XLIFF OMOS can launch Open Source projects on GitHub under the OASIS umbrella.

5 Evolution and Adoption of Bitext formats

First Computer Aided Translation (CAT) Tools were just scripts over Rich Text Format (rtf). rtf had the advantage that it was widely supported by office suits and was not proprietary as the internal word-processing formats of the office suit producers. So the first technological representations of bitext were interlinear and the target translations were hidden from sight of the normal user not armed with the special script (or the skill to manipulate the visibility of the invisible style) making the other part of the bitext visible. This was a fairly reasonable idea and the first generation of these tools helped to build up the Translation and Localisation industry. Unfortunately, rtf (its implementations rather than the format itself) was not great for handling different scripts and the rtf based tools caused lot of headache in localisation engineers working between scripts or across text flow directionality.
XML started to be used in Localisation very early, as UTF-8 encoded XML provides a very robust and suitable data model for handling data in multiple languages. Pretty much all current and legacy localisation standards are indeed XML vocabularies. TBX was started as SGML but moved to XML also very early. Albeit XLIFF was available in some form or other as early as 2002, the first XLIFF Version that achieved a full OASIS ratification and the Standard publication status was XLIFF Version 1.2 in early 2008. It should be noted that the localisation providers were in general either working on rtf based tool stacks or using the Trados proprietary TTX format that – although not publicly documented – was widely understood and implemented also by tools other than Trados. The industry was immature (even more so than now) and there was little interest in an open standard format as long as the market leader Trados remained independent from major Language Service Providers. All changed when SDL made a series of tools acquisitions between 2005 and 2008 starting with the Trados acquisition in 2005. The buyers and some service providers decided to back the open standard to make sure that there was a viable bitext format outside of the direct control of a single Language Service Provider. This led to the XLIFF 1.2 success story but also to all lessons learnt from XLIFF 1.2 adoption between 2007 (before OASIS approval) until 2010 and beyond. In 2010, the XLIFF TC made the decision not to work on any other 1.x versions (not only not to add features in a 1.3 Version, even not to finish a 1.2.1 hotfix or 1.2 errata) and started pursuing the XLIFF 2.0 design.

6 Overview of the XLIFF data structure

As explained in **2 Bitext** and **5 Evolution and Adoption of Bitext formats**, XLIFF is a bitext format. In **6 Open Standards**, we explained how XLIFF is an Open Standard (transparently developed, maintained, available and Royalty Free). Now is the time to show how it is fit for purpose.
It so happened that the development of Localisation standards largely coincided with the advent and rapid rise of XML, the eXtensible Markup Language (Bray et al., 2008) and so XLIFF, as well as several other Localisation standards, happens to be an XML Vocabulary. XLIFF 1.2 as well as XLIFF 2 are XML vocabularies. Nevertheless, there is a specific data structure behind the XML serialization and the specifics of the serialization as XML or something else, or even as different XML styles is something accidental. It stops being accidental though, as soon as a tool, tool chain or an ecosystem of tools starts relying on the particular serialization. This kind of reliance on a serialization for *processing* purposes can be dangerous for tools. It is important to stress that XLIFF in its exact XML serialization has never been intended as a processing format, it is the *interchange* bitext, it advances through distributed workflow steps, but does not prescribe the internal processing representation

(serialization) in each and every tool of the ecosystem, as long as the semantic interoperability has been preserved and a valid XLIFF exists at input and output of each *processing* step.
The XLIFF 2 specifications are largely written as for an abstract data model but the instantiation specifics are only given for one particular XML serialization that is the XLIFF format itself. But the XLIFF format still contains industry wisdom that can be abstracted into a generalized Localisation Interchange Object Model that will guarantee interoperability between arbitrary serializations based on the same Object Model.
The root element of an XLIFF file is `<xliff>` and the registered extension for the XLIFF media type `xliff+xml` is `xlf`. The root element holds at least one `<file>` element that can optionally hold a recursive structure of `<group>` elements. Logical units of content are held in `<unit>` elements that can be children of `<group>` or `<file>` elements. If you are not an XML partisan you may wonder why bother with the angle brackets and you are right. To give the abstract overview of the XLIFF data model, we don't need to rely on the XML specifics of its serialization. In fact, the `<xliff>` element is a top level or root element that allows to declare what type of XML vocabulary is to be expected in the file at hand, namespaces, version and very few other things that have payload impact such as the source or target language or white space handling. `<file>` elements represent actual extracted files (or high level content nodes) and the files (or nodes) to which the targets will be merged. `<group>` elements do correspond to actual groupings to be preserved from the source to the target documents etc. So I can easily drop the angle brackets in the following and describe the XLIFF data structure in an apparently natural language.
The most interesting feature of XLIFF is its inline data model. Given by the extensive experience dealing with transformations from one natural language to another, localisation industry was and is in a unique position to define a native format agnostic data structure that is also capable to carry business critical metadata as annotations. From the theoretical point of view the most interesting predecessor of the XLIFF inline data model is the Text Encoding Initiative, which is to the present day a vigorous movement producing a widely adopted XML serialization that supports multiple approaches to the inline data model. In fact, TEI covers the whole logical space when it comes to encoding overlapping annotations in a natural language text. TEI is largely serialization centric but it in fact strives to solve data model issues when it comes to practical problems of text encoding. The TEI encoded text needs to carry – for the most varied scholarly purposes – different sets of metadata that have overlapping scope within natural language units and segments. TEI postulated a while ago that there are basically only three different serialization options: 1) duplicate the text and annotate each copy with well-formed annotations of one and only one data type. 2) define non-well-formed annotation methods using pseudo-spans delimited by empty (from the XML DOM point of view) markers. 3) use a standoff method. The third option however doesn't solve the whole issue unless it is employed together with 2) or an extraneous "offset" method that would be capable of selecting partially overlapping spans with a reasonable persistence (too persistent can be as bad or even worse than unstable or transient).
We are not going to explore all of this in detail. Suffice it to say that XLIFF 1.2 had used a less than ideal mixture of options 1) and 2); whereas XLIFF 2 is a full-fledged option 2) combined with option 3) implemented as standoff metadata containers referencing to or (less so) being referenced from well-formed as well as non-well-formed spans within the inline payload.
Interestingly XLIFF 2.0 is the first ever version of the XLIFF bitext format that defines a specific fragment identification mechanism that allows for external, internal and custom referencing, which among other things allows for a workable standoff solutions. Standoff methods are used mostly for XLIFF 2 modules data, but also for arbitrary extensions and core notes/commenting mechanism. Extension and module metadata containers are allowed on

three structural levels, the file level, the group level and the unit level. Modules and extensions that have data on the unit level can reference arbitrary spans of the inline payload within the same unit as the payload to which the metadata applies. To address cases where no suitable spans had been delimited, modules and extensions can extend the core inline annotation mechanism and use it for marking suitable spans within the unit in scope.
XLIFF inline content can contain plain text (PCDATA) and inline elements in any order, whereas well-formed pair codes (<pc>) and markers (<mrk>) have an exhaustively defined equivalence with non-well-formed pseudo-spans delimited by empty start (<sc/>) and end (<ec/>) code pairs or empty start (<sm/>) and end () marker pairs. Importantly, pseudo-spans can overlap not only other pseudo-spans but also a strictly defined subset of well-formed spans. As result, the XLIFF inline data model cannot be represented by a tree graph. The well-formed spans (other than the pair code and markers) that do not limit formation of overarching pseudo-spans are the segment () and ignorable (<ignorable>) spans that are used to represent segmentation within a logical unit of text. Moreover, orphaned native codes can be represented with orphaned XLIFF inline start or end codes, while orphaned start or end markers are not allowed. The rationale is that annotators (Enrichers) are in control of their own markup while Extractor might not be able to access corresponding opening or closing native markup for valid processing reasons.
Apart from the above described spanning methods, XLIFF has an empty placeholder (<ph/>) inline element and an empty code point (<cp/>) representation element for representing XML illegal Unicode characters.
The fact that XLIFF inline data model is not of a tree type, codes and annotations can survive the change of the DOM structure on the sub-unit level. Although segment and ignorable qualify as structural elements in the XLIFF 2 specifications, it is a fluid structure and CAT Tools (Modifiers) can and are expected to change this structure according to their segmentation rules or according to the human translator preferences. Thus finally the translator has the power to encode their bi-text rendition as an actual bitext artefact. In case the structure is changed, the original rendition can still be stored within the Change Tracking Module. Finally, XLIFF 2 inline data model brings a target order attribute that allows for thema-rema reordering of target content within logical units (paragraphs), a feature so important when moving between Germanic and Romance languages.

6.1 Why XLIFF 2?

This paper is supposed to be consumable by translator and business decision making audiences, hence we will omit a detailed comparison between XLIFF 1.2 and XLIFF 2 in this rationale. Why was it necessary to develop XLIFF 2.0?
Although XLIFF 1.2 is a widely adopted format and is the format behind several spectacular localisation interoperability success stories, it has – due to its old age – a number of inherent limitations that could not simply be fixed by another “dot release”. Making of XLIFF 2.0 signified that there was a substantially new version that strived to build on all the good old things but has to part from backwards compatibility due to some irredeemable issues of XLIFF 1.2. XLIFF TC indeed carefully studied the usage of XLIFF 1.2 and listened to implementers feedback in XLIFF Symposia held ever since 2010. XLIFF TC (Promotion and Liaison Subcommittee) produced two editions of the XLIFF 1.2 State of the art report; Morado Vázquez and Filip (2012), Filip and Morado Vázquez (2013) analysed in detail, which XLIFF 1.2 features were largely supported by implementers and which less so. (Filip and Wasala, 2013a) brought a frequency analysis of a large (albeit not demonstrably representative) corpus of XLIFF 1.2 files as currently used in the industry, Filip and Wasala (2013b) analysed those findings in greater detail and also brought one of the major advances in addressing conformance of XLIFF Agents that was later adopted in XLIFF 2. (Morado

Vázquez and Filip, 2014) are the first State of the Art report exploring the XLIFF 2 implementations.
In general, XLIFF 1.2 suffered from ailments common in many first generation standards. We can name a few based on Cargill's taxonomy of standardization failures (Cargill, 2011). For instance, overreliance on compromise is one of issues largely present in XLIFF 1.2, the result of which is the inline markup "salad" (courtesy of Rodolfo Raya) that allowed at least two different styles of inline markup without actually defining their equivalence or saying what to do in case one uses their style of choice and someone else another. So we ended up with implementers with very fierce arguments for using just their own subset of XLIFF 1.2 features. Interoperability Now! (see e.g. Andrä et al., 2011) was one other classic Cargill failure, an attempt of three toolmakers and one localization buyer to standardize an XLIFF 1.2 profile outside of standardization bodies. This particular profile attempt failed because it missed the market opportunity and failed to draw together a representative critical mass of stakeholders. In 2010/2011 IN! proclaimed that they don't want to wait for XLIFF 2.0 and will create (surprise, surprise) Interoperability Now! However, they ended up creating a pseudo-standardization body with a mailing list and bi-weekly meetings that in the end arrived at a release around the time when XLIFF 2.0 started settling into technical stability; late 2013, when subsequent XLIFF 2.0 public reviews took place (Comerford et al., 2013, 2014b). All IN! feature requests were in fact covered with XLIFF 2.0 except one feature covered in XLIFF 2.1 (via the ITS module that was before available as an extension).
The original XLIFF 1.x – of which only XLIFF 1.2 made it to an OASIS Standard back in 2008 – was intended as a fire and die format, a bitext intended for interchange between two agents, an Extractor/Merger and an Editor "beyond the wall".
XLIFF 2.0 can afford not to be backwards compatible because XLIFF is a transient interchange format, most of its usefulness expires when the target becomes stable throughout an arbitrary number of bitext transforms, including QA checks, engineering checks etc. After the target content is successfully merged and published in the native format but in the new target language and locale, the primary role of XLIFF has ended.
However, in many organizations XLIFF has a second life. Because it is an extremely metadata rich bitext format, XLIFF stores can have multiple business functions that go far beyond the primary role of the interchange facilitating bitext. This is however out of scope of this paper.
The single biggest issue of XLIFF 1.2 is that the core is too big, in fact everything in XLIFF 1.2 is core. This makes the whole standard unwieldy and let's implementers to decide what are the most important parts of the standard. Effectively, the only universally adopted characteristic of XLIFF 1.2 was the bitext aspect. Thus the standard was infallibly delivering on its promise to keep source and target aligned during industrial bitext transformations.

XLIFF 2 applied an acid test on what is core and what isn't:

> *The core of XLIFF 2.0 consists of the minimum set of XML elements and attributes required to (a) prepare a document that contains text extracted from one or more files for localization, (b) allow it to be completed with the translation of the extracted text, and (c) allow the generation of* Translated *versions of the original document.*
>
> *The XML namespace that corresponds to the core subset of XLIFF 2.0 is* `"urn:oasis:names:tc:xliff:document:2.0"`.

(Comerford, Filip et al., 2014)

The core namespace and the functionality covered by the core elements and attributes is about 20% or less of what had the XLIFF 1.2 provided. Not much, you might say, but in fact, this is

an extremely good news for interoperability. There is a 100% lossless interoperability promise guaranteed by the tight and non-negotiable core. At the same time XLIFF 2.0 defined eight (8) optional modules for advanced functionality in certain areas:
Translation Candidates Module allows for inclusion of relevant Translation Memory or Machine Translation suggestions within each unit. Glossary Module in turn is for inclusion of relevant Terminology subsets, or for sourcing new bilingual term glossaries. Format Style Module enables CAT Tools to create simple HTML previews of the content and metadata. The Resource Data Module replaces the bin-unit functionality of XLIFF 1.2 and allows for inclusion of external binary data either as context or localizable payload. Change Tracking Module allows for simple content tracking within units and notes. However in practice the usefulness is limited to simple comparison tasks, such as comparing raw MT with the post-edited version, storing a note or QA history for auditing purposes. This module cannot serve as a traditional full-fledged versioning (for that you can manage your XLIFF store on an XML database or a classic version control such as git) since it is optional and the core can be edited by tools that don't support this module. Usefulness of this module could be enhanced in controlled workflows where usage of the Change Tracking module could be mandated. Size and Length restriction module is an extensible framework for defining any sort of content limitations given by the target systems storage or display capability restrictions. It offers two standard profiles for the two most common use cases: restriction by storage size or number of code points of the target content. Validation Module is a simple mechanism to encode and enforce some common QA rules patterns. Metadata module is an extensibility mechanism without the need of using namespaces not defined in XLIFF. One cannot expect transparent machine to machine interoperability based on this module's data, but the general logic of this module allows to group different types of metadata and the data structure is one of key and value pairs. Therefore all agents supporting this module should be able to at least display the grouped key-value pairs. Since this is an extensibility mechanism, users of the module must observe the general extension functionality restriction of the XLIFF 2 architecture, i.e. that extensions must not implement any features that are available through core or other modules.
In XLIFF 2.1 a large new module was added, the ITS Module. The number of features added and the general expressivity added through this module can be compared to at least six modules in XLIFF 2.0. The module fully defines all elements and attributes to natively support six (6) ITS (Filip et al., 2013) data categories: Allowed Characters (restricting character sets for instance for legacy Unicode incapable systems), Domain (for conveying of topic or specialization of content, useful for instance for selecting of a domain trained MT engine or subject matter expert translators), Locale Filter (for conditional profiling of content for various locales in monolingual XLIFF files, i.e. before the targets in one of the relevant locales are added and in fact informing the workflow or process to do so). Localization Quality Issue and Rating are two categories that serve for injecting of QA metadata during the process of performing QA. Notably, MQM originates from the list of issue types in the Localization Quality Issue data type and is in effect the only machine readable MQM profile as of now. Text Analysis data category lets enrichers include data from disambiguation and entity recognition services.
Four (4) ITS metadata categories (Localization Note, Preserve Space, Translate and External Resource) were fully expressible in XLIFF 2.0, therefore the XLIFF 2.1 ITS module only informatively describes how these categories correspond to native XLIFF features.
Five (5) ITS metadata categories have partial overlap with XLIFF 2.0 features. For Language Information, MT Confidence, Provenance and Terminology, the XLIFF 2.1 ITS module defines how to extend the existing XLIFF 2.0 features with ITS module defined elements and attributes to fully express them. ITS Storage Size data category would be fully expressible within the Size and Length Restriction Module framework as a third party profile, however

the profile itself has not been specified at this point due to lack of current industry interest in expressing this profile.
Five (5) ITS data categories are not expressed as metadata when extracting content into XLIFF, instead the category data is fully "consumed" by the extraction behaviour. The categories are: Directionality (the directionality provisions of ITS 2.0 were obsoleted by recent changes in Unicode and HTML5 that are supported by the XLIFF 2 data model), Elements within Text, ID Value, Locale Filter (in the standard use case this category is not needed as metadata in a bilingual XLIFF file) and Target Pointer (there is a simple rule that points ITS Processors to XLIFF target content).

7 Conclusion

We have seen how bi-text and bitext are critical in producing fit for purpose translations in varied contexts. After looking in more detail at industrial localisation, we concluded that interoperable bitext representations are necessary for effective localisation operation at translation buyer organizations above a certain size and maturity level and most language service providers and freelancers. Existence of an open standard bitext format allows sound competition in the marketplace and helps translation buyers to prevent vendor lock-in. Technology agnostic vendors and freelancers benefit as they are not forced to maintain parallel tool stacks. We have reviewed at a conceptual level the XLIFF data model and shown that it is a fully expressive and adequate vehicle for interoperable bitext representation in the ever growing translation and localisation market.

Acknowledgements

I wish to thank Alan Melby and Olaf-Michael Stefanov for introducing me to the JIAMCATT network and making me work on the deep reasoning for "Why bitext?" and "Why open standard bitext?".
This research was supported by the Theme E of the SFI ADAPT Centre at the Trinity College Dublin (SFI Research Centres Programme Grant 13/RC/2106).
BPMN 2.0 compliant model exports were created with Bizagi Modeler 64-bit Version 3.1.0.011. Thanks to Bizagi for developing their free Process Modeler.

References

Andrä, S.C., Coffey, D., Filip, D., Reynolds, P., Ugray, G., 2011. Interoperability Now! Presented at the MemoQ Fest, Kilgray, Budapest.

Best, K.F., 2001. [tc-announce] OASIS TC Call For Participation: XML LocalisationInterchange File Format TC.

Bray, T., Paoli, J., Sperberg-McQueen, C.M., Maler, E., Yergeau, F. (Eds.), 2008. Extensible Markup Language (XML) 1.0 (Fifth Edition), Fifth. ed, W3C Recommendation. W3C.

Cargill, C., 2011. Why Standardization Efforts Fail. J. Electron. Publ. 14. doi:http://dx.doi.org/10.3998/3336451.0014.103

Comerford, T., Filip, D., Raya, R.M., Savourel, Y. (Eds.), 2014a. XLIFF Version 2.0, OASIS Standard. ed, Standard. OASIS.

Comerford, T., Filip, D., Raya, R.M., Savourel, Y. (Eds.), 2014b. XLIFF Version 2.0 [csprd03], Committee Specification Draft 03 / Public Review Draft 03. ed, Standard. OASIS.

Comerford, T., Filip, D., Raya, R.M., Savourel, Y. (Eds.), 2013. XLIFF Version 2.0 [csprd02], Committee Specification Draft 02 / Public Review Draft 02. ed, Standard. OASIS.

Cover, R., 2016. GitHub Repositories for OASIS TC Members' Chartered Work [WWW Document]. OASIS. URL https://www.oasis-open.org/resources/tcadmin/github-repositories-for-oasis-tc-members-chartered-work (accessed 9.29.16).

DePalma, D.A., Beninatto, R.S., Sargent, B.B., 2006. Localization Maturity Model (paid research No. Release 1.0). Common Sense Advisory, Inc., Lowell, MA.
Filip, D., 2015. [xliff-comment] Draft Meeting Minutes from FEISGILTT 2015 Infosessions (Meeting mInutes of Record). OASIS XLIFF TC, Berlin.
Filip, D., 2012. Using Business Process Management and Modelling to Analyse the Role of Human Translators and Reviewers in Bitext Management Workflows, in: IATIS 2012. Presented at the IATIS 2012, Belfast.
Filip, D., Comerford, T., Saadatfar, S., Sasaki, F., Savourel, Y. (Eds.), 2016. XLIFF Version 2.1, Committee Specification Draft 01 / Public Review Draft 01. ed, Standard. OASIS.
Filip, D., McCance, S., Lewis, D., Lieske, C., Lommel, A., Kosek, J., Sasaki, F., Savourel, Y. (Eds.), 2013. Internationalization Tag Set (ITS) Version 2.0, Recommendation. ed, Recommendation. W3C.
Filip, D., Morado Vázquez, L., 2013. XLIFF Support in CAT Tools (Subcomittee Report No. 2), XLIFF State of the Art. OASIS XLIFF TC.
Filip, D., Ó Conchúir, E., 2011. An Argument for Business Process Management in Localisation. Localis. Focus 10, 4–17.
Filip, D., Wasala, A., 2013a. Addressing Interoperability Issues of XLIFF: Towards the Definition and Classification of Agents and Processes. Presented at the FEISGILTT 2013, Localization World, London.
Filip, D., Wasala, A., 2013b. Process and Agent Classification Based Interoperability in the Emerging XLIFF 2.0 Standard. Localis. Focus, Special Standards Issue I 12, 61–73.
Harris, B., 1988. Bi-text, A New Concept in Translation Theory. Lang. Mon. 54, 8–10.
King, R., 2015. Microsoft/XLIFF2-Object-Model. Microsoft.
Krechmer, K., 2006. Open Standards Requirements. Int. J. IT Stand. Stand. Res. IJITSR 4, 43–61. doi:10.4018/jitsr.2006010103
Melby, A., Lommel, A., Vázquez, L., 2015. Bitext, in: Routledge Encyclopedia of Translation Technology. Routledge, pp. 409–424.
Microsoft News, 2015. Microsoft XLIFF 2.0 Object Model is now available on GitHub [WWW Document]. Microsoft News Cent. Blog. URL https://blogs.microsoft.com/firehose/2015/11/11/microsoft-xliff-2-0-object-model-is-now-available-on-github/ (accessed 10.16.16).
Morado Vázquez, L., Filip, D., 2014. XLIFF 2.0 Support in CAT Tools (Subcomittee Report No. 3), XLIFF State of the Art. OASIS XLIFF TC.
Morado Vázquez, L., Filip, D., 2012. XLIFF Support in CAT Tools [1st ed.] (Subcomittee Report No. 1), XLIFF State of the Art. OASIS XLIFF TC.
OASIS Board of Directors, 2010. Intellectual Property Rights (IPR) Policy | OASIS.
Open Source Initiative, 2012. Open Standards Requirement for Software [WWW Document]. Open Source Initiat. URL https://opensource.org/osr (accessed 9.29.16).
Petersen, P., 2015. XLIFF 2.0 Object Model is now Open Source on GitHub [WWW Document]. Microsoft Lang. Portal Blog. URL https://blogs.technet.microsoft.com/terminology/2015/11/11/xliff-2-0-object-model-is-now-open-source-on-github/ (accessed 10.16.16).
Rosen, L., 2004. Open Source Licensing Software Freedom and Intellectual Property Law [WWW Document]. URL http://www.rosenlaw.com/oslbook.htm (accessed 5.29.12).
Saadatfar, S., Filip, D., 2016. Best Practice for DSDL-based Validation, in: XML London 2016 Conference Proceedings. Presented at the XML London 2016, XML London, London.
Saadatfar, S., Filip, D., 2015. Advanced Validation Techniques for XLIFF 2. Localis. Focus, Special Standards Issue II 14, 43–50.
Savourel, Y., Reid, J., Jewtushenko, T., Raya, R.M. (Eds.), 2008. XLIFF Version 1.2, OASIS Standard. ed, Standard. OASIS.
XLIFF TC, 2014. OASIS XML Localisation Interchange File Format Technical Committee [Charter].

Can you trust a TM? Results of an experiment conducted in November 2015 and August 2016 with students and professional translators.

Daniela Ford
Centre for Translation Studies (CenTraS)
University College London
Gower Street
London
WC1E 6BT
UK
daniela.ford@ucl.ac.uk

Abstract

This paper describes an experiment conducted by the author in November 2015 with 69 MSc Translation students at CenTraS @ UCL (covering 14 target languages) and in August 2016 with 30 professional translators in Saudi Arabia (covering English to Arabic). The experiment was inspired by Lynne Bowker's pilot study **Productivity vs Quality? A pilot study on the impact of translation memory systems** (published in *Localisation Focus* in March 2005). The author of this paper wanted to find out whether translators who are fairly new to translation technology would "blindly" trust the content of a TM or whether they would still check the content thoroughly and make any necessary changes to the translation. Students and professional translators were asked to translate a short text consisting of 14 sentences and a total of 217 words in Wordfast Anywhere/SDL Trados Studio 2015. They also received a translation memory (TM) for their respective language combination. All TMs contained mistakes, which the author did not mention to the students and the professional translators. Interestingly, while the professional translators fared better at editing fuzzy matches than the students, they did not pick up on incorrect 100% matches as well as the student translators, tended to lack attention to detail by, for example, introducing double spaces into sentences, and not all professional translators translated the new sentences given for translation.

1 Introduction

Ever since the author came across Lynne Bowker's pilot study **Productivity vs Quality? A pilot study on the impact of translation memory systems** (published in *Localisation Focus* in March 2005), she wanted to conduct a similar experiment with her own students. This finally happened in November 2015, at the Centre for Translation Studies (CenTraS) at University College London (UCL).

Bowker had conducted an experiment with students of French and English in order to investigate the impact of translation memory (TM) tools on both the speed and the quality of the translation. She had divided her students into three groups and asked them to translate the same text. The first group was asked to translate the text without the use of a translation memory tool, while the second and third groups were asked to use a translation memory tool, together with a translation memory which Bowker had provided. The translation memory for the second group was of good quality whilst the translation memory for the third group contained mistakes which Bowker had not told the students about.

Bowker's first group translated the text relatively well but was slower than the second and the third groups, and while both the second and the third groups translated the text more quickly, the third group did not pick up on all the mistakes contained in the translation

Proceedings of the 38th Conference Translating and the Computer, pages 69–80,
London, UK, November 17-18, 2016.

memory, thereby producing a translation of a lower quality. Bowker's conclusion was that translators who use a translation memory tool may therefore not be critical enough of translations suggested by the translation memory, which in turn also means that proper training is required for using translation technology.

The author of this paper wanted to replicate the experiment which Bowker conducted with her third group, however, this time with a total of 14 target languages, with 69 MSc Translation students at CenTraS @ UCL, as well as 30 professional translators. The student translators had completed 12 contact hours (2 hours of face to face teaching per week in a lab at UCL over 6 weeks) by the time the experiment was conducted. By this time, students had been taught about the concept of a translation memory (4 contact hours) and had learned to use Wordfast Anywhere (WFA) (8 contact hours). The professional translators had completed a one day training course (7 hours) on the concept of a translation memory tool and on how to translate using SDL Trados Studio 2015.

The author's aim of the experiment was to find out whether relatively new users of translation memory tools would "blindly" trust the content of a translation memory and whether there were distinct differences in terms of thoroughness between students and professional translators and possibly also between different languages/nationalities.

2 Set up of the experiment

The chosen source language for the text to be translated as part of the experiment was English, and translations were to be provided into the mother tongue of the sample groups. Target languages covered by the students were Italian (5 students), Simplified Chinese (31 students), Traditional Chinese (2 students), Russian (3 students), Swedish (1 student), Japanese (3 students), Portuguese (1 student), Greek (1 student), German (3 students), French (2 students), Spanish (4 students), Norwegian (1 student), Polish (2 students), Arabic (1 student but she did not submit), and Korean (1 student). Native English students (of which there were very few; 2 for German and 1 for Spanish) were asked to translate out of English for this experiment. The student numbers above are for those who actually submitted their translation. The age range of the students was roughly between 22 and 40 years, with some students having just completed a Bachelor's degree and some other students having worked for a number of years already, either as translators or in another profession.

Before the experiment, the author had asked the students to fill in a short questionnaire, which 44 out of the 69 students did. 12 students indicated that they had prior knowledge of translation memory tools, ranging from 2 days to 10 years. Tools mentioned were Trados in first place, followed by memoQ, Wordfast, Deja Vu, and OmegaT. The majority of the students who indicated prior knowledge originated from a European country (Italy, Sweden, Germany, UK, Spain, Norway, Portugal). Only 2 students from mainland China indicated that they had used a translation memory tool before (in both cases Trados, for only a couple of days). The other students had not heard of/not used a TM tool before coming to UCL.

The same experiment was then conducted with 30 professional translators (20 men and 10 women) in Saudi Arabia who had only learned to use a translation memory the day before the experiment was conducted.

The professional translators worked in pairs which resulted in a total of 15 translations for the author to analyse. The age range of the professional translators was roughly between 22 and 55 years, with the youngest professionals having just completed university and the oldest ones having worked as translators for up to 30 years already, however not with translation memory tools.

Students and professional translators were given a short text consisting of 14 sentences and a total of 217 words about the difference between Office 365 subscription plans and Office as a one-time purchase (which the author had copied from a website into a Word document), as

well as a translation memory (TM) for their respective language combination (which the author had previously created from the original text as well as the existing translation on a website and then prepared for the experiment, see 2.1).

For the experiment, student translators were asked to translate the short text together with the respective TM for their language combination using Wordfast Anywhere (WFA) in class. Detailed instructions were given out to the students on how to set up WFA for the experiment.

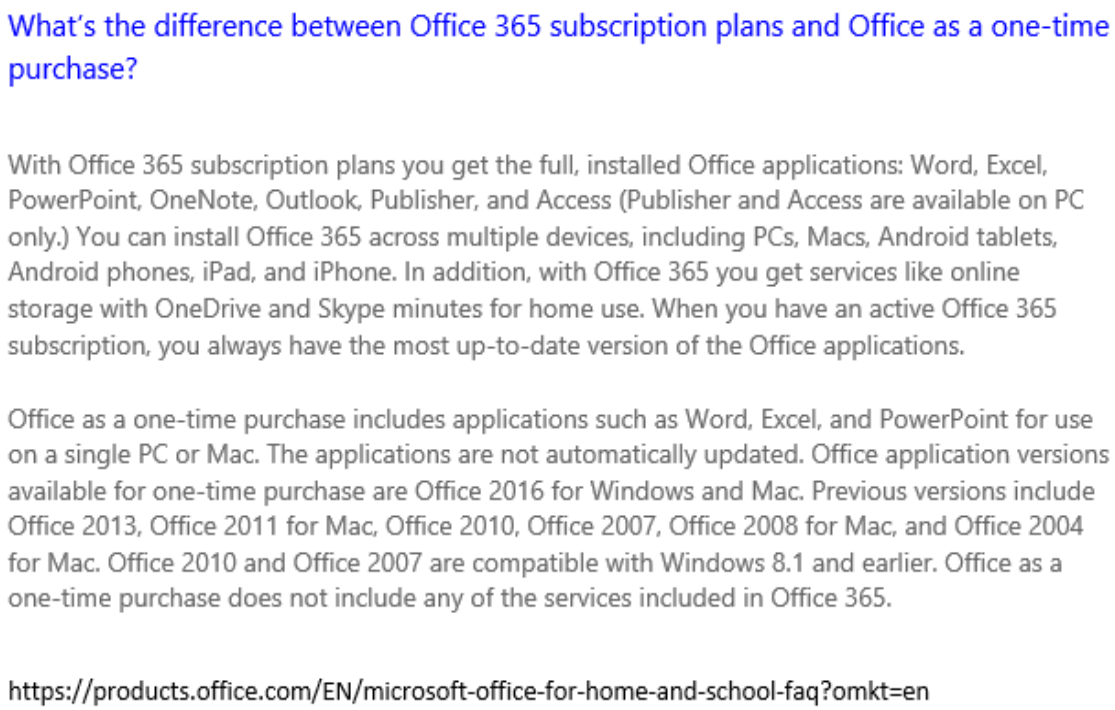
A similar approach was used for the professional translators, however, they received a project package for translation in SDL Trados Studio 2015 and were told that this was a revision exercise, rather than an experiment.

Students translated the text using WFA and submitted their updated TM as well as their bilingual file on Moodle, the virtual learning environment used at UCL. Professional translators opened the project package in SDL Trados Studio 2015, translated the file in SDL Trados Studio 2015 and then created a return package which they saved on their desktops. The author then collected the files from each desktop.

Neither the students nor the professional translators were given a time frame in which they had to complete the translation.

2.1 How the text for translation was prepared

The author decided to use a source text from the Microsoft website since it was possible to obtain translations of the source text into all the 14 languages required for the experiment. The source text is shown below, copied into a Word document.



What's the difference between Office 365 subscription plans and Office as a one-time purchase?

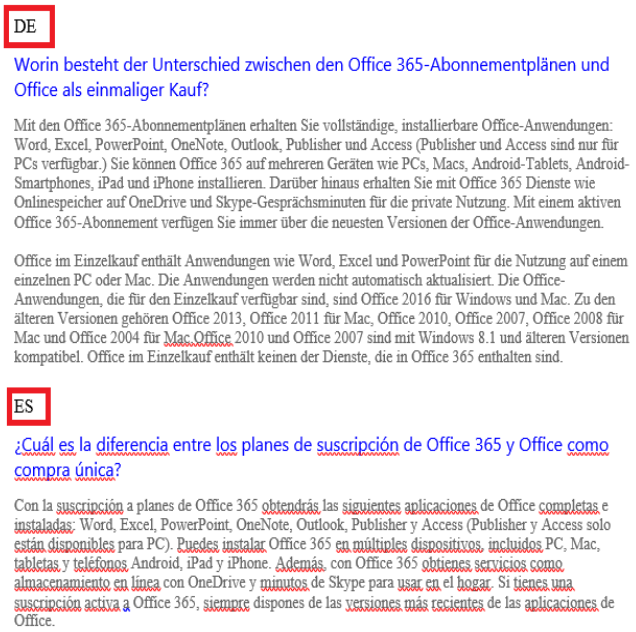
With Office 365 subscription plans you get the full, installed Office applications: Word, Excel, PowerPoint, OneNote, Outlook, Publisher, and Access (Publisher and Access are available on PC only.) You can install Office 365 across multiple devices, including PCs, Macs, Android tablets, Android phones, iPad, and iPhone. In addition, with Office 365 you get services like online storage with OneDrive and Skype minutes for home use. When you have an active Office 365 subscription, you always have the most up-to-date version of the Office applications.

Office as a one-time purchase includes applications such as Word, Excel, and PowerPoint for use on a single PC or Mac. The applications are not automatically updated. Office application versions available for one-time purchase are Office 2016 for Windows and Mac. Previous versions include Office 2013, Office 2011 for Mac, Office 2010, Office 2007, Office 2008 for Mac, and Office 2004 for Mac. Office 2010 and Office 2007 are compatible with Windows 8.1 and earlier. Office as a one-time purchase does not include any of the services included in Office 365.

https://products.office.com/EN/microsoft-office-for-home-and-school-faq?omkt=en

Figure 1. The source text, as copied from the Microsoft website into a Word document.

The author then copied all the required translations of this text from the Microsoft website into a second Word document.



DE

Worin besteht der Unterschied zwischen den Office 365-Abonnementplänen und Office als einmaliger Kauf?

Mit den Office 365-Abonnementplänen erhalten Sie vollständige, installierbare Office-Anwendungen: Word, Excel, PowerPoint, OneNote, Outlook, Publisher und Access (Publisher und Access sind nur für PCs verfügbar.) Sie können Office 365 auf mehreren Geräten wie PCs, Macs, Android-Tablets, Android-Smartphones, iPad und iPhone installieren. Darüber hinaus erhalten Sie mit Office 365 Dienste wie Onlinespeicher auf OneDrive und Skype-Gesprächsminuten für die private Nutzung. Mit einem aktiven Office 365-Abonnement verfügen Sie immer über die neuesten Versionen der Office-Anwendungen.

Office im Einzelkauf enthält Anwendungen wie Word, Excel und PowerPoint für die Nutzung auf einem einzelnen PC oder Mac. Die Anwendungen werden nicht automatisch aktualisiert. Die Office-Anwendungen, die für den Einzelkauf verfügbar sind, sind Office 2016 für Windows und Mac. Zu den älteren Versionen gehören Office 2013, Office 2011 für Mac, Office 2010, Office 2007, Office 2008 für Mac und Office 2004 für Mac Office 2010 und Office 2007 sind mit Windows 8.1 und älteren Versionen kompatibel. Office im Einzelkauf enthält keinen der Dienste, die in Office 365 enthalten sind.

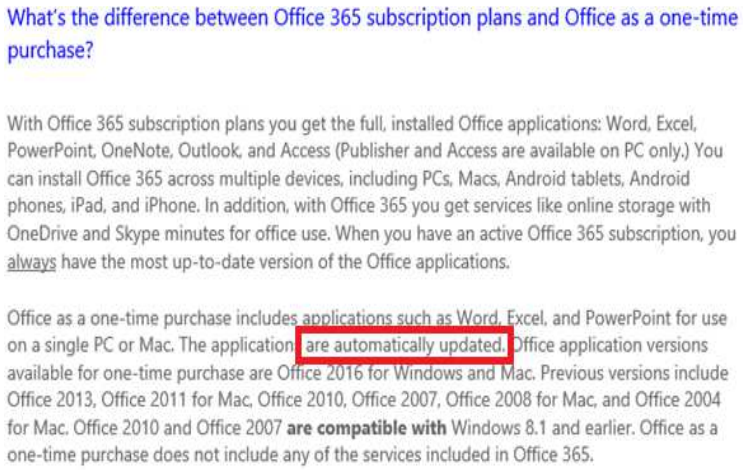
ES

¿Cuál es la diferencia entre los planes de suscripción de Office 365 y Office como compra única?

Con la suscripción a planes de Office 365 obtendrás las siguientes aplicaciones de Office completas e instaladas: Word, Excel, PowerPoint, OneNote, Outlook, Publisher y Access (Publisher y Access solo están disponibles para PC). Puedes instalar Office 365 en múltiples dispositivos, incluidos PC, Mac, tabletas y teléfonos Android, iPad y iPhone. Además, con Office 365 obtienes servicios como almacenamiento en línea con OneDrive y minutos de Skype para usar en el hogar. Si tienes una suscripción activa a Office 365, siempre dispones de las versiones más recientes de las aplicaciones de Office.

Figure 2. All translations required for the experiment were copied from the Microsoft website into a second Word document.

As the author wanted to introduce mistakes in the translation memory without making these mistakes obvious in each of the target languages, the author then modified the original English text (rather than modifying each of the 14 translations) by changing some formatting, adding and deleting words as well as sentences in the English source text. In the last paragraph, for example, “are not automatically updated” was changed to “are automatically updated” in the source text. A number of further small changes were introduced, with an attempt to prevent them being obvious.



What's the difference between Office 365 subscription plans and Office as a one-time purchase?

With Office 365 subscription plans you get the full, installed Office applications: Word, Excel, PowerPoint, OneNote, Outlook, and Access (Publisher and Access are available on PC only.) You can install Office 365 across multiple devices, including PCs, Macs, Android tablets, Android phones, iPad, and iPhone. In addition, with Office 365 you get services like online storage with OneDrive and Skype minutes for office use. When you have an active Office 365 subscription, you always have the most up-to-date version of the Office applications.

Office as a one-time purchase includes applications such as Word, Excel, and PowerPoint for use on a single PC or Mac. The applications are automatically updated. Office application versions available for one-time purchase are Office 2016 for Windows and Mac. Previous versions include Office 2013, Office 2011 for Mac, Office 2010, Office 2007, Office 2008 for Mac, and Office 2004 for Mac. Office 2010 and Office 2007 **are compatible with** Windows 8.1 and earlier. Office as a one-time purchase does not include any of the services included in Office 365.

Figure 3. The modified English source text. Example: In the last paragraph, line 2, “are not automatically updated” was changed to “are automatically updated”, however, the translations still read “are not automatically updated”.

The author then created 14 translation memories, using the modified English source text and the unchanged translation as taken from the Microsoft website which had been copied into a Word document. The purpose of this was to create “false” 100% matches, as shown in the

example below. Each translation memory was then exported as a tmx (translation memory exchange) file.

```
<tu creationdate="20151120T141818Z" creationid="GJUGuV" usagecount="0">
  <prop type="x-attribute1">1=A</prop>
  <tuv xml:lang="EN-US">
    <seg>The applications are automatically updated.</seg>
  </tuv>
  <tuv xml:lang="DE-DE">
    <seg>Die Anwendungen werden nicht automatisch aktualisiert.</seg>
  </tuv>
</tu>

<tu creationdate="20151120T141731Z" creationid="GJUGuV" usagecount="0">
  <prop type="x-attribute1">1=A</prop>
  <tuv xml:lang="EN-US">
    <seg>You can install Office 365 across multiple devices, including PCs, Macs, Android tablets, Android phones, iPad, and iPhone.</seg>
  </tuv>
  <tuv xml:lang="DE-DE">
    <seg>Sie können Office 365 auf mehreren Geräten wie PCs, Macs, Android-Tablets, Android-Smartphones, iPad und iPhone installieren.</seg>
  </tuv>
</tu>

<tu creationdate="20151120T141709Z" creationid="GJUGuV" usagecount="0">
  <prop type="x-attribute1">1=A</prop>
  <tuv xml:lang="EN-US">
    <seg>Word, Excel, PowerPoint, OneNote, Outlook, and Access (Publisher and Access are available on PC only.)</seg>
  </tuv>
  <tuv xml:lang="DE-DE">
    <seg>Word, Excel, PowerPoint, OneNote, Outlook, Publisher und Access (Publisher und Access sind nur für PCs verfügbar.)</seg>
  </tuv>
</tu>
```

Figure 4. tmx for EN-US to DE-DE which includes “false” 100% matches.

In the final step, the author made the following changes to the source text (shown below using track changes for illustrative purposes only). This file was then given to the students/professional translators for translation. The aim was to create 100% matches, fuzzy matches and no matches which the students and professional translators would have to work on.

The students downloaded the Word document for translation and their respective tmx file from Moodle, while the professional translators received the text for translation and the translation memory as a project package for SDL Trados Studio 2015.

What's tThe difference between Office 365 subscription plans and Office as a one-time purchase?

With Office 365 subscription plans you get the full, installed Office applications: Word, Excel, PowerPoint, OneNote, Outlook, and Access (Publisher and Access are available on PC only.) You can install Office 365 across multiple devices, including PCs, Macs, and Android tablets, Android phones, iPad, and iPhone. In addition, with Office 365 you get services like online storage with OneDrive and Skype minutes for office use. When you have an active Office 365 subscription, you always have the most up-to-date version of the Office applications.

Office applications are tailored to work best on each platform and device. The Office applications available for Mac users and the version numbers may be different from those available for PC users.

Office as a one-time purchase includes applications such as Word, Excel, and PowerPoint for use on a single PC or Mac. The applications are automatically updated. Office application versions available for one-time purchase are Office 2016 for Windows and Mac. Previous versions include Office 2013, Office 2011 for Mac, Office 2010, Office 2007, Office 2008 for Mac, and Office 2004 for Mac. Office 2010 and Office 2007 **are compatible with** Windows 8.1 and earlier. Office as a one-time purchase does not include any of the services included in Office 365.

Figure 5. The file for translation, shown with track changes. All track changes were accepted before the file was given to the students and the professional translators.

2.2 What the students and professional translators were expected to do with the text for translation

The following table shows the 14 sentences as well as what the author expected the students and professional translators to do with them. The actual word count for new translations was very low (only sentences 7 and 8 had to be translated from scratch); all other sentences were either correct or incorrect 100% matches or fuzzy matches which had to be edited. This table was obviously not provided to the test subjects.

#	The text as given to student/professional translators	What the student/professional translators should have done when they translated the text:
1	The difference between Office 365 subscription plans and Office as a one-time purchase	Fuzzy match: Should have changed the translation from the TM (which was a question) to match the sentence shown on the left (i.e. delete the word "What's" as well as the question mark at the end).
2	With Office 365 subscription plans you get the full, installed Office applications:	100% match: Should have kept what the TM provided.
3	Word, Excel, PowerPoint, OneNote, Outlook, and Access (Publisher and Access are available on PC only.)	100% match from the TM but I had inserted a mistake: Should have deleted the word "Publisher" from the translation: the translation should have read ...**Outlook, and Access** INSTEAD of **Outlook, Publisher, and Access**.
4	You can install Office 365 across multiple devices, including PCs, and Android tablets, Android phones, iPad, and iPhone.	Fuzzy match: Should have changed the translation from the TM and deleted "Macs" and added the word "and": the translation should have read **...including PCs, and Android tablets** INSTEAD of **...including PCs, Macs, and Android tablets**.
5	In addition, with Office 365 you get services like online storage with OneDrive and Skype minutes for office use.	Incorrect 100% match from the TM: Should have changed "home use" to "office use" in the translation.
6	When you have an active Office 365 subscription, you always have the most up-to-date version of the Office applications.	Incorrect 100% match from the TM: Should have underlined the word for "always" in the translation.
7	Office applications are tailored to work best on each platform and device.	This was a new sentence for translation which had to be translated from scratch.
8	The Office applications available for Mac users and the version numbers may be different from those available for PC users.	This was a new sentence for translation which had to be translated from scratch.
9	Office as a one-time purchase includes applications such as Word, Excel, and PowerPoint for use on a single PC or Mac.	100% match: Should have kept what the TM provided.
10	The applications are automatically updated.	Incorrect 100% match from the TM: Should have deleted the word "not" (**are not automatically updated -> are automatically updated**) in the translation.
11	Office application versions available for one-time purchase are Office, for Windows and Mac.	Fuzzy match: Should have deleted "2016" (Office 2016 -> Office) in the translation.
12	Previous versions include Office 2013, Office 2011 for Mac, Office 2010, Office 2007, Office 2008 for Mac, and Office 2004 for Mac.	100% match: Should have kept what the TM provided.
13	Office 2010 and Office 2007 **are compatible with** Windows 8.1 and earlier.	Incorrect 100% match from the TM: Should have made "are compatible with" bold in the translation.
14	Office as a one-time purchase does not include any of the services included in Office 365.	100% match: Should have kept what the TM provided.

Figure 6. The file for translation, and what students and professional translators should have done.

3 Evaluation of the results

3.1 Submitted translations and time spent

Of the 69 student translators who completed the experiment in class, not all students submitted their files and some students submitted wrong files. Two students were from Taiwan and had worked with the tmx for Mainland China thereby mixing simplified with traditional Chinese characters (they had not mentioned at the start of the course that they were from Taiwan which is why the author had not provided a TM with Traditional Chinese for this experiment). All in all, the valid sample for analysis consisted of 60 student translations.

All professional translators (English into Arabic) submitted the correct file, however, this was as expected, as they had no other files which they could have submitted, and the author collected the files directly from their desktops. In terms of time, both students and professional translators spent between 20 and 50 minutes to complete the experiment. It was

also interesting to see how some student translators revisited/checked their work once they had completed the files, something which the professional translators did not do.

In order to complete the experiment, the students had to download the file for translation and the tmx from Moodle, log on to WFA, create an empty TM and import the tmx file into this newly created TM, open the file for translation in WFA, translate it together with the TM, download both the translated bilingual file as well as the tmx from WFA and submit it on Moodle. The professional translators only had to open the project package in SDL Trados Studio 2015, translate the file and create a return package which they saved on their desktops.

3.2 Evaluation of the results: 100% matches which should not have been changed

The text contained 4 sentences (# 2, 9, 12 and 14) for which correct 100% matches had been provided in the TM, and these sentences should have therefore been kept unchanged.

Of the student translators, 72% kept the existing translation and 28% changed it. Those who changed the translation generally improved it, e.g. by using a better wording.

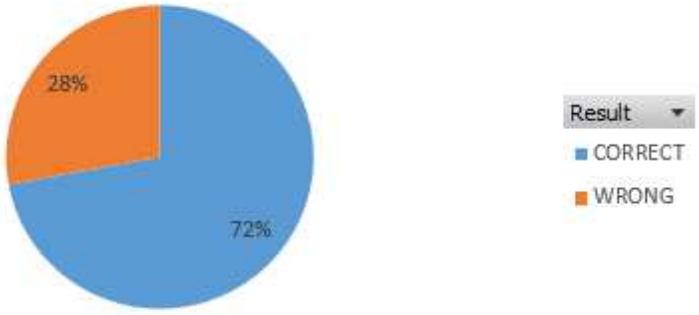


Figure 7. 72% of the student translators left sentences 2, 9, 12 and 14 unchanged.

An analysis of the largest student group (Simplified Chinese: 31) reveals the following:

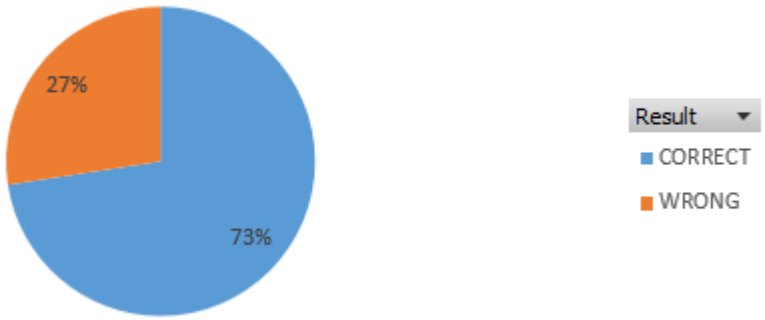


Figure 8. 73% of the Simplified Chinese student translators left sentences 2, 9, 12 and 14 unchanged.

Although the sample group for the Germanic languages (German, Norwegian, Swedish) was small, the result was as follows:

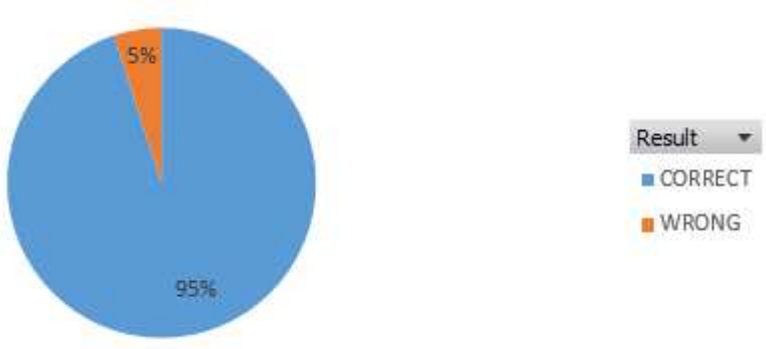


Figure 9. 95% of the Germanic student translators left sentences 2, 9, 12 and 14 unchanged.

The result for the Romance languages (French, Italian, Spanish, Portuguese) was as follows:

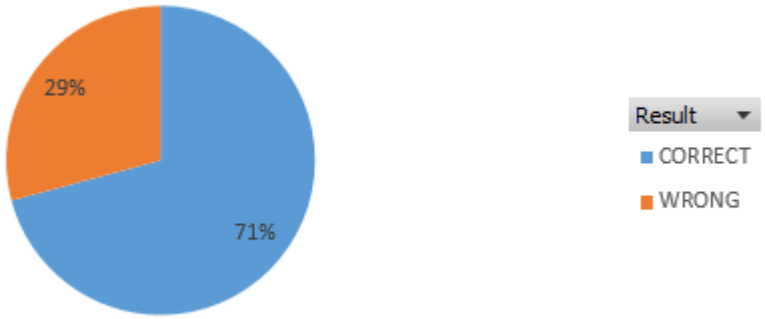


Figure 10. 71% of the Romance student translators left sentences 2, 9, 12 and 14 unchanged.

Of the professional translators, 97% left sentences 2, 9, 12 and 14 unchanged; the 3% who changed the translation said that the original translation was not good.

Figure 11. 97% of the professional translators left sentences 2, 9, 12 and 14 unchanged.

There are a number of possible interpretations for this result: The student translators may have been more thorough or more inexperienced and therefore changed a perfectly good translation. The professional translators were from Saudi Arabia and the Arabic provided to them was in a different Arabic dialect, or the professional translators simply did not consider it important to really look at the translation critically. A comment which the author kept hearing from the professionals was: “We are only in training and you will delete the files

anyway so it doesn't matter whether we do a good job or not – this training is only about the process of learning how to work with the tool, not about producing a nice translation".

3.3 Evaluation of the results: Sentences which had to be translated from scratch

For sentences 7 and 8 (shown below as segments 8 and 9), no translation was provided so these sentences had to be translated from scratch. Interestingly, all student translators translated these two sentences but of the 15 professional translator teams, 3 teams (all men) left these sentences untranslated.

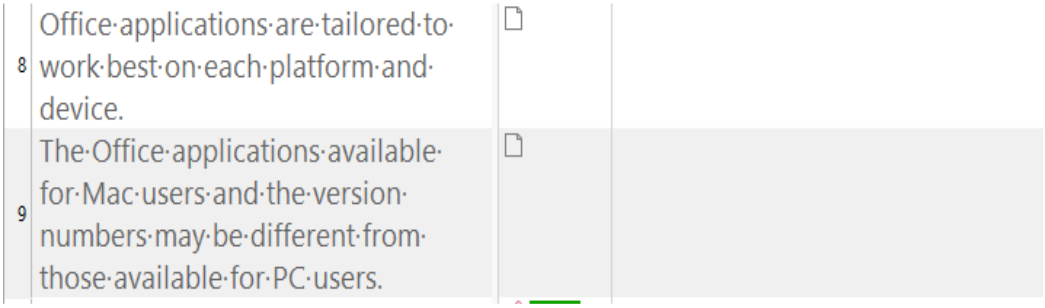


Figure 12. New segments left empty.

3.4 Evaluation of the results: Sentences which were 100% matches but contained mistakes

Sentences 3, 5, 6, 10 and 13 were provided as 100% matches but actually contained mistakes which had been introduced by the author. The author wanted to find out whether the students/professional translators checked 100% matches carefully or whether they "blindly" trusted the TM and accepted these matches with the mistakes.

60% of the student translators spotted the mistakes in these 100% matches and corrected them, whereas 40% "blindly" accepted the incorrect 100% matches.

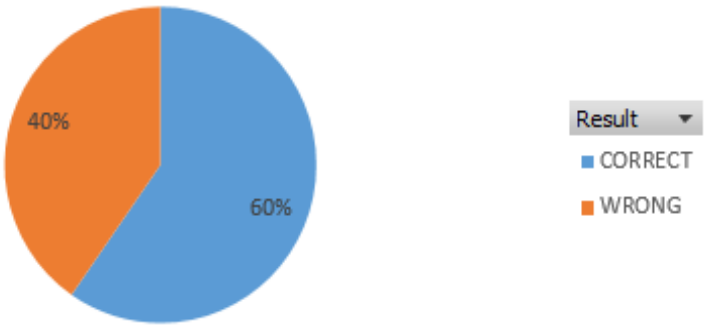


Figure 13. 60% of the student translators spotted the mistakes in the incorrect 100% matches.

As for the professional translators, the result is quite staggering and the opposite to the students' result: Only 40% spotted the mistakes in the incorrect 100% matches.

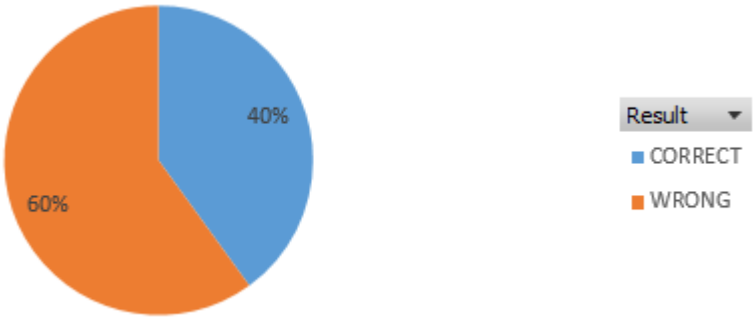


Figure 14. Only 40% of the professional translators spotted the mistakes in the incorrect 100% matches.

This is quite a surprising result, and again, several interpretations are possible. Upon querying this with some of the professional translators, the author was told that "we spotted the difference but didn't change it". Whether this is true or simply a way of saving face and not admitting that this was missed is not clear.

3.5 Evaluation of the results: Fuzzy matches which should have been edited

Sentences 1, 4 and 11 were fuzzy matches which should have been edited. The author had deliberately introduced very small changes in these sentences so as to not make the changes too obvious.

The result of the student translators was interesting and maybe the most surprising result as only 53% of the students spotted the differences (which were shown in the Translation Memory window and should therefore have been obvious) and changed the translation whereas 47% did not change the translation and thereby ended up with a wrong translation.

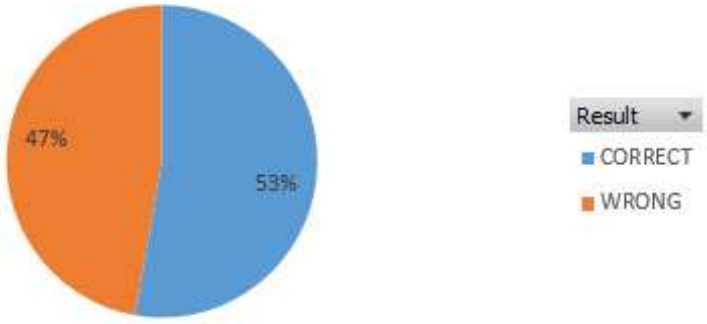


Figure 15. Only 53% of the student translators corrected the fuzzy matches.

The professional translators fared better: 76% edited the fuzzy matches correctly.

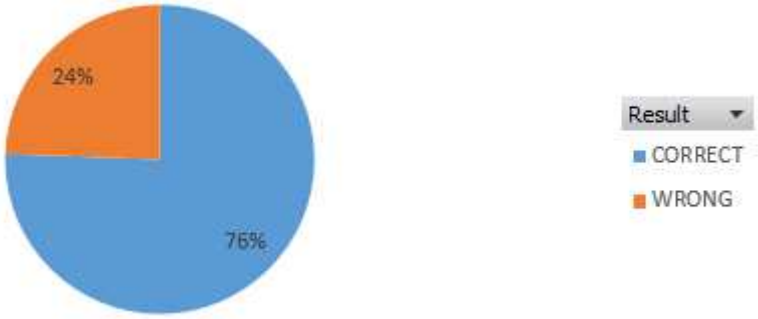


Figure 16. 76% of the professional translators corrected the fuzzy matches.

3.6 Evaluation of the results: Other aspects

Other aspects which the author noticed in the translations submitted by the professional translators – mistakes which the student translators did not make – was a lack of attention to detail, mainly too many spaces within a sentence or underlining a space where only a word should have been underlined.

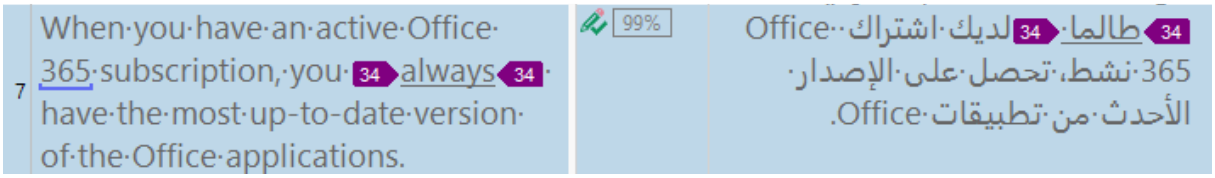


Figure 17. Too many spaces within the Arabic sentence (shown by two dots) and the underline goes over the space.

Several professional translators also had not confirmed segments, as shown below for segment 12. This is something which the student translators got right. One reason for this could be that the student translators worked with WFA which forces you to confirm and jump to the next segment, whereas in SDL Trados Studio 2015 it is possible to simply jump to another segment without first confirming it.

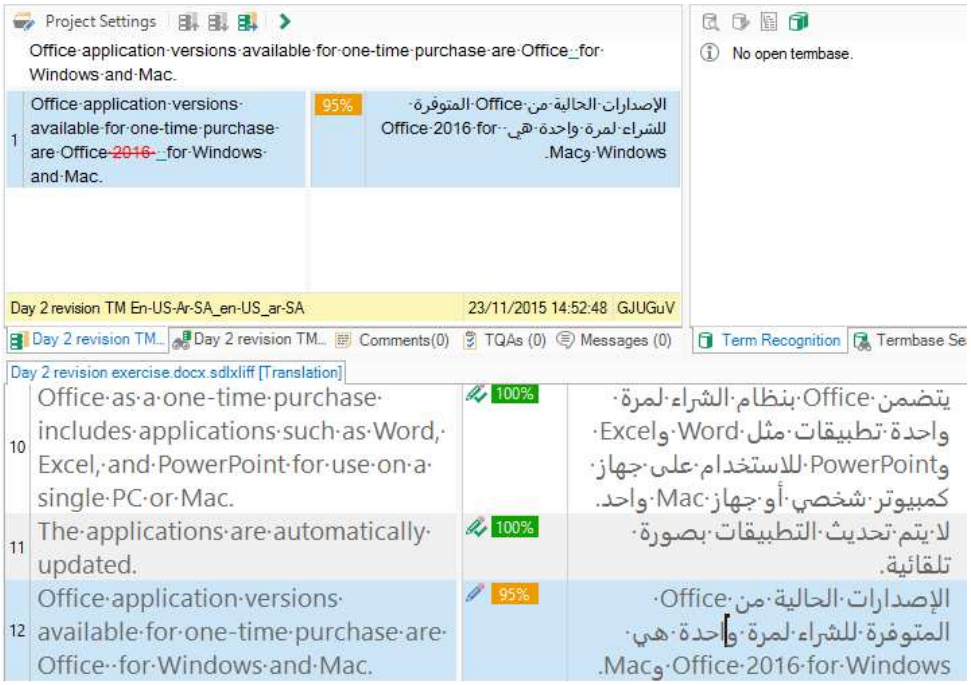


Figure 18. Segment 12 is unconfirmed and unedited.

There were also formatting issues in some of the professional translators' files, something which the student translators got right.

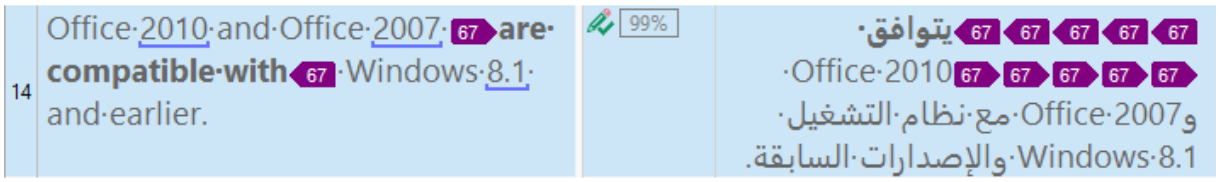


Figure 19. Too many formatting tags in the Arabic.

4 Conclusion and further research

The author is very glad that she has conducted this experiment as it has provided some interesting insights. It was interesting to see that while professional translators fared better at editing fuzzy matches than the student translators, they did not pick up on incorrect 100% matches as well as the student translators, tended to lack attention to detail by for example introducing double spaces into sentences, and not all professional translators translated the new sentences given for translation. The sample was small and therefore statistically insignificant, with Chinese and Arabic being the largest language groups in this experiment, however, the author believes that trends can be deduced from this experiment. It would be interesting to repeat this experiment with larger sample groups, other professional translators and undergraduate students who would be younger and be less likely to have been exposed to real translation work already.

Acknowledgements

The author would like to thank the following MSc Trans students for their help in compiling the student data: Yuchen Chen, Kristina Sarning-Haigh and Davide Stroppiana. The author would also like to thank her husband Chris Ford for helping with the evaluation.

References

Bowker, Lynne. Productivity vs Quality? A pilot study on the impact of translation memory systems. In *Localisation Focus (March 2005)*. https://www.localisation.ie/sites/default/files/publications/Vol4_1Bowker.pdf (last acessed: 7 October 2016)

Microsoft. What's the difference between Office 365 subscription plans and Office as a one-time purchase? https://products.office.com/EN/microsoft-office-for-home-and-school-faq?omkt=en (last accessed: 7 October 2016)

How translators can improve multilingual terminology in a link: teaching case study examples

Carmen Gomez-Camarero
Universidad de Málaga
gomez@uma.es

Rocío Palomares Perraut
Universidad de Málaga
perraut@uma.es

Abstract

This work describes a teaching method to enrich and improve the translators' and interpreters' multilingual terminology to translate specialized texts from a single link. This method consists of using controlled vocabularies such as thesauri, classification schemes, subject heading systems and taxonomies that employ Linked Open Data (LOD) technology in the framework of Semantic Web.

1 Introduction

The Web has become an unavoidable tool to get information. It is a platform where nodes such as objects, people and identities are interconnected. What they called *The Web of Things* (Atzori *et al.*, 2010). User translators collaborate and exchange information and knowledge in order to obtain the most accurate and appropriate data to ensure the quality of a translation work. To do that, however, the translator's literacy skills are crucial in determining the suitability to choose a selected appropriate term or expression. According to ACRL (Association of College and Research Libraries, 2016)), Information Literacy (IL) "forms the basis for lifelong learning. It is common to all disciplines, to all learning environments, and to all levels of education. It enables learners to master content and extend their investigations, become more self-directed, and assume greater control over their own learning".

In this context, IL is taught in Translation Studies at the University of Málaga (Spain) to developing in students the skills they need to get the accurate and precise information to translate. One of those skills is to access to the information required effectively and efficiently. But previously, they have to know which terms are the most accurate. Why? Because keywords are essential for searching online and for narrowing results. As well, they are vital part of any scientific text paper because someone has a previous and fast view of the content of the text (Vrkic, 2014; Yang, 2010). Students are familiar with dictionaries and terminological databases to look up terms and concepts to translate but they are not used to managing controlled vocabularies as a terminological tools (Lin *et al.*, 2009; Dancette, 2011; Ramírez-Polo, 2012; Dancette, 2015). We consider they are very interesting and usefull as we argue below.

It is thus in this work we describe a method to enrich and improve the translators' and interpreters' multilingual terminology to translate specialized texts from a single link. This method consists of using controlled vocabularies such as thesauri, classification schemes, subject heading systems and taxonomies. Those controlled vocabularies employ Linked Open Data (LOD) technology in the framework of Semantic Web (Pastor-Sánchez *et al.*, 2009; Pastor-Sánchez, 2011). This standard is about using the Web to connect related data that wasn't previously linked in a way that, in the context of terminology, each concept is uniquely identified by an URIs (Uniform Resource Identifier) and SKOS (Simple Knowledge Organization System) to support the use of knowledge organization systems.

This way of publishing and connecting data on the Web has myriad possibilities and challenges for accessing and identifying quality and up-to-date information and knowledge. Particulary, in the case of controlled vocabularies, they are betting in recent years in the rapidly expanding LOD landscape because is the latest advancement in the natural evolution

Proceedings of the 38th Conference Translating and the Computer, pages 81–87,
London, UK, November 17-18, 2016.

of the Semantic Web. Controlled vocabularies valued as authoritative are being structured and published to make them openly accesible and shareable on the Semantic Web (Summers *et al.*, 2008; Subirats-Colls, 2013).

These controlled vocabularies have various functions. These include, among others: gathering together the richness of variant terms and synonyms for concept and to link concepts in a logical order, sorting terms into categories, promoting consistency in preferred terms, improving and enriching the translators' multilingual and specialized terminology. Other very important propose of these controlled vocabulary is to organize information and provide terminology to catalogue and retrieve information. In doing so, the student can also increase their knowledge in accessing to the documents that have been catalogued and indexed by those controlled vocabularies.

2 Teaching practices using controlled vocabularies

To that end, some examples employed in class in order to teach these functions and usefulness of controlled vocabularies are described. The specialized texts used are taken from SINC (Servicio de Información y Noticias Científicas), a Spanish website to disseminate scientific information. Students must identify and select the proper keywords looking up these different controlled vocabularies: UNESCO Thesaurus, Spanish Cultural Heritage Thesauri (Tesauros del Patrimonio Cultural de España), Authorities Catalogue of the National Library of Spain (Catálogo de Autoridades de la Biblioteca Nacional de España –BNE-), Répertoire d'Autorité-Matière Encyclopédique et Alphabétique Unifié (RAMEAU) of the National Library of France and the Library of Congress Subject Headings (LCSH).

2.1 A teaching practice using Authorities Catalogue of the Spanish National Library (BNE)

We will describe in this work two practices. The first one consists in offering the student a text from SINC tittled "La dependencia y la edad son los principales factores de riesgo de maltrato en personas mayores" (http://www.agenciasinc.es/Noticias/La-dependencia-y-la-edad-son-los-principales-factores-de-riesgo-de-maltrato-en-personas-mayores) using one of the controlled vocabularies. In this practice, we employ the Authorities Catalogue of the National Library of Spain (Catálogo de Autoridades de la Biblioteca Nacional de España). This is a tool, normally used by National Libraries, to control documents for storing and retrieve information on the catalogue. The terms are known as Subject Headings and they should be looked up by the user first before starting a researching for information.

Shorly, the text is an Spanish medical research about the high risk in older people to be abused because of their age and grade of pshycological dependency.

The aims of this practice are:

a) Identifty the main keywords of the text, selecting only three terms that represent the content of the text. The three keywords should be as accurate as possible. These terms can be chosen from the title, from the abstract or even from the text.
b) Search these three terms in the Authorities Catalogue and verify if those are considered authorized and non-authorized entries.
c) Compare and choose authorized entries according to the content and meaning of the text .
d) Identify non-authorized entries in order to enrich their vocabulary.

e) Finally, in this practice, through LOD (Authorities Catalogue of the National Library of Spain is linked with LCSH and RAMEAU) students have to search others controlled vocabularies in other languages. This is our teaching goal in this practice: improve their multilingual terminology specialized of the text they are working on.

The second step of this practice consists in describe the keywords and their semantic relationships according to the Authorities Catalogue of the Spanish National Library (BNE). In this practice, we describe the three chosen keywords by students:

Abuse (Maltrato): Searching this term on Authorities Catalogue of BNE does not appear alone but the searching result is a list of a non-authorized terms that include this word, such as: *maltrato a las personas mayores, maltrato animal, maltrato entre alumnos, maltrato infantil, maltrato psicológico en el trabajo.* In this case, the student must choose the closest non-authorized term to the meaning of the text. That it is, *maltrato a las personas mayores.* This term is non-authorized term but is linked with an authorized term: *Ancianos-Malos Tratos.*

Variants of this concept (in Spanish Usado Por –UP-) and non-authorized terms are: *Agresión a los ancianos, Ancianos—Abusos, Malos tratos a las personas mayores, Maltrato a las personas mayores* and *Violencia contra los ancianos.*

Finally, LOD technology offers the possibility to check the equivalents of this concept in other controlled vocabularies: *Older people-Abuse* in English *(*LCSH) and *Personnes âgées-Violence envers* (RAMEAU) in French. At the same time, the student can browse and learn the same semantic relationships in other languages. Therefore, the student enriches widely his vocabulary in other languages in a link. That is why we think this way of working the specialized terminology is very interesting for translating.

Figure 1: Term "*Abuse of older people*"

Dependency (Dependencia): When searching for "dependencia" on Authorities Catalogue of BNE, the result is a list with different meanings for the same word. It is an authorized term but the meaning is not the equivalent according to the context of the text. It is referred to "political dependence". Therefore, the student applies his critical thinking skill (one of the IL

skills) to find out the right one among the other options. It is has to be said that any term is controlled by their scope. In our example, the meaning of "dependencia" and the right one is " psychological dependence".

As described above, the student has the link to LCSH to find the equivalent of this term in English and to RAMEAU to find the equivalent of that term in French.

Older people (Personas mayores): This multi-term concept is not on Authorities Catalogue of the BNE. Again, the student has to work his critical thinking to find a controlled and authorized synonym. One of them is "viejos" but non-authorized term, among others. But according to the meaning and scope of the text, there is "ancianos dependientes" which is the proper one. Once again, searching for an equivalent in English the student will find "frail ederly" on LCSH and in French *"personnes âgées dépendantes"* on RAMEAU.

Figure 2: Term "*Frail elderly*"

As we said above, students can extend their knowledge of the meaning of the text by cliking on the link "Obras", a link where you can reach the documents that are stored in the Spanish National Library (BNE) and indexed by the subject heading "Ancianos dependientes". In this example we have retrieved 4 documents, as we can see in the image below.

Figure 3: Documents about the term "*Ancianos dependientes*"

2.2 A teaching practice using the UNESCO Thesaurus

The aim of this second practice is to train the students' skills in using the multilingual UNESCO Thesaurus. We offer to the students a scientific popular text of approximately 500 words (http://www.agenciasinc.es/Noticias/Los-ninos-que-se-culpan-a-si-mismos-y-a-sus-familias-por-haber-sufrido-abusos-sexuales-tienen-mayores-tasas-de-estres-post-traumatico). After a quick reading of it, they have to choose three keywords which are supposed to represent the main subjects.

Roughly, the text is about the role of coping strategies and attributions of blame on the psychological adjustment of victims of child sexual abuse. According to that, the three main keywords chosen by the students are: Children, Sexual abuse and post traumatic stress disorder.

Children (Niños): On searching for the term "niños" in plural, the student find out that that term like this does not exist. So, they are trained to apply other flexible abilities and critical thinking skills to search the concept in other way like, for example, trying to find the concept in singular and not in plural. In doing that, students get 22 results related to the concept of children, as we can see in the picture below (Figure 4).

But "niño" is a descriptor too broad and it is expressed in the context of several narrower subjects, such as Addiction, Delinquency, Discrimination, and so on. Again, students have to practice their information literacy skills in trying to find the term "niño" closer to the concept "sexual abuses". UNESCO Thesaurus, as a controlled vocabulary, offers a list of hierarchical and related concepts of the concept "sexual abuse of children". Luckly, they find a related concepts (RT) "Child abuse". Students have to find a term "niño" related to "sexual abuses".

Each descriptor has a file card where are shown and are provided all the conceptual relationships between descriptors and non-descriptors and the equivalents in English and French as well.

Figure 4: Concept of "children"

Sexual abuses (Abusos sexuales): This term does not exist in plural. So, the student has to work the possibilites that UNESCO Thesaurus offers. In this case, the term "abuso sexual" is in singular. On the other hand, the student can check others terms for "abuso a menores" that are synonyms but non-descriptors. They are represented in the thesaurus with the symbol UP/UF (Usado Por/Used For): "violencia sexual", "violación", "acoso sexual".

Post traumatic stress disorder (Estrés post-traumático): Finally, this descriptor is not represented in UNESCO Thesaurus. In this case, the student, working their information literacy skills, has to choose for another solution. Searching for any other synonym such as "estrés mental", the student will find out other terms, such as: "tensión mentale" in French or "Mental stress" in English. In the same way, he can discover related and broader terms of this concept.

Using the Unesco Thesaurus the student can retrieve all documents stored in the Repository of the UNESCO known as UNESDOC, "a database that contains the full text and bibliographic records of documents and publications published by UNESCO since 1945 as well as bibliographic records of library acquisitions" (Unesco http://www.unesco.org/ulis/en/faq.html).

3 Conclusion

In conclusion, this kind of teaching practice is very effective because contributes to students' acquisition of some information literacy skills for their future as translators. These information literacy skills are:

- Extracting and identifying the main ideas of a text by keywords
- Identifying the necessary information sources for their translating work. In this case, selecting terminologycal sources such as Subject Headings and Thesauri.
- Getting to know, understanding and managing controlled vocabularies.
- Getting to know how to searches for information more effectively.
- Being flexible, creative and resourceful searchers to find data and information.

- Being capable to combine, reformulate, research, infer and experiment new ways of searching on the Web or any database because there is a saying "All the roads leads to Rome".

References

Atzori, Luigi, Antonio Iera and Giacomo Morabito. 2010. The Internet of Things: A survey, *Computer Networks*, 4: 2787–2805. DOI: 10.1016/j.comnet.2010.05.010c

ACRL. 2016. *Information Literacy Competency Standards for Higher Education*. Chicago: American Library Association.

Catálogo de Autoridades BNE. 2016. Madrid: Biblioteca Nacional de España.

Dancette, J. 2015. A context-rich dictionary with a relational structure: A tool for economic translation. *inTRAlinea*. Special Issue: New Insights into Specialised Translation. Retrieved from http://www.intralinea.org/specials/article/a_context_rich_dictionary_with_a_relational_structure (september 2016).

Dancette, J. 2011. L'intégration des relations sémantiques dans les dictionnaires spécialisés multilingues : du corpus ciblé à l'organisation des connaissances, *Meta*, 56 (2):284-300. URI: http://id.erudit.org/iderudit/1006177ar. DOI: 10.7202/1006177ar.

Lin, J., G.C. Murray, B.J. Dorr, J. Hajič, and P. Pecina. 2009. A cost-effective lexical acquisition process for large-scale thesaurus translation. *Language Resources and Evaluation*, 43 (1): 27-40.

Pastor-Sánchez, Juan-Antonio. 2011. *Tecnologías de la web semántica*. Barcelona: UOC.

Pastor-Sánchez, Juan-Antonio, Francisco-Javier Martínez-Méndez and José-Vicente Rodríguez-Muñoz. 2009. Advantages of thesaurus representation using the simple knowledge organization system (SKOS) compared with proposed alternatives. *Information research*, 14 (4), paper 422.

Servicio de Información y Noticias Científicas. 2016. *SINC: La ciencia es noticia*. Ministerio de Economía y Competitividad-FECYT. http://www.agenciasinc.es/ (September 2016).

Ramírez Polo, Laura. 2012. Los lenguajes controlados y la documentación técnica: mejorando la traducibilidad. *Revista Tradumática: Tecnologías de la traducción*, 10:192-204. Retrieved from http://ddd.uab.cat/pub/tradumatica/tradumatica_a2012n10/tradumatica_a2012n10p192.pdf (september 2016).

SKOS-Tesauro de la UNESCO. 2016. http://skos.um.es/unescothes/?l=es (September 2016).

Subirats-Coll, Imma. 2013. La Web Semántica y su aplicación en servicios de información : El caso de la Organización de las Naciones Unidas para la Alimentación y la Agricultura. In *Tercer Simposio Nacional De Patrimonio Bibliográfico y Documental*, Bogotá (Colombia), 30 September - 1 October 2013. [Presentation]. Retrieved from http://eprints.rclis.org/22452/1/simposionacionalbogota-131024043410-phpapp01.pdf (September 2016).

Summers, E., A. Isaac, C. Redding, C. And D. Krech. 2008. LCSH, SKOS and Linked Data. In *Proc. Int'l Conf. on Dublin Core and Metadata Applications,* pages 25-33. Retrieved from http://dcpapers.dublincore.org/pubs/article/viewFile/916/912 (september 2016).

Vrkic, Dina. 2014. Are they a perfect match? Analysis of usage of author suggested keywords, IEEE terms and social tags. In *37th International Convention on ,Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, Opatija (Croatia) 26-30 May 2014, pages 732-737. Doi: 10.1109/MIPRO.2014.6859662.

Yang, Chan-Jin. 2010. Study on Keywords and Their Use of Academic Theses - Focused on Database Development and Information Link. *Humanities Contents*, 19: 395-416.

Drawing a Route Map of Making a Small Domain-Specific Parallel Corpus for Translators and Beyond

Xiaotian Guo
New Vision Languages
Llandudno
North Wales, UK
garlickfred@gmail.com

Abstract

After years of development of corpus technologies, it has become obvious that translators can benefit directly from the achievements of this field. However, it seems that corpus advancement has not been deployed accordingly by translators to aid their translation. As a corpus linguist and translator myself, I believe that when corpus technologies are made attractive and simple enough and when they do feel a strong need and burning desire to make their own corpus to assist their translation, then application of such technology will gradually become part of a translator's life, just as other computer-assisted translation (CAT) tools have done over the past ten years or so. This paper attempts to make a demonstration as to how easy it can be to DIY a corpus by building a small domain-specific corpus between English and Chinese in the field of financial services. The making of such a corpus has been summarised into three simple steps: 1) Collection of raw parallel language data; 2) Alignment; 3) Segmentation and Annotation. It is hoped that other users of corpora including translation trainers, language teachers and students will also find this presentation informative and beneficial.

1. Introduction

At the Fourth International Conference on Corpus Use and Learning to Translate held in Alicante in 2015, there was a strongly felt concern that even though corpus technologies are available and abundant for translators to use, surveys had shown that a very small number of translators are currently deploying this new technology to aid their work in translation (Frérot, 2015). Actually, this phenomenon was addressed several year ago by Bernardini and Castagnoli (2008). In my opinion, there might be at least two reasons that have contributed to this undesirable outcome. One reason is that translators do not find it absolutely necessary to take the trouble to build and use a corpus of their own because they are short of time in learning the method of making a corpus and putting their hands on it while they can use the Google search engine instead (also see Bernardini, 2015). And the other reason is that the corpus technology has not been presented as a user-friendly and efficient means of assistance to translators yet, most probably due to its seemingly daunting complexity involved in it. Even though awareness has been raised of the usefulness of corpus technology for translators (i.e. Bowker, 2002; Quah, 2006), it seems that there is a need to simplify and systemise the process of construction with clearer instructions and perhaps more importantly examples while the technology details are presented to translators. As a corpus linguist and translator myself, I believe that when corpus technologies are made attractive and simple enough and when they do feel a strong need to make their own corpus to assist their translation, then they will gradually integrate the application of corpus technology into their translators work, just as other computer-assisted translation (CAT) tools have been in the past few decades. This paper attempts to make a demonstration through DIYing a corpus by building a small domain-specific corpus in the broad field of financial services, with English as the source and Chinese as the translation. The significance of this demonstration should be applicable to other domains. The paper is to introduce the necessary stages to take if a small domain specific corpus is to be build, i.e. a) collection of raw parallel texts, b) alignment of the collected parallel texts, and c) segmentation to the Chinese texts and annotation to both English and

Proceedings of the 38th Conference Translating and the Computer, pages 88–99,
London, UK, November 17-18, 2016.

Chinese. The issues of translation quality control and copyright in building a corpus by using texts collected from the web are addressed in brief. At the centre of my target readers are professional translators who are new to corpora, but I also hope that others such as translation trainees and teachers, language learners and teachers could find this paper informative and relevant to their interests.

2. Collection of raw parallel language data

The making of this corpus is primarily to demonstrate how it can aid translators in a specific domain, therefore, the size of the corpus does not have be very big. This section describes my method in collecting English-Chinese parallel texts in financial services. English is the source language and simplified Chinese (as mainly used in the mainland of China) is the translation. Collection of raw language data is carried out through a combination of pure manual collection and semi-automatic collection assisted by a web-crawling programme called Wget[1].

2.1 Pure manual collection

Unlike collecting monolingual texts, starting to collect parallel texts can be somewhat tricky. One way to be adopted is to find out some websites containing texts of the relevant languages in a searching engine such as Google. In order for Google to search from the internet some candidate websites, a few key words can be tried in both the source language English (in this case) and the translation language Chinese (in this case) in Google. For example, some of the English terms and their translation in Chinese can be typed into the Google search engine for a preliminary search such as *financial services*, *foreign exchange*, *trading*, *platform*, *risks*, *terms and conditions*, and the equivalent Chinese translation of these terms. When you have the retrieved websites by Google, you may select and open some potential websites for detailed look for possible parallel texts. Sometimes a few different sets of key words need to be fed into Google before serious candidate websites can be captured. Double quotation marks can be used to search multiple key words in a string so that Google concentrates on the exact phrasing instead of a combination of the individual words in the string. Sometimes you need to try different sets of key words for several times before there appear some websites containing the right information needed. Some other detailed advice for this purpose is available online for users reference[2]. Translators who find it necessary to build a domain specific corpus of their own normally would have known some websites containing potential parallel texts while they are working on their translation tasks. Therefore, these websites could serve as a starting point for the collection of raw parallel texts.

Manually collected parallel texts from the internet through keying bilingual key words or terms in a search engine are normally mixed in languages in one document and cannot be directly passed for alignment programmes to carry out alignment because most current alignment programmes only accept the input of the parallel data in the way that the source language is in one file and the translation in another (see Section 3 for alignment). Separating Chinese from English and saving the two texts originally in one document into two individual files takes two stages. The first stage is to separate the mixed texts with a uniform marker called delimiter to facilitate the recognition of the boundary of the two differently coded languages by the next software. In this process, regular expressions can be used to separate the two languages properly. It may take several rounds to conduct the separation completely due to the various situations of the mixture of the Chinese language. Other programmes using

[1] The manual is available at https://www.gnu.org/software/wget/manual/wget.html, last accessed on September 30, 2016.
[2] http://www.multilingual.ch/Help_find_parallel_texts_using_google_extended_help.htm

regular expressions such as Replace Pioneer can also be used for this purpose[3]. The second stage is to use Excel to input the delimited document with parallel texts and separate English and Chinese in two different columns which enables us to save the source file in one document and the translation in another by copy and paste, once the boundary of each string of source language is defined in relation to its translation.

2.2 The semi-automatic approach

Apart from manually collecting parallel texts, there are programmes available to assist this purpose. Since there must be human involvement to some degree and at some point, I would call this type of collection semi-automatic approach. There are many programmes used in the acquisition of parallel texts such as Parallel Text Miner (Nie, 1999), STRAND (Resnik, 2003), Bilingual Internet Text Search (Ma and Liberman, 1999), the Parallel Text Identification System (Chen et al. 2004), and Wget. This research is going to use Wget[4], a well-known and widely used freeware web-crawler used for downloading data from the internet. Once the programme is downloaded and installed onto your computer, you will need to read the manual before you test it for the first time. The programme works in command prompt, you will need some basic knowledge in handling file paths and commands to work with it. The Manual contains all the various and possible commands for different tasks but you do not have to know all of them in order to assign a task to it. Due to the importance of collecting raw data for the construction of the corpus, a detailed and clear guidance is provided. However, due to the space it takes, this part is included in Appendix A.

Knowing the names of as many websites as you need to search is crucial to the semi-automatic approach of data collection. You might need to start from the websites you know from the translation tasks you complete and gradually build up more websites relevant to your purpose of corpus construction. Some users prefer to make a base word list to feed the programme to maximise the output of the search in the internet (see Wang and Su, 2009), especially when the corpus to be constructed is meant to be big for purposes such as machine translation (for example, Koehn, 2005), dictionary compilation (for example, Héja, 2010), and term extraction (for example, Baisa et al, 2015). Since parallel texts are stored in different directories, normally one file of text contains one language only rather than two languages mixed together. This means two files of different languages can be directly passed for the aligner programme for alignment. But in cases when the two languages are mixed in one file, they would need to be separated by a uniform delimiter to enter the next stage (refer to Section 2.1 for details). A research by Zhang et al (2006) in their automatic acquisition of Chinese-English parallel corpus from the website uses a pre-defined strings indicating English and Chinese versions in several possibilities used in a website having several language versions (see Appendix B for details). This first-hand experience in parallel data collection could lend some ideas for the collection of parallel texts as well.

No matter the data collection is through a pure manual approach or a semi-automatic approach, it is necessary and worthwhile to have a scan of the texts to have a preliminary idea about the quality of the translation. Texts found with too many errors in translations even at a glance should be abandoned at this stage and not allowed to enter the next stage of corpus construction. The manual method sounds slow but can be pragmatic for building a small corpus. What is more important, the quality of the translation can be supervised more easily. However, the semi-automatic approach is faster downloading raw data for scrutiny obviously. A combination of the two approaches might suit most professional translators if the quality of translation is essential and overrides the quantity of the corpus.

[3] Visit http://www.mind-pioneer.com/ to download the programme, last accessed on September 29, 2016.
[4] https://www.gnu.org/software/wget/

3. Alignment of parallel texts

At the end of Section 2, collected parallel texts of the source language English and its translation Chinese of a particular title (or theme or topic) are stored in two separate files. For parallel texts to be converted to translation memories, each and every language unit (segment) has to be matched perfectly well, be it a full sentence or a clause or a phrase or even an individual word such as those in titles, headings, list items, and table cells. The processing of putting each segment in one line and different segments in different lines is called alignment and a programme conducting such a task is called an aligner. Normally, aligners use various parameters (also called anchors) for aligning segment pairs such as segment length and punctuation marks, which work well to certain language pairs, especially languages in a close family, and certain text types. Ideally, all the segments in the source should be matched by the same number of sentences in the translation, and all the punctuation marks in the two languages are exclusively identical. Actually, due to the differences between languages and cultures, it is difficult or even impossible for a translation to reach that level of equivalence. Take the English and Chinese language pair for example, a translator may have to use several simpler and shorter sentences in the Chinese translation to match a long and complex sentence in the source language English. And a question mark at the end of the English greeting "How are you?" would need a different punctuation mark in the Chinese translation "你好!"[5]. As a result, it is not surprising that an aligner would be able to do part of the job of automatic alignment but not the entire job. Human involvement would be a must to ensure each and every segment is correctly matched. There are many programmes that can carry out the task of alignment, for example, the Uplug by Tiedemann (2000) and of course the various versions of CAT tools including SDL Trados. In this study SDL Trados Studio 2014 will be used to demonstrate how this could be done. It takes a series of stages to complete the process of alignment, i.e. a) creating a translation memory (TM), b) introducing the two separate parallel text files into the Studio, c) correcting the alignment carried out by the aligner, and d) importing the segment pairs into the TM. To save space these details are included in Appendix C. Users who are familiar with the Studio need no further and more detailed explanations but for those who are new to the Studio and for those who have never used the function of aligning documents, they can refer to the manual of the Studio for other details. There are also tutorial videos on youtube introducing tips and tricks in alignment and TM import[6]. In the same manner, you can have your finely collected and selected parallel texts aligned and converted into your own TMs for reference. Apart from Studio 2014, users are recommended to try other aligners and see which one works better in a particular aspect and which one works better in other aspects. In my comparative investigation into Studio 2014 and another CAT tool developed in China called Xue-Ren (literally Snowman) CAT, I found that Xue-Ren CAT does a much better job in alignment than Studio 2014. For example, it is easier to split and merge both the translation and the source segments. And above all, it is possible to edit the source segments in Xue-Ren CAT which is a huge advantage if the user finds it necessary to do some editing on the source, for example to delete a serial number at the beginning of a segment which is not in the translation segment.

Till now, translators shall be able to benefit from corpus technology because the aligned segment pairs can be technically converted to a TM and aid translation in CAT programmes.

[5] Sadly, in the current day Mandarin teaching overseas, the daily greeting of Chinese equivalent to "How are you?" tends to be taught as "你好吗？" - a literal translation copy of "How are you?" Actually, in real Chinese conversations, this is seldom used.

[6] Refer to https://www.youtube.com/watch?v=EKlkZEkLL8E for details of the process, last accessed on September 24, 2016.

However, for those who wish to make a full use of the corpus (for those beyond translators such as translator trainers, language teachers and students) there is one step further to take to benefit more from the construction of the corpus.

4. Segmentation and annotation

After the data of your parallel texts has been properly collected and finely selected, there are two options to use it. As illustrated in Section 3, it could be aligned and then converted into a TM. It could also enter a different stage to be explored for different purposes. For example, how a parallel corpus can be used to examine the features of the languages under study in a particular syntactic structure, how a parallel corpus can be used for translator training for the awareness of language differences in different aspects of linguistic parameters, and how it can benefit terminologists in term retrieval, and even how it can aid language teaching and learning.

As is well known, the Chinese language is a distant language from English and differs from it in many different ways. Unlike the English language, Chinese characters are not separated by spaces and therefore the boundaries between Chinese characters are not defined. Unless the characters are separated semantically (no matter it is one character or two characters or multiple characters), there is no way for a programme to identify and process a given order. The process of separating Chinese characters is called segmentation. For the collected parallel texts to become a usable corpus, the Chinese files need to be processed by segmentation.

4.1 Segmentation

Segmentation technology is an important part of natural language processing to the Chinese language. There are programmes available for public use, among which ICTCLAS is probably the most often used for the segmentation of Chinese characters. The ICTCLAS (Institute of Computing Technology, Chinese Lexical Analysis System) (Zhang et al., 2003) can be downloaded free and installed in your computer.

By now the entirety of the parallel texts collected and selected into the final folder on your computer could be treated as a corpus, but it is a raw corpus. Simple queries such as individual words or strings of characters can be searched by a programme which is able to process two files at the same time to display the source language segment with the translation segment side by side. Such kind of programme is called a concordancer in corpus linguistics. Since processing a parallel corpus (two files and two languages at the same time) is different from processing a monolingual corpus, it takes a different concordancer to work on a parallel corpus. Currently, the most often cited concordancer for a parallel corpus is probably ParaConc by Barlow (2003). Of course, professional translators may also use the TM function of their CAT tools to examine the parallel texts in them. In a raw and untagged corpus, it is possible to examine part of the performance of individual words or characters, but it is not possible to examine the languages from a broader view such as how many verbs often occur in a particular syntactic structure in one of the languages (or even in the two languages) and what they are. For more complicated and specific queries, a corpus would need to be marked by certain tags to enable bulk identification and processing by a concordancer. Currently, the most often used and available tagging is part-of-speech (POS) tagging (also called annotation), which means each and every word would be tagged by the POS it belongs to in the context.

4.2 Annotation

Apart from the function of segmentation, ICTCLAS can also carry out POS tagging to the Chinese language. For the English language, there are many POS taggers available, but the most often cited and used one is probably CLAWS by developers at Lancaster. Created in the 1980's and developed over the last few decades, this programme requires a license to use which means a cost would be involved. After POS tagging, each word will be attached with a special coding of POS. Take the sentence "This is a black cat." for example, after POS tagging by CLAWS, the original sentence would be something like "This_DD1 is_VBZ a_AT1 black_JJ cat_NN1 ._." in which the POS of each word is encoded accordingly. Due to the differences of the two languages, the two tagging systems would have different parts of speech. However, this should not stop users from searching those parts-of-speech shared by the two languages. After the parallel texts are POS tagged, users may consult them for a much wider range of queries. For example, the belief held by some scholars can be tested that the Chinese language prefers the use of verbs whereas the English language favours the use of nouns. Supported by some analysis of the data generated, translation trainers, language teachers and students could have a better understanding of the two languages.

5. Using parallel corpora for other purposes

Apart from the values and potentials to professional translators, a parallel corpus can be very useful for other purposes such as natural language processing and machine translation, translator training, term construction and dictionary compilation, and language teaching and learning, and translation studies, to name only a few. Due to the space of this paper, I will concentrate on the use of parallel corpora in translator training and language teaching and learning.

5.1 Use of parallel corpora in translator training

Together with other types of corpora such as monolingual corpora, comparable corpora and learner corpora, parallel corpora can be used for translator training thanks to the advantages of bilingual and aligned parallel texts in a parallel corpus. How certain lexical items and syntactic structures are represented can always be examined in the other language through a bilingual concordancer. In translator training, the teacher can assign tasks to translator trainees to reflect some of the difficulties in the process of translation for this specific language pair. For example, due to the differences of the English and Chinese languages, long English sentences, especially with clauses are very often split into shorter clauses in the Chinese translation. Translation teachers could show theirs students how professional translators deal with this type of syntactic complexity through real translation examples. To further the training in this aspect, teachers could ask their students to translate long English sentences into shorter Chinese sentences in the translation. Even a further step, as a supplement, teachers may also ask their students to combine shorter Chinese sentences into suitable long English sentences for some practice of translation from Chinese to English. Of course, instead of the teacher dominating the classroom, translator trainees could have an active role to play in making use of a parallel corpus. While they are doing a translation exercise, they may consult a parallel corpus for a particular expression in the other language when they do not know how to express, or when they simply wish to confirm something about which they are not absolutely certain (see Yepes, 2011, for a few examples of using parallel corpora in translator training and see Zanettin et al, 2003 for some papers focusing on the topic "Corpora in Translator Education"). It can be envisaged that translator trainees

would have a better chance of developing not only their translation skills, but also their translation strategies and methodology, if they could observe the data carefully and sum up the implicit rules from individual examples.

5.2 Use of parallel corpora in and language teaching and learning

Language students and translator trainees have something in common in using a parallel corpus for study, although their study purposes are slightly different (the purpose of the former group is for general language acquisition whereas the purpose of the latter group is to get trained to become translators in the future). Therefore language teachers and students could use the strategies and methodology used by translator training in their language classroom. In addtion, there should be more values to explore and cultivate from a parallel corpus for language teachers and students. If we look at the literature, it would be easy to find a massive reservoir of theories and explorations in this aspect (for example, Barlow, 2000; Wang, 2001; Hunston, 2002, to name only a few). However, while we appreciate the usefulness and beauties of corpora, it is important that appropriate corpora should be selected to the right level of students because language study via examining corpora suits students of intermediate and advanced level more than beginners (see St. John, 2001). Using corpora (including parallel corpora) for language teaching and learning is probably the most discussed topic in the field of present day foreign language teaching and learning[7].

As shown above, a parallel corpus can be useful not only to professional translators but also translator trainees and language teachers and students. However, just as roses have thorns, a parallel corpus would have its own problems to be aware of. The next section talks about the issue of translation quality control and copyright awareness.

6. A few tips of caution and some advice

A corpus can be very useful after the construction is complete. However, there are at least problems users need to be cautious about: the translation quality control and the awareness of copyright. They should not assume that all translated texts would be of good quality and high standards simply because the texts have been published online or simply because they are from the website of a well-known company or organisation. Some translations, especially those of commercial agreements, terms and conditions, and guidance to products and services, are provided online for the reference of potential users only. They do not possess the degree of authority and integrity as the source documents have. That is why there are reminding clauses at the end of many legal documents alerting users that where there are conflicts between the original language and the translation the original language would prevail. Therefore, it is recommended that before any serious use of the corpus a check on the quality of the translation is carried out to ensure that the translation has reached the expected standards. Apart from the problem of translation qualities, builders and users of a corpus would face a thorny issue - the copyright issue sooner or later. Some owners of websites have stated explicitly that the information published to the public online is copyright protected and prohibits the use for commercial purposes without permission. However, there are some websites that do not prohibit the use of the information on them as long as users are aware of and responsible for any unexpected outcome that arises from the use of the information on the websites. On the safe side, it is recommended that a certain degree of caution be exerted for a selection of suitable websites and the copyright issue be cleared

[7] For a bibliography on the use of corpora and corpus-based methods in the language learning and teaching context, refer to http://www.corpora4learning.net/resources/bibliography.html, last accessed on September 28, 2016.

before use. Having said this, however, all these problems should not stop corpus users from exploring language use and translation skills in the relevant data strategically and carefully as a methodology.

7. Conclusion

I have chosen to make a parallel corpus because such a methodology can become immediately and directly usable by and useful to translators and others, and therefore more convincing to them, if they have a need or desire to DIY one for themselves. The purpose of this paper is to show that the task of making a small domain specific corpus is not as daunting as some people might think. However, it would be wrong to fall into the belief that the process of completing the task is extremely simple. As has been demonstrated in this paper, making a corpus involves several stages including preliminary planning, collection of raw materials, processing of parallel texts, converting parallel texts to translation memories and even necessary post-editing before use. It would take a considerable amount of time to complete the process, especially for those who have never put their hands on this kind of tasks before. If you wish to make a corpus on one day and to start using it the next day, you would need to have tried quite a few times using the methodology (or a similar one) and to have been quite familiar with the general routine and process shown above as your task requires. The application of corpora is almost ubiquitous thanks to the development of corpus linguistics over the past few decades. The fast growth of computer technologies and the capacity of computer processing and storage means a job taking several days in the past only requires a few minutes even seconds to complete now. Professionals in the broad area of translation should make the fullest use of this invaluable asset to aid their work. It can be envisaged that more and more professional translators would set up their own corpus to aid their work in translation, not because it is useful, not because it looks posh, not because other professionals are using it just like other CAT tools such as translation memories and term bases, but because it is gradually becoming a sort of advantage in getting jobs in the increasingly competitive translation market. It is hoped that this paper has attracted professional translators and the like one step closer to the decision to give it a go to the making of their own corpus after having heard about it for a long time.

Acknowledgements

Thanks are given to Dr. Huaqing Hong at Nanyang Technological University (Singapore) for sharing with me some of the tools used in this research.

References

Baisa, Vít, Barbora Ulipová, and Michal Cukr. 2015. Bilingual Terminology Extraction in Sketch Engine. In *Ninth Workshop on Recent Advances in Slavonic Natural Language Processing*, pages 61–67.

Barlow, Michael. 2000. Parallel Texts in Language Teaching. In Botley, Simon, Anthony McEnery, and Andrew Wilson (eds.) *Mutiligual Corproa in Teaching and Research*. Rodopi, Amsterdam, pages, 106-115.

Barlow, Michael. 2003. *Paraconc: A Concordancer for Parallel Texts*. Athelstan, Houston.

Bernardini, Silvia. 2015. Exploratory Learning in the Translation/Language Classroom: Corpora as Learning Aids. Paper presented in the CULT Conference, Alicante.

Bernardini, Silvia and Sara Castagnoli. 2008. Corpora for Translator Education and Translation Practice. In *Topics in Language Resources for Translation and Localisation*. John Benjamins, Amsterdam, pages 39-55.

Bowker, Lynne. 2002. *Computer-Aided Translation Technology: A Practical Introduction*. University of Ottawa Press, Ottawa.

Chen, Jisong, Rowena Chau, and Chung-Hsing Yeh. 2004. Discovering Parallel Text from the World Wide Web. In *Proceedings of the Second Workshop on Australasian Information Security, Data Mining and Web Intelligence, and Software Internationalisation*, pages 157–161.

Frérot, Cécile. 2015. Corpora and Corpus Technology for Translation Purposes in Professional and Academic Environments. Major Achievements and New Perspectives. Paper presented in the CULT Conference, Alicante.
Héja, Enikö. 2010. Dictionary Building Based on Parallel Corpora and Word Alignment. In Dykstra, Anne and Tanneke Schoonheim (eds): *Proceedings of the XIV. EURALEX International Congress*, pages 341-352.
Hunston, Susan. 2002. *Corpora in Applied Linguistics*. Cambridge University Press, Cambridge.
Koehn, Philipp. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings of MT Summit X*, pages 79-86.
Ma, Xiao-Yi and Mark Liberman. 1999. BITS: A Method for Bilingual Text Search over the Web. In *Proceedings of Machine Translation Summit VII*, pages 538–542.
Nie, Jian-Yun, Michel Simard, Pierre Isabelle, and Richard Durand. 1999. Cross-language information retrieval based on parallel texts and automatic mining of parallel texts from the web. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 74-81.
Quah, Chiew Kin. 2006. *Translation and Technology*. Palgrave Macmillan, Hampshire and New York.
Resnik, Philip and Noah A. Smith. 2003. The Web as a Parallel. In *Corpus Computational Linguistics*, Volume 29, Issue 3, pages 349-380.
St John, Elke. 2001. A Case for Using a Parallel Corpus and Concordancer for Beginners of a Foreign Language. In *Language Learning and Technology*. Volume 5, Number 3, pages 185-203.
Tiedemann, Jörg. 2000. Extracting Phrasal Terms Using Bitext. In *Proceedings of the Workshop on Terminology Resources and Computation*, pages 57-63.
Wang, Dong-Bo, Xin-Ning Su. 2009. Automatic Building of Sentence Level English-Chinese Parallel Corpus. In *New Technology of Library and Information Service*. Issue No. 12, pages 47-51.
Wang, Li-Xun. 2001. Exploring Parallel Concordancing in English and Chinese. In *Language Learning and Technology*, 5(3), pages 174-184.
Yepes, Guadalupe Ruiz. 2011. *Parallel Corpora in Translator Education*. http://www.redit.uma.es/archiv/n7/4.pdf [last accessed September 30, 2016].
Zanettin, Federico, Silvia Bernandini, and Dominic Stewart (eds). 2003. *Corpora in Translation Education*, Routledge, London and New York.
Zhang, Hua-Ping, Hong-Kui Yu, De-Yi Xiong, and Qun Liu. 2003. HHMM-based Chinese Lexical Analyzer ICTCLAS. In *Proceedings of the Second SIGHAN Workshop on Chinese Language Processing*, pages 184–187.
Zhang, Yi, Ke Wu, Jian-Feng Gao and Philip Vines. 2006. Automatic Acquisition of Chinese-English Parallel Corpus from the Web. In *Proceedings of ECIR-06*, pages 420-431.

Appendix A: Using Wget to download parallel texts

Suppose you wish for Wget to download the website www.example1.com onto your own computer, the command you need to type onto the command prompt[8] would be C:\wget wget -r www.example1.com in which C:\wget shows the path of wget, wget activates the executable programme called wget, the letter r in -r (meaning recursive) shows how the recursive downloading is going to take place, and the www.example1.com is the website to be downloaded. Wget is able to display the directories of a website in the way the directories are stored in the original website, enabling users to have an idea how the directories and sub-directories are related to each other.

Some websites contain several versions of different languages in different directories and even sub-directories under the same root domain. For example, the English text from www.example2.com/en/ and the Chinese text of translation from www.example2.com/cn share the same root domain www.example2.com. However, some organisations use totally different domains for their different versions of websites in which case you will need to type in two domains to acquire parallel texts.

The raw data withdrawn by Wget may be downloaded into different directories but may also be into the same directory. As a clue, the name of folders such as en (for English) or cn

[8] Assuming that you have saved the Wget directly under the C disk.

(for Chinese) indicates the information under the directory en would be related to the English version of the website whereas the directory cn would be related to the Chinese version. Sometimes, when you open a directory, you find it empty. This might be because the depth of downloading has been reached and Wget cannot dig any deeper than that level of directory. The default depth of downloading is five. Wget is able to process multiple websites in one command. Therefore, it is worthwhile to make a list of website domains as a file and save it in the same directory as the Wget application programme and give a relevant command to download all these websites in one go. Downloading files from the servers of documents takes time depending on how fast your internet connection speed is and how responsive the servers are. It is the advantage of Wget that you may leave the computer working on the task in the background while you work on something else.

Appendix B: A list of pre-defined strings

english
chinese
simplified chinese
chineses implified
traditional chinese
chinese traditional
english version
simplified chinese version
traditional chinese version
英文 英文首页
简体 中文首页
繁體 中文简体
英文版 中文繁體
中文版 简体中文
简体版 简体中文版
+繁體版 繁體中文
英文网站 繁體中文版
中文网站

Appendix C: Using SDL Trados Studio 2014 for parallel text alignment and converting the aligned parallel texts into a TM

1. Creating a translation memory

For the parallel texts to be aligned and afterwards to become a part of your translation memory, open SDL Studio 2014 (henceforth called the Studio) and enter the TM database from the left panel where the user can create a new translation memory. If you wish to add these parallel texts to one of your existing TM, you can skip this process and add it to your alignment project when you are asked by the Studio to do so. It is recommended that a new TM is created at this moment because you may like to do some post-editing for quality assurance purpose. For professional translators who are familiar with the Studio, they can wait for later to create a new TM at a later stage while the parallel files are selected from your computer.

2. Introducing the two separate parallel text files into the Studio

The two separate files can be introduced to the programme from the "align documents" function under the Home menu. In the drop down menu of "Align Documents" choose "Align Single File Pair" and browse to pick your source language file first and then your translation file second so that the source language comes on the left column of the alignment window. If you have created a new TM just now, then click "Add" on the top of the small window to pick the TM. If you have already set up a TM before this new and empty TM , click "add" to pick it in the same way. For those skillful users of Studio who have not created a new TM at the beginning, this is the time to do it. Once the parallel files have been selected and the TM created or added, click "Finish" to complete this process.

3. Correcting the alignment carried out by the aligner

Ideally, the aligner could "understand" the segments of the two languages and put them in the cells in which the right column is exactly the translation of the left column, no more and no less. But in fact this is seldom the case especially when the files are big, due to various reasons including those mentioned above. This is where the human involvement must come in. At this moment, there are four columns in the alignment window. In the left two columns are the segments of the source language and their correspondent serial numbers. In the right columns are the serial numbers of all the segments of the translation, matching the serial numbers of the source segments, and the segments of the translation. There are links between the source segments and the translation segments with different colours (green, yellow and red). The green colour shows that there is a better chance that the segments in the right column is the translation of the segment in the left column. The yellow colour shows less confidence and the red colour even less. Experienced users of the alignment function of the Studio would normally disconnect all the pre-connected links. This is because it is easier to "Disconnect All" first and then find the right pairs than disconnecting and finding individual matching ones. To select one pair of segments, hover your cursor on the serial number of one of the segments and click (the background colour will change as a response) and move cursor to the serial number of the other segment and right click your mouse to disconnect or connect. Sometimes more than one cell needs to be executed and this is can be realised by clicking one of the cells first and then click the next line whilst pressing the control key.

4. Saving the alignment and importing the segment pairs into the TM

When each and every segment is correctly connected to its pair, the alignment project can be saved for future use and what is more important the well aligned segment pairs can be saved into the TM through the function "Import Into Translation Memory" under the Home menu. The TM can be tested like other TMs in translation tasks. It is expected that with the TM added into the translation project, once the source file is introduced to the Studio for translation, the newly created TM will be triggered automatically and each and every source segment will be matched in the translation column by their correspondent segment in the TM.

Appendix D: Splitting English sentence and Chinese sentence into different lines[9]

Download and install "Replace Pioneer" on windows platform to finish following steps.
1. ctrl-o open text file
2. ctrl-h open 'replace' dialogue
* set 'search for pattern' to:
  [A-Za-z][a-zA-Z\W]{15,}
* set 'replace with pattern' to:
  \n$match\n
3. click 'replace', done.
Note1: if you only want to add a # before a Chinese sentence, you can set 'replace with pattern' to: $match# in step2.
Note2: we allow Chinese sentences to contain English word less than 15 letters. The user can change 15 to other number in [A-Za-z][a-zA-Z\W]{15,} in step 2.
To see a screenshot of the Replace Pioneer window, visit http://www.mind-pioneer.com/services/1351_Advanced_search_and_replace.html to see the page.

[9] This content of this appendix is based on the content from http://www.mind-pioneer.com/services/1351_Advanced_search_and_replace.html, last accessed on September 29, 2016.

A Case Study of German into English by Machine Translation: to Evaluate Moses using Moses for Mere Mortals

Roger Haycock
Haycock Technical Services
Purley Rise LE12 9JT
rhaycock@theiet.org

Abstract

This paper evaluates the usefulness of Moses, an open source statistical machine translation (SMT) engine, for professional translators and post editors. It takes a look behind the scenes at the workings of Moses and reports on experiments to investigate how translators can contribute to advances in the use of SMT as a tool. In particular the difference in quality of output was compared as the amount of training data was increased using four SMT engines.

This small study works with the German-English language pair to investigate the difficulty of building a personal SMT engine on a PC with no connection to the Internet to overcome the problems of confidentiality and security that prevent the use of online tools. The paper reports on the ease of installing Moses on an Ubuntu PC using Moses for Mere Mortals. Translations were compared using the Bleu metric and human evaluation.

Introduction

Pym (2012) considers that translators are destined to become post-editors because the amalgamation of statistical machine translation (SMT) into translation memory (TM) suites will cause changes to the skills required by translators. He believes that machine translation (MT) systems are improving with use and a virtuous circle should result. However, free online MT, for example Google Translate (GT), could lead to a vicious circle caused by the recycling of poor unedited translations. Technologists have a blind faith in the quality of translations used as 'gold standards' and Bowker (2005) found that TM users tend to accept matches without any critical checks. Further, the re-use of short sentences leads to inconsistent terminology, lexical anaphora, deictic errors and instances where the meaning is not as foreseen in the original. Pym (2010, p.127) suggests that this could be avoided if each organisation has its own in-house SMT system.

There is another compelling reason for in-house SMT. Achim Klabunde (2014), Data Protection Supervisor at the EU warns against using free translation services on the Internet. He asserts that someone is paying for what appears to be a free service. It is likely that users pay by providing their personal data. Translators using these services may however be in breach of confidentiality agreements because the data could be harvested by others and recycled in their translations. Googles terms and conditions are clear that they could use any content submitted to their services (Google, 2014).

The increased volume of translation caused by localisation does, however, call for automation (Carson-Berndsen et al, 2010, p.53). Advances in computer power have enhanced MT and good quality full post editing can be included in TM for segments (sentences, paragraphs or sentence-like units eg. headings, titles or elements in a list) where no match or fuzzy match is available (Garcia, 2011, p.218). TM developers now offer the facility to generate MT matches to freelance translators, eg. Google Translator Toolkit, Smartcat, and Lilt. This technology will increase the rate of production and according to Garcia (2011, p.228) industry expects that post-editing, with experienced post-editors and in-domain trained engines, for publication should be able to process 5000 words a day. Pym (2012, p.2) maintains that post-editors will require excellent target language (TL) skills, good subject

Proceedings of the 38th Conference Translating and the Computer, pages 100–112,
London, UK, November 17-18, 2016.

knowledge but only weak source language (SL) skills. For this reason I elected to work from German, my weaker foreign language, into English my native language. I have used Lilt to post-edit translated segments to help with evaluation of the SMT as will be explained later.

This paper reports a case study that used Moses for Mere Mortals (MMM) to investigate how difficult it might be for a freelance translator to incorporate an in-house SMT engine into a single workstation by building four distinct Moses engines with different amounts of data. Following an overview of the project the method followed to install, create, train and use the Moses engines using MMM is explained. Then an explanation of how the raw MT output was obtained, processed and evaluated will be given before presenting the results and drawing a conclusion.

Overview

A study carried out by Machado and Fontes (2011) for the Directorate General for Translation at the European Commission forms the basis for the methods adopted in the experiments. The aim was to explore integrating a personal SMT engine into a translator's workbench whereby the MT system was to be within a single computer with no connection to the Internet. It is trained with data that the user owns or has permission to use. The MT output is post-edited by the user to a level that Taus (2014) defines as being comprehensible (i.e. the content is perfectly understandable), accurate (i.e. it communicates the ST meaning) and the style is good but probably not that of a L1 human translator. Punctuation should be correct and syntax and grammar should be normal as follows:

- The post-edited machine translation (PEMT) should be grammatically, semantically and syntactically correct.
- Key terminology should be correctly translated.
- No information should be accidentally added or omitted.
- Offensive, inappropriate or culturally unacceptable content should be edited.
- As much raw MT output as possible should be used.
- Basic rules of spelling, punctuation and hyphenation should apply.
- Formatting should be correct.

McElhany and Vasconellos (1988, p.147) warn that because editing is not rewriting corrections should be minimal.

To carry out this study, I installed Moses on a desktop computer using MT software MMM that claims to be user-friendly and able to be understood by users who are not computational linguists or computer scientists. Such users are referred to as 'mere mortals' (Machado and Fontes, 2014, p. 2).

A large parallel corpus is required for training Moses. TM is ideal for this because it produces aligned bi-texts that can be used with minimal changes. The Canadian Parliament's Hansard, which is bilingual, was the source of data for early work on SMT (Brown et al, 1988, p. 71.)

A data source created and often used to promote the progress of SMT development is the Europarl corpus produced from the European Parliament's multilingual proceedings, which are published on the EU website. Koehn (2005) arranged them into the corpus. He confirmed that it can be used freely (personal communication, 25 January 2016). It was chosen to

simulate a TM for this project because it was used in the study made by Machado and Fontes (2011). When aligned with German it has 1,011,476 sentences.

I used MMM to build four MT systems with different amounts of data and tested them with a test document of 1000 isolated sentences extracted, together with their translations from the corpus.

Moses' developers suggest that by varying the tuning weights it is possible to tune the system for a particular language pair and corpus (Koehn, 2015, p.62). MMM facilitates some adjustments and the effect of these was studied using the largest training.

Before explaining how the experiments were conducted I will describe how I installed MMM and built the Moses engines.

.

Equipment, and software installation

MMM (Machado, and Fontes 2014, p.2) is intended to make the SMT system Moses available to many users and is distributed under a GNU General Public Licence (p.10). There is a very comprehensive MMM tutorial (Machado, and Fontes 2014) giving a step-by-step guide to SMT for newcomers like myself.

The tutorial recommends a PC with at least 8GB of RAM, an 8 core processor and 0.5 TB hard disk (p.14) but no less than 4 GB of RAM and a 2 core processor. I used a machine with 8 GB of ram, 4 processors but only a 148GB hard disk. There is a 'transfer-to-another-location' script that can be used to transfer training to another machine with a much lower specification for translating/decoding only. I tried this using a 1GB laptop but would not recommend it. It was able to complete the translation but it took hours rather than the minutes taken by the 8GB machine.

MMM consists of a series of scripts that automate installation, test file creation, training, translation and automatic evaluation or scoring. Following the tutorial, I installed Ubuntu on the computer, choosing 14.04(LTS)(64 bits), although MMM will also run on 12.04 (LTS).

The next step was to download a zipped archive of MMM files and unpack them onto the computer.

The MMM tutorial explains how to prepare the corpus for training and build the system. Training the full Europarl corpus took 30 hours.

Although Ubuntu has a Graphical User Interface (GUI), I preferred the Command Line Interface (CLI) (see figure 2). The script 'Install', was run next to install all the files required onto the computer. MMM includes all the files necessary to run Moses but the script downloads, any Ubuntu files that are needed but not present in the computer from the Internet

With MMM installed running the 'Create' script completes installation of Moses.

There is a demo corpus that translates from English into Portuguese included with MMM for trying out Moses. I used this to experience preparing the corpora, extracting test data, translating and scoring before doing it with the German and English parts of the Europarl corpus.

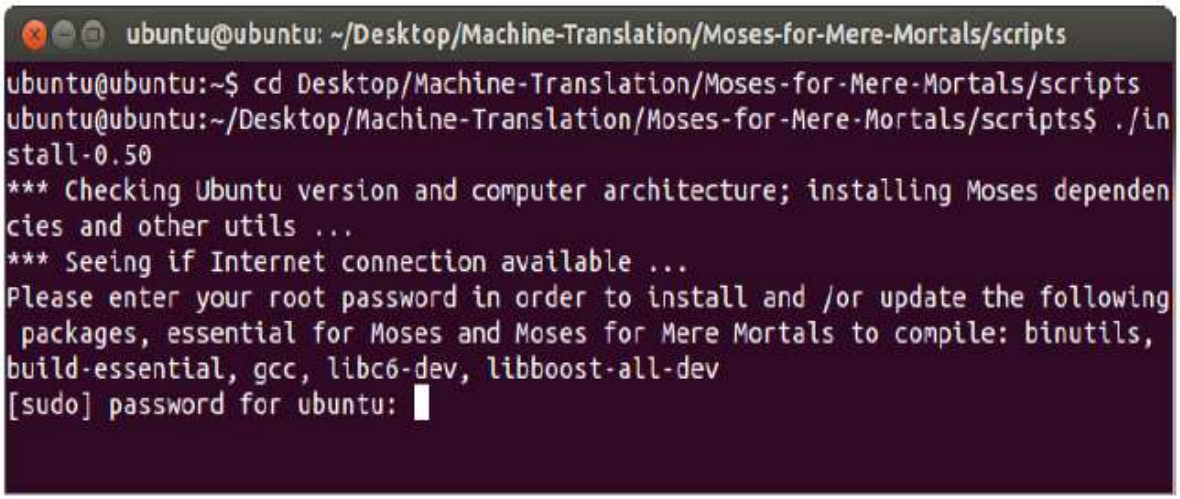


Figure 1 Installing Ubuntu packages

Preparation of the corpora.

The 'Make-test-files' script was used to extract a 1000 segment test file from the Europarl corpus before using it for training.

Training

With MMM it is possible to build multiple translation engines on one computer. Where possible files generated by earlier ‘trainings’ are re-used. The training to be used for a particular translation is selected from the ‘translation’ script.

A monolingual TL corpus is used to build the language model. I used the English side of the bilingual corpus in all trainings. The aligned texts are placed in a folder named ‘corpora-for- training’ and the train script is run. When the training is complete a report is generated that is required by the ‘translation’ script to select the correct training.

Four basic trainings were built and tested. The first used the whole corpus. Then a second engine was built by splitting out the first 200,000 segments. This was repeated for 400,000 and 800,000 segments. The 1,000 segment test document was translated by each of the engines and Bleu scores obtained using the MMM ‘Score’ script. A sample of 50 segments from each translation was post-edited and evaluated by me.

Translation

The tests were divided into two parts. Before studying the difference in translation quality using the different sized corpora, the effect of the tuning weights was examined with the whole corpus.

With the German ST part of the test document in the ‘translation-in’ folder and the required data entered into the ‘Translate’ script, translation was initiated by running the script from the CLI.

According to Koehn (2015, p.62) a good phrase translation table is the key to good performance. He goes on to explain how some tuning can be done using the decoder. Significantly, the weighting of the four Moses models can be adjusted. They are combined by a log linear model (Koehn, 2010, p.137), which is well known in MT circles. The four models or features are:

- The **phrase translation** table contains English and German phrases that are good translations. A phrase in SMT is one or more contiguous words. It is not a grammatical unit.
- The **language model** contributes by keeping the output in fluent English.
- The **distortion model** permits the input sentence to be reordered but at a price: The translation costs more the more reordering there is.
- The **word penalty** prevents the translations from getting too long or too short.

There are three weights that can be adjusted in the 'Translation' script. These are Wl, Wd and Ww. They have default values of 1,1 and 0.

The tuning weights were adjusted in turn using the translation script.

With all the weights left at their default levels I produced the first translation. The reference translation was placed in the MMM 'reference-translation' folder and a Bleu score was obtained by running the 'score' script. I then post-edited 50 segments and evaluated the MT as explained below. This was repeated with Wd set to 0.5 and then 0.1. Then with Wd set back at 1, and with Wl set to 0.5 and then 0.1 further MTs were gathered and evaluated.

Similar experiments were conducted with Ww set to –3 and then 3 and with Minimum Bayes Risk (MBR) decoding (MBR decoding outputs the translation that is most similar to the most likely translation).

Having explained how the system was built and the MTs obtained, the methods used to evaluate the results will be described.

Evaluation

A total of 8 measurement points generated translations that were evaluated by both automatic and manual techniques.

Metrics

Machado and Fontes (2011, p.4) utilised the automatic evaluation metric Bleu (bilingual evaluation understudy), which compares the closeness of the MT to a professional translation relying on there being at least one good quality human reference translation available (Papineni et al, 2001, p.1). It is measured with an n-gram algorithm developed by IBM. The algorithm tabulates the number of n-grams in the test MT that are also present in the reference translation(s) and scores quality as a weighted sum of the counts of matching n-grams. In computing the n-gram overlap of the MT output and the reference translation the IBM algorithm penalises translations that are significantly longer or shorter than the reference. For computational linguists Bleu is a cheap quick language independent method of evaluation and correlates well with human techniques (Papineni et al, 2001).

In many cases this correlation has been shown to be correct (Doddington, 2002, p.138-145) and a study by Coughlin (2003, p.6) claims that Bleu correlates with the ranking of the TM and also 'provides a rough but reliable indication of the magnitude of the difference between the systems'. However, Callison-Burch et al (2006) take the view that higher Bleu scores do not necessarily indicate improved translation quality and focused manual evaluation may be preferable for some research projects. They conclude that for systems with similar translation structures Bleu is appropriate.

Manual Evaluation

Machado and Fontes (2011) had a team of translators performing human translations of a sample of segments even though human evaluations of MT output are extensive, expensive and take weeks or months to complete (Papineni, Roukos, Ward and Zhu, 2001, p.1). White (2003, p.213) points out that they are very subjective because there is no 'right' translation, as there is never any agreement on which is the best. Newmark (1982, p.140) is convinced that the perfect translation does not exist, but if it does Biguenet and Schulte (1989, p.12) are sure that it will never be found. Evaluators are always biased (White, 2003, p.219). For example, seeing a really bad segment might make the next one seem relatively better than it is and vice-versa. Another example is where a mistake such as a trivial software bug is forgiven. An evaluator may also become bored or tired, resulting in segments graded early in the cycle receiving a more favourable treatment to those graded later.

'Fluency' and 'adequacy' are commonly used for evaluating raw MT output. Two scores are combined, averaged, and written as a percentage. Machado and Fontes (2011, p.7) did not follow this method and use what they call 'real life conditions' by post-editing and classifying the effort required for each segment on a scale of 1 to 5.

Their scale was adopted in this study:

1. Bad: Many changes for an acceptable translation; no time saved.
2. So So: Quite a number of changes, but some time saved.
3. Good: Few changes; time saved.
4. Very Good: Only minor changes, a lot of time saved.
5. Fully correct: Could be used without any change, even if I would still change it if it were my own translation.

Machado and Fontes do not mention whether or not time saved was measured but they do say that their objective was classifying segments by translation quality. Only a translation that can be used without change scores 5. A segment that is understandable and correct apart from one or maybe two errors receives a score of 4. One that should be translated from scratch scores 1.

Scores were recorded segment-by-segment on a spreadsheet and averaged for the fifty segments. Since I was the only post editor 50 different segments were post-edited for each MT. This avoided previous knowledge influencing the scoring and permitted a PEMT version of the test text to be gradually produced. For consistency with the Bleu scores the averages were divided by 5 to express them on a scale of 0 to 1.

There were eight measurement points in the first part of the experiments. A further four measurement points were made for the second part. These were for the 200000, 400000, 800000 and, for comparison, GT.

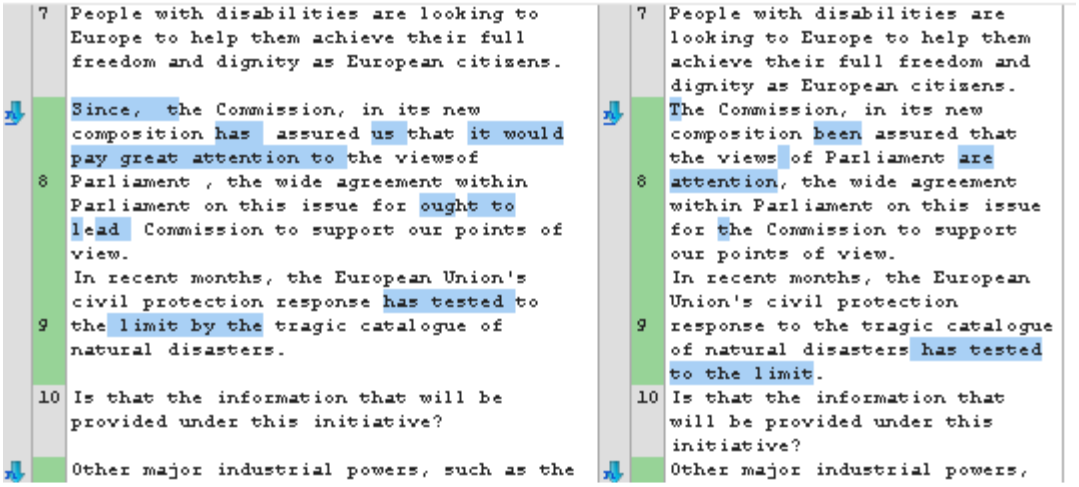


Figure 2 from text-compare.com

For the second part a third method of evaluation was introduced. This was based on Translation Edit Rate (TER), as a less subjective technique to check the quality of assessment, which should be quantitative.

TER is defined as the lowest number of edits needed to change a MT segment so that it matches the reference, which is the post-edited segment, normalised by the length of the reference (Snover et al, 2006, p.3).

TER = number of edits / number of words in the PEMT sentence.

The raw MT was compared with the post-edited text using text-compare.com as shown in figure 3. The number of edits and the number of words in the PEMT were counted manually.

I termed this hter because of the human involvement. I used a spread sheet to obtain a score for each segment, which were then ranked on a scale of 1 to 5 as follows:

TER	Hter
0	5
0 to 0.25	4
0.25 to 0.5	3
0.5 to 0.75	2
>0.75	1

These segment scores were then averaged over the 50 segments to obtain an overall score.

Since the problems associated with using online MT systems are the rationale for investigating personal SMT, a comparison with the online MT engine GT was made. In order to compare the quality of the MMM trainings with GT, an MT was generated using GT and Bleu, human and hter scores were also obtained.

Results

Before seeing and discussing the results we will look at some sample translations that demonstrate the scoring levels. The source text (ST) is a segment from the test document and the reference translation (RT) is the corresponding English segment. The raw machine translation (MT) was produced by Moses. The MT was post-edited (PE) by me.

Starting with an example that scored 5

ST : *Wir müssen und können handeln.*
MT : We can and must act.
RT: We can and must take action.

I considered that this was a good translation based on the TAUS guidelines and did not require post-editing. The reference translations are given as an aid to non-German speaking readers. They are not necessarily better or worse than the MT or my PEMT.

The next example was given a score of 4. It only needed a few minor edits.

ST: *Wir halten es für unbedingt notwendig und nicht weniger dringlich, dass wir alle gemeinsam - und natürlich mit der völlig unabdingbaren Unterstützung dieses Parlaments - darauf hinwirken, dass dieses Recht der Petersberg-Aufgaben sofort zur Anwendung kommen kann, wenn diese Missionen ausgeführt werden.*

MT: We believe it is essential, and no less urgent, that amongst all of us - and with the completely indispensable cooperation of this Parliament - we start creating this law for Petersberg tasks to be applied if this mission

PE: We believe it is absolutely essential, and no less urgent, that between all of us - and with the completely indispensable cooperation of this Parliament - we start working towards this law for Petersberg tasks being immediately applied if these missions are carried out.

RT: We believe it is essential, and no less urgent, that amongst all of us - and with **t**he completely indispensable cooperation of this Parliament of course - we start creating this law for Petersberg tasks, which can be applied from the start of any mission.

A score of 3 was given to the following example. The MT cannot be understood

ST: *Die Vorstellungen des Vorsitzes im Umweltbereich klingen zwar gut, sollten aber in Resultate umgemünzt werden.*

MT: Which the presidency on the environment is sound, but results umgemünzt.

PE: The presidency's ideas on the environment sound good, but should be converted into results.

RT: The presidency's ideas in the environmental field sound good but should be translated into results.

Weight change	Average Human score	Average Human score /5	Bleu score
Default	3.78	0.756	0.5076
Distortion weight=0.5	3.58	0.716	0.5955
Distortion weight = 0.1	3.48	0.696	0.5912
Word penalty weight =3	2.36	0.472	0.342
Word penalty weight = -3	2.88	0.576	0.376
Language model weight =0.1	3.82	0.764	0.5076
Language model weight = 0.5	4.1	0.82	0.6203
Maximum Baye's risk = 0	3.66	0.732	0.5948

Table 1 Results from first part of experiments.

The following MT does not require re-translating from scratch but it needs a lot of editing. It scored 2.

ST: *Vor Ihnen liegen zwei große Hindernisse, und zwar geht es darum, ob wir uns für die gegenseitige Anerkennung oder die Standardisierung entscheiden.*

MT: To two large obstacles, is whether we mutual recognition and the standardisation.

PE: Two large obstacles lie ahead of you. It is about whether we opt for mutual recognition or standardisation.

RT: Two big obstacles lie ahead of you. There is a problem of mutual recognition versus standardisation.

A score of 1 was given to the following MT because the ST had to be translated from scratch. My translation is more literal than the reference translation and *meine Vorredner* is plural.

ST: *Ich möchte meine Vorredner unterstützen.*
MT: To others.
PE: I would like to support the previous speakers.
RT: I would like to second what the previous speaker had to say.

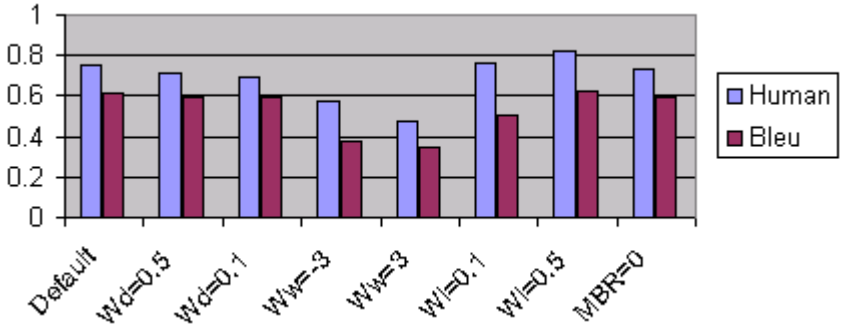


Figure 3 Results from first part of experiments

From table1 and figure 3 we can see that reducing the distortion weights (Wd) reduces the translation quality marginally.

For example the first segment in the test data is

ST: *Ich möchte meine Vorredner unterstützen.*

The MT with default weights is: 'To others'. With the distortion weight at 0.5 it is the same but with the distortion weight reduced to 0.1 it is 'I would support my' which is clearly different.

Varying the word penalty weight had a greater effect but reduced the quality for both increasing and reducing the weighting. Negative values should favour longer output and positive values should prevent short translations.

For the ST sentence *Wir haben dann abgestimmt.* [We then put it to a vote.. (my translation)]

With word penalty weight set to 0, 3 and -3 the MTs were:
We have voted.
We voted.
At the same time, we have to say that we will be able to put to the vote.

Surprisingly reducing the LM weighting increased the MT quality indicating that with the default weights this training favours a poorer translation if it is better English.

The following example illustrates this.

Auch hier möchte ich Sie darauf verweisen, daß das Verfahren beschleunigt werden muß.

[Also here would like I you thereon refer, that the process speeded up become must].

This MT with language model weight =1 has been favoured
'You must realise that the process needs to be accelerated'.

With the language model weight = 0.5 the better translation- 'Also here I would like to remind you that this process needs to be accelerated' is produced. The problem is that *möchte ich Sie darauf verweisen* means 'may I remind you' but it is mis-translated in the corpus as 'you must realise' and given a low probability of being a translation by the phrase table. It is given a higher probability by the LM than the correct translation. Reducing the LM weighting diminishes this effect.

Size of training corpus	Average Human score	Average Human score/5	Bleu score	Average Hter score	Average Hter score/5
200000 segments	2.52	0.504	0.2385	2.4	0.48
400000 segments	2.86	0.572	0.308	2.76	0.552
800000 segments	3.64	0.728	0.31	3.68	0.736
Full corpus	3.77	0.754	0.6129	3.9	0.78
Google	3.76	0.752	0.31	4.085	0.817

Table 2 scores for different sized corpora and GT

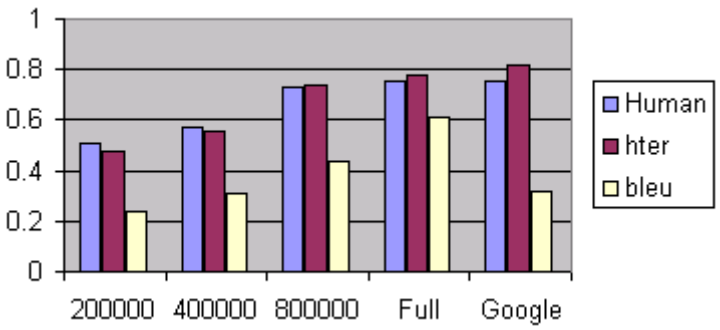


Figure 4 Results in chart form

As expected the quality of the MT output increases with the size of the training data as shown in table 2 and figure 4. The Bleu score trend follows the human and hter scores for the MMM trainings but the Google Bleu score is lower showing agreement with the notion that Bleu scores cannot be used to compare two MT systems with different architectures (Štajner et al, p.595).

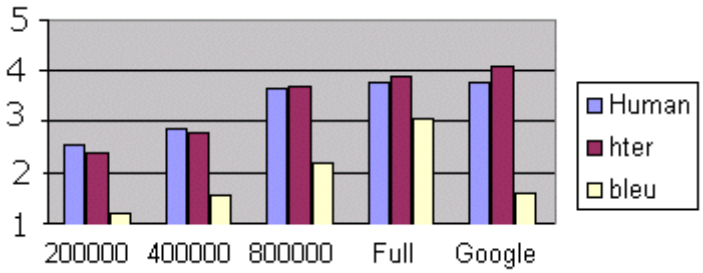


Figure 5 Results on 5 point base

In figure 5 the chart scale has been redrawn to reflect the five-point scale of the human and hter scores. For this 0.2 corresponds to a score of 1 and 0.4 is an average score of 2.

None of the trainings scored 1, the level for which post editing is not worthwhile, but equally none of them crossed the 4 threshold. Robinson (2012, p.38) discusses the use of Google translate to create a first translation draft. This sets a benchmark the equal of which should be the aim of a personal SMT engine.

Looking again at the first segment of the test data.

ST *Ich möchte meine Vorredner unterstützen.*
All of the MTs are poor and I would score them as 1

Size of training	MT
200k	I others.
400k	To others
800k	My previous
Full	To others

Whereas Google's MT I would score 5 following the TAUS guidelines even though strictly 'want to' should be edited to 'would like to'.

Google	I want to support the previous speakers.
PE	I would like to support the previous speakers.
Ref	I would like to second what the previous speaker had to say

In figure 6 the 8.5% improvement achieved for manual scoring with a language model weighting of 0.5 has been applied to all the trainings on the basis that they are the same language and genre. The full corpus gives a score of 0.8 or an average score of more than 4 and 800000 segments are needed to equal GT, which according to Champollion (2007, p.2) represents, for an average translator, producing 50,000 translation units a year, 16 years work. Additionally the freelancer may not have the rights to use this material especially the STs that belong to the author who may withhold permission to include them in the corpus.

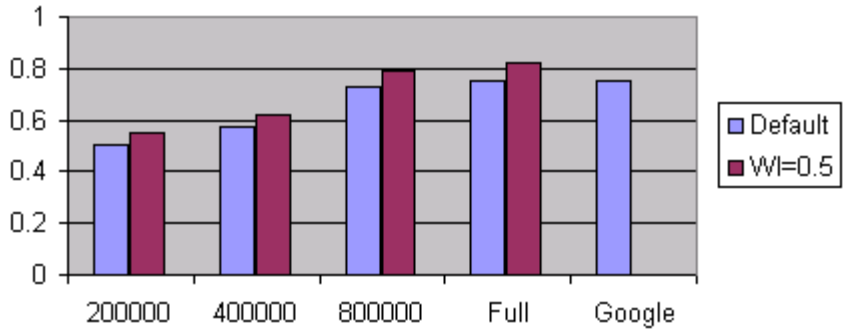


Figure 6 Corrected for Wl=0.5

Conclusion

This case study has successfully shown that it is possible to build a personalised SMT engine with MMM and that the quality of SMT output is directly proportional to the amount of training data available. In this instance for a performance equal to an online MT system the amount of data required was over 75% of the full Europarl corpus of 1.1 million segments. This is very high compared to the amount of material that an individual translator is able to produce, indicating that freelancers may struggle to find enough data to build an adequate system.

In addition to having enough material to build an SMT system with MMM a reasonably high degree of IT ability and knowledge is required or at least an interest in getting involved, even though it is aimed at translators rather than computational linguists and is free. Considering that there is a need to expand the amount of translation capacity available these results are disappointing for freelance translators.

Although this study is only a 'first glance' at using Moses as a personal MT engine it shows that SMT requires the very large amounts of data that are available to online translation engines. It was not carried out by a computational linguist/technologist but by a translator, which is important because now that translators have started to use MT as a tool to quickly produce a first draft, the translation community should take more interest in the development of MT tools. Somehow MT has to embrace and be embraced by TS. My observation from this study is that Pym's vicious circle is rooted in the fundamental techniques of SMT. The vast amount of data needed is far too much to be reasonably checked by humans but Moses generates its probabilities on what it sees in the training data and recycles the errors. Another source of errors observed is caused by the communicative nature of translations and the way that PBSMT relies on word alignment. Finding ways to improve the quality of MT with a limited size bi-text would help to provide post-editing and predictive tools for freelance translators. A first step might be to build an engine with real TM data in a specialised field and conducting experiments. Techniques such as hierarchical phrase tables might then permit data harvested from the Internet to be used but this would be a move away from baseline MMM requiring input from computer scientists.

References

Bowker, Lynne . 2005. Productivity vs Quality? A Pilot Study on the Impact of Translation Memory Systems. In *Localisation Focus* 4(1): 13–20

Brown, Peter. Cocke, John. Della Pietra, Stephen. Della Pietra, Vincent. Jelinek, Frederik. Mercer, Robert and Roossin, Paul. 1988. "A statistical approach to language translation." In *Proceedings, 12th International Conference on Computational Linguistics (COLING-88).*Budapest, Hungary, pp. 71-76.

Callison-Burch, Chris. Osborne, Miles. and Koehn, Philipp. 2006. Re-evaluating the role of Bleu in machine translation research. In *Proceedings EACL*, pp. 249-256 Retrieved from http://www.citeulike.org/user/JeremyKahn/article/2945969

Carson-Berndsen,Julie, Somers,harold, Vogel, Carl and Way, Andy.2010.Integrated language technology as part of the next generation of localisation in *The international journal of localisation* 8(1).

Champollion,Yves. 2007.The free, universal TM: are idealism and pragmatism compatible*? Technical seminar on copyright, intelectual property and translation tools Barcelone*.Retrieved from http://www.fit-europe.org/vault/barcelone/Champollion.pdf

Coughlin, Deborah.2003.Correlating automated and human assessments of machine translation quality. In *Proceedings of MT Summit IX*

Doddington,G. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. *In Human language technology : Notebook proceedings*, pp.128-132 San Diego

Garcia, Ignatius. 2011. Translating by post-editing: is it the way forward? in *Machine translation* vol 25 pp.217-237

Google. 2014.*Google terms of service.* Retrieved from https://www.google.com/policies/terms/

Klabunde, Achim. 2014.Cybersecurity in the era of freely available machine translation service in internet. Paper presented at MT@Work - Public Service Redesigned? Retrieved from https://scic.ec.europa.eu/streaming/index.php?es=2&sessionno=f4661398cb1a3abd3ffe58600bf11322v

Koehn,Philipp. 2005. *Europarl: a parallel corpus for statistical machine translation.*MT summit 2005. Retrieved from http://homepages.inf.ed.ac.uk/pkoehn/publications/europarl.pdf

Koehn,Philipp. 2010. *Statistical machine translation*. Cambridge: Cambridge University Press

Koehn,Philipp. 2015. *Moses statistical machine translation system user manual and code guide.* Retrieved from http://www.statmt.org/moses/manual/manual.pdf

Machado, Maria and Fontes, Hilario. 2011. *Machine translation: case study – English into Portuguese — evaluation of Moses in dgt Portuguese language department using Moses for mere mortals* Retrieved from http://ec.europa.eu/translation/portuguese/magazine/documents/folha37_moses_en.pdf

Machado, Maria and Fontes, Hilario. 2014.*Moses for Mere Mortals tutorial: A machine translation chain for the real world.* Retrieved from https://github.com/jladcr/Moses-for- Mere-Mortals/blob/dfbfe799ebee1e1e0a3fa370fb4c6050511d5b0c/Tutorial.pdf

McElhaney,Terrence And Vasconellos,Muriel. 1988. The translator and the postediting experience. In Vasconellos,M (ed). *Technology as translation strategy.* Amsterdam: John Benjamins

Papineni, Kishore. Roukos, Salim. Ward, Todd and Zhu, Wei-Jing. 2001. *Bleu: a Method for automatic evaluation of machine translation IBM Research Report* Retrieved from http://www1.cs.columbia.edu/nlp/sgd/bleu.pdf

Pym.Anthony. 2010.*Exploring translation theories*, Abingdon: Routledge

Pym.Anthony. 2012. Translation skill-sets in a machine-translation age *Journal des traducteurs / Translators' Journa*l Volume 58(3) , pages 487-503.

Robinson, Douglas. 2012.*Becoming a translator.* 3rd ed London: Routledge

Snover, Matthew, Dorr,Bonnie.Schwartz,Richard. Micciulla,Linea and Makhoul.John. 2006. A study of translation edit rate with targeted human n-notation. In *Proceedings of association for machine translation in the americas* Retrieved from https://www.cs.umd.edu/~snover/pub/wsmt09/terp_wsmt09.pdf

Štajner, Sanja, Querido, Andreia, Rendeiro, Nuno, Rodrigues, Jo˜ao and Branco, António. 2016. Use of Domain-Specific Language Resources in Machine Translation in *Proceedings of the International Conference on Language Resources and Evaluation (LREC) 2016*. Portoroz, Slovenia, May 25-27

Taus. 2010. *MT post-editing guidelines* retrieved from https://www.taus.net/academy/best-practices/postedit-best-practices/machine-translation-post-editing-guidelines

White, John. 2003. How to evaluate machine translation. In Somers, Harold.(ed). *Computers and translation. A translator's guide*. Philadelphia, PA, USA: John Benjamins Publishing Company

The Annotation System

Ronan Martin
SAS Institute
ronan.martin@sas.com

Abstract

Two of the main challenges of translation are comprehension and terminology: understanding the text, and knowing what to call things in the target language. The focus of this topic is the comprehension component, not so much "how to help translators in understanding what is meant by a text string", but more, "how to deploy the solution to translation queries so that all translators have access to the solutions when they highlight a string in the translation (CAT) tool". The CAT tools we use do have partial solutions, but we did not find these viable for different reasons. We needed a way of annotating source files once and for all. However, it was vital that we did not leave a footprint in our source files. Any footprint would lead to a breakdown at build (compilation) time.

The challenge: How do you create an external annotation that will always find its target string in the source files. We opted for a methodology borrowed from the terminology paradigm. Our source string was like a term, and the annotation like a term comment. The termbank became a stringbank, and the term dictionary an annotation dictionary.

1 Introduction

An easily understood text can be translated quite quickly. However, the more technical the content, the more challenging the task becomes. For an experienced translator, the main challenge is one of comprehension, and this can be due to terminology, lack of context, ambiguity, difficult syntax, or a host of other reasons.

Terminology is a domain that is widely recognized as being central to any translation process and the preferred goal in most cases is to try and deal with difficult terminology in a proactive way. All the other things that make a text difficult to comprehend, however, tend to be dealt with on an ad hoc basis. For example, you arrive at a sentence or a text string and discover that it doesn't readily render a translatable meaning and it is then that you set in motion queries which move upstream back to the author of the text, or to somebody who is likely to know what the text is trying to express.

So terminology is dealt with proactively and centrally and resolved terminology moves downstream to translators via interactive dictionaries. Other translation queries tend to be dealt with retroactively, with queries moving upstream from each translator back to the source. Resolutions to queries can be published or distributed, but generally there is no way to integrate them with CAT tools in the same interactive way that you can with terminology.

The Annotation System is a method we are developing to try and address translation challenges that are not specifically due to terminology. We are hoping that it will be possible to make the task of solving comprehension issues much more streamlined and proactive than it is at present.

2 Old Problem - New Idea - Design - Implementation

2.1 Background

At ELC (the European Localization Center at SAS) we localize SAS software solutions. SAS is a large software company (the largest privately owned software company in the world). Most SAS products are extensive software suites aimed at corporate customers and relate to

Proceedings of the 38th Conference Translating and the Computer, pages 113–121,
London, UK, November 17-18, 2016.

the application of analytics to, or statistical analysis on large volumes of data. Many products are tailored to a specific domain and can encompass advanced technical concepts. There are two principal components of the typical localization task: localizing the user interface (UI) and localizing support documentation. By far the most challenging task for translators is localizing UI files.

When UI text strings appear in source files they are *de facto* decontextualized. Their natural context is the interface in which they will appear, but for now they reside in properties files as string lists and are only retrieved at runtime. Translators have to translate these lists which run up to volumes of 10, 50, sometimes 150 thousand words.

By contrast the user documentation is more verbose and coherent. When authoring documentation, technical writers provide a rich linguistic context for the reader to follow, and translators benefit from this.

UI strings are written by developers. Developers are not renowned for being great linguists. In addition, some development may take place overseas by developers whose mother tongue is not the same as the source language they are writing strings in. A comprehensive review of linguistic quality is difficult, given the typical kind of development environment. Groups of strings can be altered or removed one day, only to be added back the next day. Components are moved around, re-written, revised and recoded right up until the point of code lockdown, when the release phase starts. Time to market is pressed by competitors releasing similar products and trying to get these out there first. Linguistic reviewers tend to get in the way of rapid development and it isn't here where a great number of resources are generally employed. Thus translators inherit lists of strings that are of mixed quality. There are cryptic acronyms and abbreviations, developer speak, camel-case words, typos, sequences of words written in telegraph style. In addition, many strings hold just one or two words, the reason for the brevity being that the text is destined for a field in a dialog pane or screen with limited space.

2.2 Current Querying Practices

As far as possible localization project managers try and provide background information to translators about the strings like existing technical papers and screen shots, and the documentation itself if this has been completed. Dictionaries are also provided that target terminology and chunk-sized phrases. But there are still many strings which need to be queried by the translator. These queries are routed back to the developer responsible for authoring the strings. A query is often a longer thread than just a question-answer pair. The developer doesn't always understand what it is about the string that is a challenge, or they may need to re-route the query to another expert.

At some point the query is resolved. Quite often the string is removed or re-written, but there are many resolutions where the developer provides important contextual information, or explains the terminology or acronym or strange wording. Once the solution is received, the translator can translate the string appropriately and continue with the work.

Now, consider that we translate to 30 languages. Each translator needs to be aware of queries that are currently in progress, pending answers, or that have been resolved. Otherwise they will end up sending the same query again to developers.

2.3 Possible Solutions

One possible solution is to somehow annotate the files. The developer or someone else could enter comments into the properties files. This is doable, and developers sometimes do this, but there are drawbacks. One is that the developer cannot know up-front, which strings are going to cause translators problems. And by the time translators get to the file, developers are usually close to signing off on them. Secondly, we need to imagine that when translators are

working quickly, comments are not always displayed in a timely manner. They can fall outside the small display window as translators jump from a string to the next translatable string. Sometimes translators use subset views and the comments become detached from the string they are referring to.

As regards translators annotating the files, this is too risky. Any editing of source files bears a risk of introducing code errors and only a developer or someone working close to the developer should do this.

How about using mechanisms available in the CAT tool for storing annotations, for example in the TM? Here there are also a few drawbacks which reduce the viability of this approach. It is possible to enter notes for individual segments in the CAT tool. These notes are generally saved together with the segment. However, the TM is language-specific. Therefore the note does not reach other translators. Also, when TM is applied, the translation from a previous version may be suggested, but the note is not automatically displayed. Translators need to actively open the TM segment in order to view most metadata relating to it.

These considerations led us to wonder about whether it was possible to associate the query-solution text with a given segment in some other way. The problem had existed for many years and we seemed no closer to a solution apart from advising translators to track all queries and to try and implement them if they are relevant for the product they are currently sitting translating.

2.4 Analogies with Terminology

In the following it is important that you understand the dictionary mechanism of CAT tools. When you give focus to a translation segment (usually a sentence or text string), any relevant terminology for that string is displayed in a terminology window. The terminology content changes as you jump from segment to segment.

Quite a few of the queries had been routed to me, the terminology manager, over the years as terminology queries. A few were bone fide source-terminology queries and I could place the solution in the dictionary so that it was displayed whenever translators came to it. Others were clearly not terminology queries but were similar in some ways. A problem with terminology is a special case of a general comprehension problem. Perhaps we could place the solution to these non-terminology queries in a dictionary as well. But then it occurred to us that these kinds of queries related to a specific instance of a string, not every occurrence as is the case for terminology.

However, if you extend the length of the "term" beyond the boundaries of the word or phrase being asked about, what were the chances that your sequence of characters was unique within the corpus for a given project. Investigations showed me that you didn't have to include a very long string before you had a unique identifier for that string - and in essence, the identifier could be the string itself. If you include this string in a dictionary, it will display an entry for that segment only. Instead of a target term you could just write a text that explained to translators how to interpret the string in order to translate it.

This meant that we could annotate a document, and keep the annotations separately. The link would be a string reference which was the string, or a subset of the string itself. On the basis of a few tests we decided to pursue this option.

Taking the analogy of terminology a bit further, it seemed what we needed was a stringbank which would be the storage repository for annotations. The dictionary would be the delivery mechanism. Currently we already generate two kinds of xml-based term dictionaries for translators to use with a project. The annotation dictionary would just be a third one of these.

2.5 Components

We had envisaged a storage method and a delivery mechanism. Now we needed to set up an environment that supported a workflow where:

- queries were processed and resolved
- resolutions were entered in an annotation system (repository)
- an editing interface was provided
- an annotation-dictionary build process was made available
- there was an annotation selection mechanism that only returned relevant annotations for a given project

We already had a good query system which was just being set up and overhauled, and this provided a good opportunity for incorporating interaction between the query system and the new annotation system.

We wanted the whole process to be devolved rather than centrally steered, with self-service features wherever possible.

2.6 Processing and Resolution of Queries

The purpose here is not to describe the query system, which in many ways is a standard type application. It is mail based. When a translator runs into problems, they can enter a question into the query system. Some values need to be set indicating which project the query relates to, where the string comes from etc. After this the query is submitted. A "ticket" is set up, and mails are sent to interested parties. In this case we predefine interested parties to include the developers who have written the code.

A mail thread ensues between the translator and the developer. This may consist of two mails, question/answer, or can extend to as many as a dozen exchanges. Eventually the translator feels that they have received a satisfactory answer and the ticket is closed.

Previously we requested that all in-house translators follow all mail threads and monitor them for relevance. In the case of outsourced projects, the localization project manager for a project usually had to meticulously compile a list of resolved queries that related to a project, and send this list when they handed off a project to a language vendor.

2.7 Entering Query Solutions into the Annotation System

As we were in the process of implementing a homegrown query system, this gave us some freedom in designing some interfaces. For example, when a query is created it is possible to take some basic data and metadata and bundle this into a set of properties/values that can later be passed to the annotation system. The vehicle we chose was to add the property/value list to a URL that opened the Add View of the Annotation System. Something like this:

http://koelcterm.sdk.sas.com/add_string_info.htm?string=Remediation%%2ALead%%2A&product=\vertical\Products\RiskAndCompliance\Monitor\Source\mrm\Config\Deployment\Content\Preload\Config::d4grc62&properties_key=linkType.MRM.finding_remediationLead.name1.txt%3D&scope=MRM&author=spnmxc

It looks cumbersome, but this hyperlink was neatly embedded into the query ticket. When the query is resolved, the person who is in charge of the query (generally the person who

opened it), clicks on a pre-agreed word and this opens up the Annotation System on the "Add View" page.

Below is an example of the top of a query (which grows downwards as comments are added). I have highlighted the word String. This is the pre-agreed word that translators can click on to open the Annotation System web page and enter a new annotation. The hyperlink behind this word is the one above.

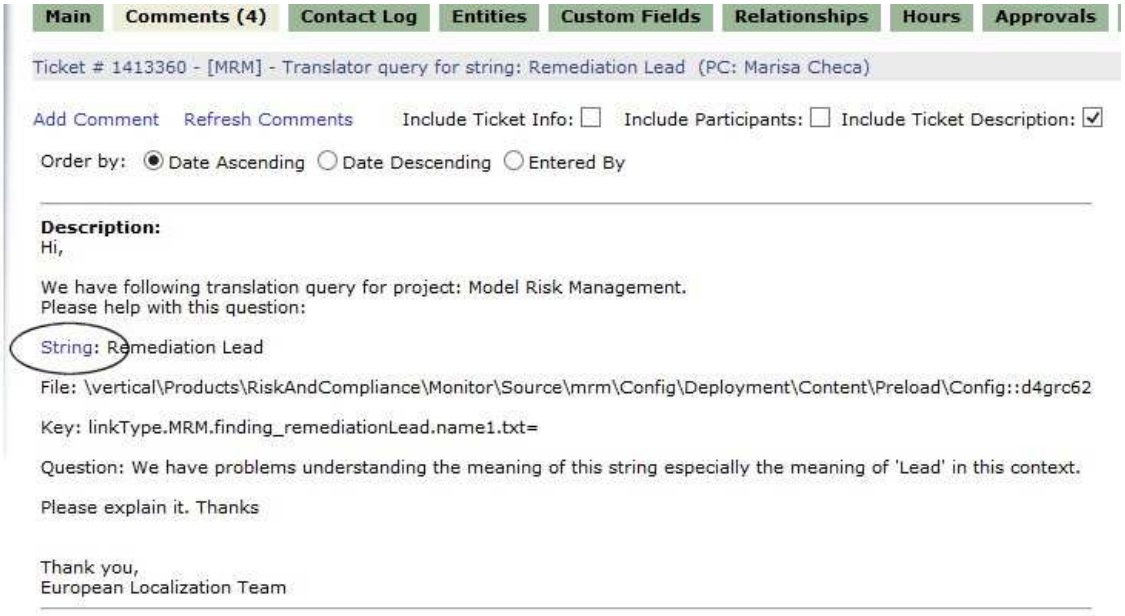


Figure 1: Ticket in the Query System

The Add View of the Annotation System is a simple web page containing a form. Thus, from the query it is possible to click on the URL, have the form displayed, and have many of the fields pre-populated. Javascript in the receiving page unwraps the parameters that are passed with the URL and inserts these in the relevant fields. Fields will display the troublesome string itself, the project it relates to, the key associated with the string in the source file (for UI properties files most strings appear as key/value pairs, where the value is the string), the sender etc.

Here is an example:

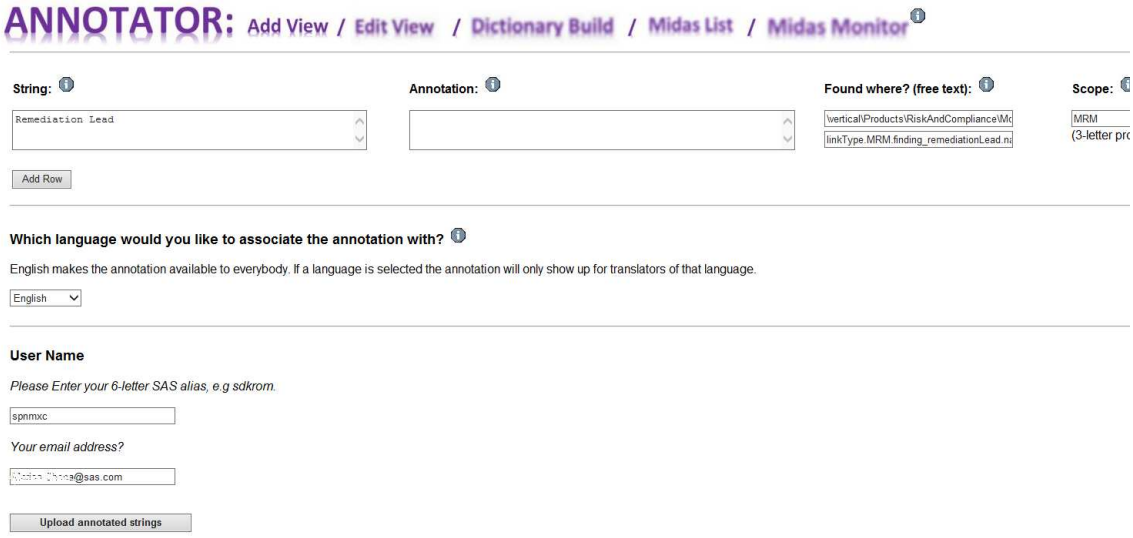


Figure 2: Example of the Add View

The only thing that isn't included is the resolution itself which must be placed in the Annotation field manually. Usually this can be copy/pasted from the mail thread for the relevant query. All of these things can be edited before submitting the form. When submitted, a new string/annotation entry is set up in the Annotation System repository.

2.8 Editing Interface

When a repository of this kind is being built up, it is important to give contributors the opportunity to edit their entries. The Edit View displays a list of existing annotations, reflecting the contents of the annotation repository. The string and the annotation are editable, and can be uploaded once more to overwrite the existing previous entries.

The interface contains various types of filter and subsetting so translators can locate the precise query they are looking for.

2.9 Dictionary Build

The whole point of the Annotation System is that a list of annotations relating to the source files for a localization project can be attached to the project, and thus warn or advise translators about certain segments when they bring these into focus in the CAT tool.

Looking ahead, I was concerned about the number of annotation entries that may build up over time. I didn't want to distribute the whole list every time a project was started. In fact, I was faced with a number of questions. How often should annotation dictionaries be built, how could I limit their size.

The Annotation System and the query system change from day to day. Queries only start to roll in once a localization project has been launched. Thus, it wasn't enough to be content with creating an annotation dictionary when you sat down to start localizing a project. After a few days many resolutions may have resulted in new annotations. The dictionary had to be small, and of a throw-away variety, because translators may have to apply the dictionary several times during the localization, and probably also at the end.

The dictionary build interface is simple enough. It is possible to build an annotation dictionary for any project which has annotations associated with it.

2.10 Ensuring only Relevant Strings are Returned

What goes on behind the scenes is a little more complex. When a dictionary is requested I need to ensure that any annotation associated with it, does in fact relate to a string that is still part of the project. Strings become removed from projects, and I didn't want the size of the dictionary to be swollen with obsolete strings. For this reason the build script needs to retrieve the very latest version of the source files and perform a concordance search between the source strings and the annotated strings in the repository.

Once created the dictionary is zipped and made available through an ftp link, and can be downloaded and attached to a project.

3 What we discovered along the way...

3.1 Embedded Dictionary Entries Hidden ☹☹

We are to an extent restrained by our CAT tool. We have a clear idea about what we would like to achieve, but this has to be squeezed through the functionality that is available in our CAT tools. As we are using a terminology tool to surface the annotations, certain challenges arise.

The first major challenge is that our current tool gives precedence to the longest match, when comparing dictionary entries and the source segment. If there is a match for both a single-word term, and a multi-word term, and the one-word term is embedded in the longer term, then only the multi-word term entry is displayed in the terminology pane. The other entry is hidden away. This means that an annotation entry, which is like a long term, always hides any term entries that may occur in the same string.

To date we have no fix for this, but we are in the process of changing CAT tools and this phenomenon does not occur in our new CAT tool. As a workaround we have advised translators to have dedicated sessions where they work with the annotated entries, and otherwise only attach term dictionaries.

Interestingly, I discovered straight away that the dictionary only displayed entries if there was a target term. Initially I placed the annotation in the comments field for the dictionary entry and left the target term field blank. This didn't work and instead I entered a "-" as a kind of dummy target for all languages. This was enough to get the record displayed when you gave focus to a segment.

Because of this it is possible to run a Terminology Check, which quickly runs through all the segments and checks to see whether the target term has been applied to segments where the annotation matches all or part of the source string. If it finds a target hasn't been applied, it stops and prompts the translator. Thus it stops at all segments that have an annotation entry. Here translators can just check whether they have heeded the information, then jump to the next segment where there is a hit.

At any rate, our collection and storage mechanism is tool-independent. It is only at the delivery level we have this reliance. This leaves us with possibilities of overcoming many of these problems in creative ways as we move forward.

3.2 Language-Specific Entries ☺

Originally the system was only intended for general annotations. However, once we started up, we noticed that some query/annotation workflows were language-specific. The query system allows translators at external vendors to open queries which are sent to our in-house translators, or in-house experts. For example, which of two translation targets is preferable. These types of query relate to one language only and the resolution is very important, but only relevant for that language.

Luckily the possibility of dealing with language-specific annotations was available to us. The dictionary format we use when creating dictionaries is MARTIF (a predecessor of TBX). What I decided to do was to place the source string where the source term would have been, have the annotation as an entry-level comment, and any language-specific annotations as the target term.

Dictionaries are still multi-lingual so one size fits all for one particular project. However, when translators work with files, they will have their target language selected, and they will only see target entries for this language. Everyone will see the general annotations, but translators will only see the language-specific annotations for their language.

This mechanism opened up the possibility of storing language specific notes and annotations that weren't necessarily the resolutions to queries, but just the result of a decision made while translating.

3.3 Short and Common-Word Entries ☹

It quickly became apparent that there were problems with a certain type of string. For example, things like "Current", "Home", "Issue", "none". Queries had been entered for strings like this. They are complete segments because in the source files, there is a discrete string with this single word. If I allowed these strings to go into the dictionary they would create false positive hits in very many segments, and these annotations nearly always related to one specific segment only.

My first thought was to investigate whether the dictionary match could be made dependent on the properties key. For example, this is what the string may look like in the source file:

```
rules.ruleSasTest.js.none.txt=None;
```

The key is on the left-hand side of the equals sign. In our translation windows, the key is protected and cannot be edited. The dictionary entry only matches the editable part of segments. So, the possibility of using the key was not open to us.

In the end I had to devise an alternative workaround. Instead of writing the strings/annotations to the MARTIF file only I also wrote them to a CSV file that was packaged together with the dictionary. Here I could include the problem string together with the key and the annotation, like this:

None	"None" means the used variables value is an empty string.	rules.ruleSasTest.js.none.txt	Only printed here - not in dictionary

Translators were asked to make sure and check these entries as well, which wouldn't automatically be displayed to them while translating.

3.4 Annotations Up-Front ☺☺

As an extra bonus we now have the possibility of being truly pro-active. In cases it is possible for a reviewer to work with developers and try and improve the quality of their strings. The reviewer has a very small window in which to operate. Changes are recommended for some strings, while there is not time enough to change others. The reviewer can also query the developer about what kind of context the string will be displayed in. Any explanatory information about strings is now added to the Annotation System. Thus when translators receive the files and view the string in question, they now have vital information that will help them translate it correctly and quickly. And this may be strings that are handed off to translators in up to 30 locations simultaneously.

4 How do things look at present...?

4.1 Savings

We feel pretty sure that this system is providing us with savings on several fronts. The savings are difficult to document - we are in the process of gathering data so that we can analyze some of the effects. The system has been in operation for about six months and we have been slowly rolling it in, becoming fully operational about 3-4 months ago. At this point we have 775 annotation entries and this grows at a linear pace.

The main saving lies in the fact that only one translator sends a query, and enters the annotation in the system if this is appropriate. The other translators reap the benefit of this, and because of the dictionary mechanism they can be sure that they will not inadvertently fail to implement the solution.

Translators have a safe mechanism for storing annotations about strings they translate. It may be important in future to remember why you chose a particular target string, or perhaps it will be another translator reviewing the string in future.

Overall quality probably improves. Tricky issues in the form of ambiguous source strings are less likely to result in a bad translation. It can be easy to misinterpret many common words like "issue" (issue shares, or is issue just a problem?), "apply" (for a license, or apply a strategy?) , "policy" (company policy or insurance policy?). If just one translator notices the ambiguity, queries it and gets a solution, all translators will be made aware of it.

In-house translators are more likely to query company-specific concepts, but external translators may not even know that the concept differs from the normal usage of the word(s). Our in-house translators always get the source files first. We can now be sure that their discoveries will be passed on to translators of languages we normally outsource.

And last but not least, often we "roll in" a new language which starts localizing the product from scratch. Here the benefits will be considerable as we will be able to hand off to language vendors a fully "annotated" set of files, saving many time-consuming queries for our PMs or developers.

4.2 Some Closing Thoughts

The system needs to be scalable - time will tell whether we will need to introduce changes when the volume increases. The repository will grow in size, and at some point we may need to start tracking whether certain strings have gone away for ever. However, the dictionaries should remain at a stable size, perhaps growing slowly into an optimum size for each product over time.

I would like to see translators making more use of the language-specific annotation possibilities. This entails being able to share with others various reasons for choosing certain types of translation targets over others. This is especially important when there is a collaboration between in-house and external translators, but could be important if localization projects change hands between translators within the same language.

I've been pleasantly surprised with the involvement that our translators have shown in this new system. I couldn't really say the same for our Terminology Management System in the early days, where it took some time to get people on board. I think the degree of involvement is a measure of the relevance of the system for the daily work of translators.

What's in a Name?

Jon D Riding
United Bible Societies
Stonehill Green, Westlea,
Swindon SN5 7TJ
jonriding@biblesocieties.org

Neil J Boulton
United Bible Societies
Stonehill Green, Westlea,
Swindon SN5 7TJ
neilboulton@biblesocieties.org

Abstract

This paper describes the development of a language independent process for identifying proper-names in a text. The process is derived from a machine originally designed to analyse non-concatenative morphologies in natural languages. The particular context for this work is the task of managing the 5,000 or so proper-names found in a Bible, including the identification of close cognates and reporting instances where a related form does not appear to be present.

The need for such a system is explained and the process by which the machine is able to identify names in the target text is described. The problems posed by disparate orthographies are noted. Results obtained from Eurasian, South American and African languages are discussed, common problems for the process identified and its possible use in the context of technical vocabulary suggested. Commonalities between the task of identifying morphology templates, ordered phoneme sets and syntax patterns are noted.

1 Introduction

The Bible translator faces a peculiar set of problems. Some are common to all translation but others are particular to the task of translating a Bible. In this paper we shall explore a problem particular although not unique to Bible translation and describe how NLP techniques might be employed to limit the scale of the difficulties it causes for translators. The issue is that of managing the transliteration of proper-names found in the Bible.

At first sight this might seem a strange focus for NLP research but there are some characteristics of a Bible translation project which make it far from trivial. To begin with, the Bible is a large book. In its largest form (not all Christian denominations read all of it) it comprises 81 books whose origins can be traced back something like 3,500 years. Some are histories, others are collections of poetry or wise sayings, some are telling critiques of the politics of the times and others record personal and formal correspondence between churches and friends. As a collection it is vast and eclectic. All of these genres have in common the use of narrative to expound their message and all good stories are, in the end, about people. All of these people, their nations, tribes, cities and families have names.

The Bible contains references to between 4,500 and 5,000 names. The precise number depends on whether categories such as tribal names, names of regions etc... are included. All these names are 2,000 years old or more and as such they arise from cultures where a name was of particular significance. Not only did a name seek to express character it could also be regarded as formative of character [Blumenthal, 2009]. Biblical names, therefore, add both dimension and depth to a narrative [Krasovec, 2010]. Given this, it becomes very important that they are handled consistently by a translation team.

The translation team also brings its own problems. A translation of the Bible rarely completes in fewer than ten years. Fifteen years or more is not uncommon. Over such a period the membership of the team may well change and the team's expertise will develop as the project progresses. For all these reasons and more it is not hard to imagine the importance of handling

Proceedings of the 38th Conference Translating and the Computer, pages 122–132,
London, UK, November 17-18, 2016.

names consistently and the difficulties a team may encounter in ensuring that this is applied throughout the text.
Unlike the bulk of a text, proper-names are rarely translated [Auden, 1970], it is more usual that they are transliterated[1]. Transliteration involves recasting the phoneme stream which represents the model name in the base text into an appropriate form for the target language. Sometimes entire translations undergo this process when more than one orthography is used for a language (as in Romanian/Moldovan prior to 1989 [Chinn, 1993]). The transliteration task is the same for any given pair of model and target languages, to find a way of representing in the target language an ordered set of phonemes from the model language. Other changes may also take place where languages are more highly inflected. Our task has been to find a way of identifying the thousands of proper-names in a new translation of the Bible so that the translators can more easily review their work and ensure they have rendered names consistently throughout the text.

2 Components of Proper Names

Phonemes

Those of us accustomed to working with language as text tend to think of words as made up of letters. We recognise that particular surface forms are constructed from stem lemmata via morphological transformations but our day to day encounters with language as text encourage us to imagine that the fundamental building blocks are letters. This is both helpful and misleading. It is helpful insofar as letters map to typescript and so to the keyboards which are the medium through which we create text, it is misleading insofar as letters are only an approximation of the phonemic components which underpin spoken language. The system that truly underpins language is fundamentally about speech.
A proper-name (PN) is in reality a row of phonemes[2] which represent the sounds of that PN as an utterance. Many of these phonemes will correspond to letters in the alphabet but others are represented by letter clusters. English, for example, has a number of phonemes which cannot be represented by a single letter such as {*ch, ph, sh, th*} and it can be argued that some English vowels too may be represented by letter clusters e.g. {*ai, ea, ee, etc…*}, the list is incomplete. It is not our intention to argue the case for a defined and generally agreed phoneme set for English or any other language but rather to suggest that expanding our set of symbols beyond the alphabet to include common letter clusters which are used consistently to represent particular sounds is beneficial to the task of identifying PNs in text.

Order

If phonemes may be considered the building blocks of names there is a second element of equal importance. Order is critical in rendering and recognising names. Consider ‘David’. We might parse David into five components {D.a.v.i.d} but without taking into account that these components are ordered we get little benefit. ‘additive’, for instance, has all of the components found in ‘David’ but there is no relationship between the two words.

A PN then, is constructed from letters and letter clusters each of which may be taken to represent a phoneme and all of which are ordered to form the PN as a whole. A PN is, in other words, an ordered set of phoneme items.

---

1 There are occasional exceptions where names carry root meanings which are central to the narrative and translators decide to translate e.g. לֹא עַמִּי (Lo-Ammi) [Kittel, 1997] – Not-My-People [GNB, 1976] – Hos 1.9.

2 In this paper we use the word phoneme in a less than formal sense to describe an alphabetic character or cluster of characters that consistently represents a particular sound in a language. For a more formal discussion of phonemes see: [Davenport & Hannahs, 2010:115-132] or [Ladefoged, 2001:23-24].

3 Finding a name in text

Since PNs do not typically translate but are reproduced as a phonemic pattern in the translated text it follows that finding a PN in a text is an exercise in identifying the phonemic components of the PN ordered as we might expect them to be. Thus if we begin with a model representation of a name such as 'David' we might hope to find a similarly ordered set of items representing those sounds in another language. One of the early questions a Bible translation team must address is what they consider their 'model' to be for names. There is, rightly, an emphasis in Bible translation on returning to the Hebrew and Greek base texts but this is not always helpful. The reality on the ground may well be that the people group for whom the new translation is being prepared have had access to other translations of the Bible in regional *lingua franca*. Thus, for example, a Swahili translation may well choose to transliterate the name *Abraham* as *Ibrahimu*[3] rather than the more naturally Swahili *Abrahamu*. Such decisions recognise the need to work with existing expectations within the community rather than seeking to impose a purist view.

Where a team elects to transliterate directly from the base texts we must first render the model name into the same alphabet as the target text. This is done via a transliteration matrix which implements a simple Markov Chain[4] to record the probability of a particular base language phoneme being represented as a certain target language phoneme. Standard transliteration matrices exist for many pairs of scripts which can be adapted for particular language pairs. As the system identifies pairs of model and target names the individual transliterations these pairings represent are fed back into the matrix to improve subsequent transliterations.

Where our model and target texts share an alphabet we might hypothesise that *David* and *Daavidille* (Finnish, allative [Karlsson, 1999:119]) are references to the same PN by virtue of the two renderings sharing a common sequence of phonemes {D.a._.v.i.d_}. The order of these matched items is crucial to the identification as too is the proximity of each successive matched item to its predecessor. We should also note that we have preserved the leading capital in our examples so far but in practice many languages do not use leading capitals with PNs. In reality our matched set of phonemes needs to be {d.a._.v.i.d_} (we use '_' to represent lacunae covering one or more items).

The method adopted to assess how closely a word in a translation matches a model form of a PN is an adaptation of a process presented at TC34 [Riding, 2012]. The process compares candidates from the target text with a model form by ranging the model and candidate along the two axes of a simple matrix with the candidate on the *x* axis and the model on the *y* axis.

Thus in our example above, *David*/*Daavidille*, we generate the matrix:

		1	2	3	4	5	6	7	8	9	10	
	∅	d	a	a	v	i	d	i	l	l	e	
1	d	**d**					**d**					
2	a		**a**	**a**								
3	v				**v**							
4	i					**i**		**i**				
5	d	**d**					**d**					
												∅

Fig 1.

3 Swahili Union Bible [UBS, 1989]. In East Africa the influence of Arabic is strong, particularly closer to the Indian Ocean coast as a consequence of centuries of trade with Arab merchants.

4 For a full discussion of Markov Chains see [Puterman, 2005]

Taking origin at top left we now construct all the possible routes across the matrix via the matched items taking due account of order by following the rule that a successor must have x and y coordinates which are both greater than its predecessor. This results in six possible solutions:

S1 {d(1,1), a(2,2), v(4,3), i(5,4), d(6,5)}
S2 {d(1,1), a(2,2), v(4,3), i(7,4)}
S3 {d(1,1), a(3,2), v(4,3), i(5,4), d(6,5)}
S4 {d(1,1), a(2,2), v(4,3), i(7,4)}
S5 {d(6,1), i(7,4)}
S6 {d(1,5)}

All of these sequences are possible solutions to our problem. Clearly, we need a way to assess their relative merits. We do this by noting the degree of proximity for each pair of matched items. In S1 we have a set of four matched pairs:

$$\{(\text{d(1,1), a(2,2)}), (\text{a(2,2), v(4,3)}), (\text{v(4,3), i(5,4)}), (\text{i(5,4), d(6,5)})\}$$

Reading across the x coordinates we discover proximities for each of pair of 1, 2, 1 and 1. This measure of distance between each pair allows us to construct a score which represents the closeness with which the sequence matches the model. Given that we can calculate the value of a perfect match we express the value of this sequence as a probability with respect to a perfect match value. A more detailed description of the process is given at Appendix A.

4 In practice

This processing is now in beta test as part of the UBS CogNomen (CGN) system. CGN sits beside the translators' editor, ParaTExt, and reviews the new text as it is created. Tables exist which identify locations in the text where particular PNs might be present and using these CGN loads the model forms of those names from a model names list. Once a user has finished editing a verse CGN runs in the background and identifies candidates for each expected name storing them, together with their score, in a growing table of PNs for the new text. Names form part of a wider list of key terms which are carefully reviewed as a translation proceeds. At such reviews CGN provides a list of all names found in the pericopes under review, allowing the translator to approve renderings or not as they choose. Over time the translation team will identify a threshold value above which CGN's suggestions can be accepted with confidence without the need for detailed review.

As the translation approaches completion CGN's tables of PNs allow the team to focus on areas where review is needed, greatly reducing the scale of the task. Additional benefits are to segregate PNs from morphology based spelling checks in which context they represent little more than noise. It has been estimated that if CGN is used consistently throughout the lifetime of a project a saving of up to 5% of the time needed for the translation can be made.

5 Wider application

The application described here is particular to Bible translation but we believe that there are wider contexts for this processing. We know already that the process can automatically analyse complex discontinuous morphologies [Riding, 2012] and experiments suggest that progress might also be made in the analysis of syntax structures using a similar technique. Given that at word level this process represents a way to identify cognate forms which share a similar phonemic pattern we also hope to explore the automatic creation of technical indexes and glossaries.

6 Conclusion

The process described in this paper is able to analyse linguistic structures in a number of different and, at first sight, apparently unrelated contexts. That a single analysis process can be used in the context of phonemic and morphemic word formation, and possibly in syntax analysis, is telling us something significant about language. In every context this process relies upon identifying not only individual items within a text stream but also the order in which they are encountered. The moment we write something down we have removed it from the event stream of time. A document is not really a moment in time but a recording of a linguistic event stream conveying meaning not only in its individual components but by the order in which we encounter them.
CogNomen represents just one context in which stream based processing can help translators manage elements of large texts in an efficient and consistent manner.

Acknowledgements

We are indebted to United Bible Societies with whose support this work has been funded by Every Tribe and Every Nation (ETEN). Thanks are also due to Oxford Brookes University for continued access to their computing and library services.

References

Auden, W H (1970) A Certain World, The Viking Press, New York.
Bible Society eds. (1976) Good News Bible. BFBS
Blumenthal, F (2009) Biblical Onomastics: What's in a name? *Jewish Bible Quarterly Vol.* 37(2) 124-128
Chinn, J (1993) "The Politics of Language in Moldova", *Demokratizatsya* 1993 Vol. 2(2) pp. 309–315
Davenport, M and Hannahs, S J (2010) Introducing Phonetics and Phonology, 3rd Ed. Routledge.
Davidson, B (1981) The Analytical Hebrew and Chaldee Lexicon, 3rd Ed. Bagster, London.
Karlsson, F (1999) Finnish: An essential grammar, Routledge, London.
Kittel, R. (Ed.) (1997) Biblia Hebraica Stuttgartensia, Deutsche Bibel Gesellschaft, Stuttgart.
Krasovec, J (2010) The Transformation of Biblical Proper Names, T & T Clark International
Ladefoged P (2001), A Course in Phonetics, Harcourt, Florida
Puterman, M L (2005) Markov Decision Processes: Discrete Stochastic Dynamic Programing. *Wiley Series in Probability and Statistics*, John Wiley & Sons, New York.
Riding, J D (2012) Hunting the Snark - the problem posed for MT by complex, non-concatenative morphologies. In *Proceedings: Translating and the Computer 34,* ASLIB.
United Bible Societies eds. (1989) The Holy Bible in Kiswahili, Union Version, UBS

Appendix A: Processing Model

The Match Matrix

We begin by arranging the two items to be tested along the axes of a two dimensional matrix (fig 1). This example tests an English model *Abraham* against a Bantu candidate *Abulahamu*. Each word is bounded by the null set marker: ∅ with each cell numbered from 0 on each axis. Any individual character matches between the two names are marked by inserting the matched character into the cell at the intersection of that characters position in the two names. In this example the matched character 'b' is found in each word and the intersection of its position in each name is at coordinates (2,2) on the matrix. Likewise, the character 'h' is found in both names and the intersection of its position in the two names at at (5,6). 'm' similarly is marked at (7,8). The character 'a', however, appears three times in each name and so is matched at nine positions on the matrix:

{(1,1), (1.5), (1,7), (4,1), (4,5), (4,7), (6,1), (6,5), (6,7)}

Although we have identified all the individual character matches between the two names the machine needs a way to determine what might be the best sequence of these matches between the two names. The key to this is sequence. Character matches can only be considered valid if they are in sequence. For a match to be sequential to its predecessor in the sequence both the x and y coordinates of its match must be greater than those of its predecessor. Thus a(1,1) → b(2,2) is a valid sequence but a(1,1) ↛ a(1,5) is not as both matches share a common *x* index.

	0	1	2	3	4	5	6	7	8
0	∅	a	b	r	a	h	a	m	∅
1	a	**a**			**a**		**a**		
2	b		**b**						
3	u								
4	l								
5	a	**a**			**a**		**a**		
6	h					**h**			
7	a	**a**			**a**		**a**		
8	m							**m**	
9	u								
10	∅								∅

fig. 1

n.b. As much of this appendix will be taken up with descriptions of match sequences we shall henceforward abandon the traditional syntax for listing *x,y* coordinates as it will soon become clumsy in favour of: c_y^x where 'c' represents the matched character, 'x' its x index and 'y' its y index.

Our rule of sequence serves also to prohibit match sequences where a successor is not the immediate successor. This whilst $b_2^2 \rightarrow h_6^5$ obeys the rule that both coordinates of the successor must be greater than the coordinates of the predecessor h_6^5 is not the immediate successor of b_2^2 and so is not truly sequential. The best sequence from b_2^2 is in fact a_5^4 which then leads us on to h_6^5. We have in fact found a three character sequence: $b_2^2 \rightarrow a_5^4 \rightarrow h_6^5$. Closer examination of the matrix shows us that this sequence is in fact part of a longer sequence beginning at a_1^1 (we disregard the null set boundary markers for clarity). Starting the sequence at a_1^1 gives us a match sequence across the whole matrix of: $\{a_1^1 \rightarrow b_2^2 \rightarrow a_5^4 \rightarrow h_6^5 \rightarrow a_7^6 \rightarrow m_8^7\}$. This looks like the best match sequence across the whole matrix but it is not the whole story. There are in fact seven possible match sequences across the matrix (disregarding sequences of cardinality < 2):

	0	1	2	3	4	5	6	7	8
0	∅	a	b	r	a	h	a	m	∅
1	a	a			a		a		
2	b		b						
3	u								
4	l								
5	a	a			a		a		
6	h					h			
7	a	a			a		a		
8	m							m	
9	u								
10	∅								∅

fig. 2: $S_1\{a_5^1 \rightarrow a_7^4 \rightarrow m_8^7\}$

	0	1	2	3	4	5	6	7	8
0	∅	a	b	r	a	h	a	m	∅
1	a	a			a		a		
2	b		b						
3	u								
4	l								
5	a	a			a		a		
6	h					h			
7	a	a			a		a		
8	m							m	
9	u								
10	∅								∅

fig. 3: $S_2\{a_5^1 \rightarrow h_6^5 \rightarrow a_7^6 \rightarrow m_8^7\}$

	0	1	2	3	4	5	6	7	8
0	∅	a	b	r	a	h	a	m	∅
1	a	a			a		a		
2	b		b						
3	u								
4	l								
5	a	a			a		a		
6	h					h			
7	a	a			a		a		
8	m							m	
9	u								
10	∅								∅

fig. 4: $S_3\{a_1^1 \rightarrow b_2^2 \rightarrow a_7^4 \rightarrow m_8^7\}$

	0	1	2	3	4	5	6	7	8
0	∅	a	b	r	a	h	a	m	∅
1	a	a			a		a		
2	b		b						
3	u								
4	l								
5	a	a			a		a		
6	h					h			
7	a	a			a		a		
8	m							m	
9	u								
10	∅								∅

fig. 5: $S_4\{a_1^1 \rightarrow b_2^2 \rightarrow a_5^4 \rightarrow h_6^5 \rightarrow a_7^6 \rightarrow m_8^7\}$

	0	1	2	3	4	5	6	7	8
0	∅	a	b	r	a	h	a	m	∅
1	a	a			a		a		
2	b		b						
3	u								
4	l								
5	a	a			a		a		
6	h					h			
7	a	a			a		a		
8	m							m	
9	u								
10	∅								∅

fig. 6: $S_5\{a_1^1 \rightarrow b_2^2 \rightarrow a_5^6 \rightarrow m_8^7\}$

	0	1	2	3	4	5	6	7	8
0	∅	a	b	r	a	h	a	m	∅
1	a	a			a		a		
2	b		b						
3	u								
4	l								
5	a	a			a		a		
6	h					h			
7	a	a			a		a		
8	m							m	
9	u								
10	∅								∅

fig. 7: $S_6\{a_1^4 \rightarrow h_6^5 \rightarrow a_7^6 \rightarrow m_8^7\}$

Given seven valid match sequences across the matrix (figs. 2-8) how is the machine to decide which is best? A trivial solution is to take the longest as the one likely to be the best but there are circumstances in which this may fail, for example:

	0	1	2	3	4	5	6	7	8
0	∅	a	b	r	a	h	a	m	∅
1	a	a			**a**		a		
2	b		b						
3	u								
4	l								
5	a	a			a		**a**		
6	h					h			
7	a	a			a		a		
8	m							**m**	
9	u								
10	∅								∅

fig. 8: $S_7\{a_1^4 \rightarrow a_5^6 \rightarrow m_8^7\}$

1. Strongly agglutinative languages may stack prefix and suffix morphemes around a stem such that the longest match sequence may not identify the proper-name but elements from a common template. In these circumstances, we rely on the core proper-name phonemes being matched closely and their signal will outweigh that of any morpheme structures.
2. It is possible that no one sequence will be longer than the rest i.e. there may be a number of sequences all matching the same number of characters. In these circumstances we need a way to choose between them.

Evaluating Match Sequences

Given the seven possible match sequences in our example we might ask what it is that leads us to look more favourably on a sequence that begins $\{a_1^1 \rightarrow b_2^2 \rightarrow a_5^4 \ldots\}$ than one beginning $\{a_1^1 \rightarrow h_6^5 \rightarrow a_7^6 \ldots\}$ or $\{a_1^1 \rightarrow b_2^2 \rightarrow a_7^6 \ldots\}$. We observe that our confidence in a pair of matches being sequential is based on the distance between their x and y coordinates on the matrix. Where a pair of matches are proximal (ideally adjacent) in one or both of the words our confidence rises. Conversely, there will come a point when the distance between a matched pair in one or other of the words is such that we have no confidence that the pair is part of a valid sequence. This threshold distance can become a useful processing limit. We will call it theta – θ. The value of theta is language dependent insofar as languages with longer average word lengths may require a higher value for theta but for most languages setting theta as 20 is sufficient. Let us return to our three example sequence fragments quoted above:

1. $\{a_1^1 \rightarrow b_2^2 \rightarrow a_5^4 \ldots\}$
2. $\{a_1^1 \rightarrow h_6^5 \rightarrow a_7^6 \ldots\}$
3. $\{a_1^1 \rightarrow b_2^2 \rightarrow a_7^6 \ldots\}$

Taking example 1. we construct a set of ordered pairs which represent the pairs of matched characters along the sequence: $\{(a_1^1, b_2^2), (b_2^2, a_5^4) etc \ldots\}$ The proximity of the successor to the predecessor in each pair may be expressed as the difference between the *x* and *y* coordinates for each character in the pair. Thus for (a_1^1, b_2^2) the difference between the x coordinates for each character is: $|a^x - b^x|$ (we will call this *dx*) and for the y coordinates: $|a_y - b_y|$ (which we name *dy*). For this pair $|a^1 - b^2| = 1$ and $|a_1 - b_2| = 1$ so *dx* and *dy* are both 1. For the second pair in the sequence (b_2^2, a_5^4) *dx* = 2 and *dy* = 3. We need to combine these two values such that the larger of the two has the greatest effect on our assessment of proximity. We ensure this as follows by defining d_1 as the greater of *dx* and *dy*: $d_1 = max(dx, dy)$ and d_2 as the smaller of the two values: $d_2 = min(dx, dy)$. We now combine the two values to form a single value *d* thus: $d = \theta - \left(d_1 + \left(\frac{1}{\theta} \cdot d_2\right)\right)$.

Working this for our example we find that for the first pair (a_1^1, b_2^2) d_1 and d_2 are both 1 and so we calculate *d* for this pair as: $d = 10 - \left(1 + \left(\frac{1}{10} \cdot 1\right)\right) = 8.9$. For the second pair (b_2^2, a_5^4) *dx* =

2 and $dy = 3$ so $d_1 = 3$ and $d_2 = 2$. d for this pair is therefore: $d = 10 - \left(3 + \left(\frac{1}{10} \cdot 2\right)\right) = 6.8$.

If we apply this to example 2: $\{a_1^1 \rightarrow h_6^5 \rightarrow a_7^6 \ldots\}$ we find that for the first pair $\left(a_1^1, h_6^5\right)$ d evaluates to 4.6 and for the second pair $\left(h_6^5, a_7^6\right)$ d evaluates to 8.9. Similarly for our third example $\{a_1^1 \rightarrow b_2^2 \rightarrow a_7^6 \ldots\}$ d for the first pair evaluates as 8.9 and for the second pair as 4.6.
For the value v of an entire match sequence dS we simply calculate the value for each pair of matches in the sequence and then take the product of those values: $v = \prod_{i=1}^{|dS|-1} \theta - \left(d_1 + \left(\frac{1}{\theta} \cdot d_2\right)\right)$
For the seven sequences generated by our example match this returns the following values:

$dS_1 = 46.92$
$dS_2 = 467.34$
$dS_3 = 294.77$
$dS_4 = 42664.72$
$dS_5 = 350.04$
$dS_6 = 388.13$
$dS_7 = 40.02$

It can be seen at once that we have a clear winner. When identifying a proper-name from amongst the other words around it we can count ourselves spectacularly unlucky should another word return a match sequence of similar strength to that of the name. Clearly, the shorter the name and the longer the average word length in the language the greater the chance of a spurious match but experiment suggests that for the vast majority the match-sequence generated by the proper-name will be strongest.
To express the match value for a sequence as a probability we simply calculate the value of a perfect match p for the shorter of the two words: $p = 8.9^{|dS|-1}$ and then divide v by p to render the result as a probability between 0 and 1. So for our two words *abraham* and *abulahamu* the best possible match sequence would be 6 pairs (|*abraham*| - 1) matching every letter of *abraham* as adjacent in both items: which gives $p = 496981.29$. Our best sequence dS_4 gives $v = 42664.72$ and $P(x, y) = \frac{v}{p}$ gives a probability of 0.85 for the match being valid. The next best sequence dS_2 we find gives $P = 0.009$.

Appendix B: Mapping Discontinuous Structures in Natural Language – Results

Use Case 1: *Proper Name Identification*

Model Name	**Finnish**		**Fulfulde (Niger-Congo)**	**Tojolabal (Mexico-Mayan)**	
Abiathar	Abjatar Abjatarille	Abjatarin Abjatarilta	Abiyater	Abiatar Abiatari	
Aziel	Asielin		Ajiyel	Azieli	
Boaz	Boas Boasin Boasille		Not found	Booz Boozi	
Chedorlaomer	Kedorlaomer Kedorlaomerin Kedorlaomeria		Kedorlayomer	Kedorlaomeri Kedorlaómeri kedorlaómeri kedorla'omer	
Darius	Dareios Dareioksen Dareiokselle		Dariyus	Darío	
David	Daavid Daavidia Daavidiin Daavidkin Daavidin	Daavidille Daavidista Daavidilta Daavidilla	Daawuda	David Dabidi Davidi Davida	Dabid davidi Dvidi
Elimelech	Elimelek Elimelekin Elimelekille		Elimelek	Elimelec Elimelek Elimeleki	
Ezekiel	Hesekiel Hesekielille		Ejekiyel	Ezequiel Ezequieli	
Festus	Festus Festukselle	Festuksen	Festus	Festo	
Gehazi	Gehasi Gehasia	Gehasin Gehasille	Geehaji	Guehazi	
Haman	Haman Hamanin Hamanille	Hamanista Hamania	Haman	Amán Amani Amána	
Jezebel	Isebelin Isebel Isebelille		Ijabel	Jezabel Jezabeli Jezreel	
Lo-Ruhamah	Lo-Ruhama Lo-Ruhaman		Not found	Lo-ruhama	
Methuselah	Metuselah Metuselahin		Matusala	Matusaleni Matusalen	
Nebuchadnezzar	Nebukadnessar Nebukadnessarille Nebukadnessarin Nebukadnessaria		Buutunasar	Nabucodonosor Nabucodonosori	
Peninnah	Peninna Peninnalle	Peninnalla	Peninna	Peniná	
Rehoboam	Rehabeam Rehabeamin Rehabeamista Rehabeamille		Robo'am	Roboami Roboam	
Simon	Simon Simonin Simonia Simonille		Simon	Simon Simón Simoni	
Timothy	Timoteus Timoteus-niminen Timoteukselle	Timoteuksen Timoteusta	Timote	Timoteo	
Yahmai	Jahmai		Yakamay	Jahmai	
Zephaniah	Sefanja Sefanjalle		No text	No text	

Use Case 2: *Non-Concatenative Morphology Analysis (Classical Hebrew)*

The text of the first 11 chapters of the book Genesis [Kittel, 1997] was used as input data. Results are presented using Michigan-Claremont encoding using '_' to represent lacunae in patterns.

First Iteration Output:

Morphology:

Template	*PoS*
T.O_A_:NFH	Qal. fut. 3ppf
T.O_A_	Qal. fut. 2psm / 3psf
TO_:_W.	Qal. fut. 2ppm
TO_A_	Qal. fut. 2psm / 3psf
Y.I_:_:_W.	Qal. fut. 3ppm
Y.O_:_W.	Qal. fut. 3ppm
Y.O_A_	Qal. fut. 3ppm
YI_:_:_W.	Qal. fut. 3ppm
YO_A_	Qal. fut. 3psm
_.E_E_	n. m. coll.
_:E_O_	Qal. imp. s. m.
_:_F_IYM	n. m. p.
_:_F_	Qal. pret. 3psm
_A_:_W.	Piel fut. 3ppm
_E_E_	n. m. coll.
_F_:_FH	Qal. pret. 3psf
_F_:_W.	Qal. pret. 3pp
_F_:_	Qal. 3psm
_F_A_:T.F	Qal. pret. 2psm
_F_A_:T.IY	Qal. pret. 1ps
_F_A_	Qal. 3psm
_F_F_	Qal. pret. 3psm
_I_:_FH	Hiph. pret. 3ps
_I_:_IY	Hiph. pret. 1ps
_I_:_W.	Hiph. pret. 3pp
_O_:_FH	Qal. part. act. f.
_O_:_IYM	Qal. part. p. m. abs.
_O_A_	Qal. fut. 1ps

verification: [Davidson, 1981]

Stems:

_$_B_	שׁב	_L_Y_L_	ליל
_$_B_(_	שׁבע	_M_L_)_	מלא
_$_B_R_	שׁבר	_M_L_K_	מלך
_$_L_$_	שׁלשׁ	_M_N_X_	מנח
_$_L_X_	שׁלח	_M_R_)_	מרא
_$_M_(_	שׁמע	_N_$_)_	נשׂא
_$_M_N_	שׁמן	_N_$_M_	נשׂמ
_$_M_R_	שׁמר	_N_B_L_	נבל
_$_P_X_	שׁפח	_N_P_L_	נפל
_$_R_C_	שׁרץ	_N_S_(_	נסע
_&_M_._	שׂמ	_Q_B_R_	קבר
_&_M_L_	שׂמל	_Q_L_L_	קלל
_(_B_D_	עבד	_Q_R_B_	קרב
_(_B_R_	עבר	_R_K_$_	רכשׁ
_)_K_L_	אכל	_R_P_)_	רפא
_)_M_R_	אמר	_X_P_R_	חפר
_B_R_K_	ברך	_Y_C_)_	יצא
_C_D_Q_	צדק	_Y_K_L_	יכל
_D_B_R_	דבר	_Y_L_D_	ילד
_G_D_L_	גדל	_Y_R_)_	ירא
_H_L_K_	הלך	_Z_Q_N_	זקנ
_L_Q_X_	לקח		

Professional precariat or digital elite? - Workshop on interpreters' workflows and fees in the digital era

Dr. Anja Rütten
Sprachmanagement.net
Paradiesstr. 3
41849 Wassenberg
Germany
ruetten@sprachmanagement.net

Abstract

In today's digital and connected environment, it has become much easier to dissect services like interpretation into very small units. In some cases, interpreters working in micro units, i.e. within a limited space of information, may have a business case, in others, they operate in less restricted and predictable "macro" information space, having to recur to a wide range of secondary context, background and linguistic knowledge. Accordingly, payment can occur on micro and macro level, i.e. taking account or not of the secondary knowledge work involved in an assignment. Small payment units (minutes, or words) are not the most useful way of remunerating "informed" macro level knowledge work, but don't necessarily have to exclude it. Software and digital platforms might not only be the catalyst of small-piece contracting, it could also serve as a means to make interpreters' knowledge work more efficient and profitable, thus provide optimum quality and value for money to the customer.

1 Introduction

Interpreters being paid by the minute (or hour) nowadays does not seem as inconceivable as it used to be. Technically speaking, small worktime and payment units have become easier to handle, thus more probable to be applied. The question arises how to distinguish between micro and macro knowledge work and which working/payment units best to apply to suit the needs of both interpreters/translators and customers.

2 Knowledge Work

Interpreters and translators are knowledge workers constantly moving back and forth between different contexts, languages and conceptual systems. In order to do so, they rely on their own knowledge base being complemented by external information sources explored in what can be called "secondary knowledge work" in order to properly perform the actual, primary knowledge work, i.e. the interpreting assignment or translation at hand (Rütten 2007:102ff). Interpreters do so mainly during preparation and, to a limited extent, on the job when doing ad hoc research and after the job, while a translator's secondary knowledge work tends to be more intermittent and less clearly distinguishable from the primary task of translating.

Proceedings of the 38th Conference Translating and the Computer, pages 133–137,
London, UK, November 17-18, 2016.

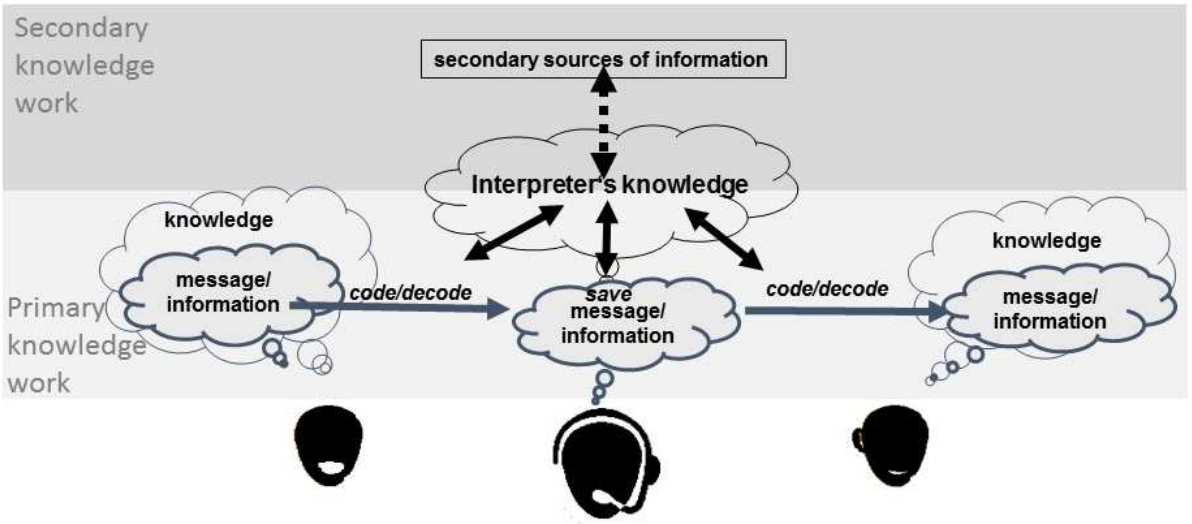


Figure 1: Primary and secondary knowledge work in interpreting

2.1 Macro Knowledge Work

What interpreters need to know in order to interpret a certain speech usually goes far beyond the text itself – both conceptually, linguistically and pragmatically. In a macro knowledge work scenario, they do indeed have this knowledge. Ideally even freelance interpreters are familiar with the broader context of a company or organisation and the respective industry, political framework etc., the technicalities as well as the language used in that environment. This makes them all-round language service providers taking care of anything ranging from translations (documentation, website, brochures, meeting documents, even short emails) and terminology to interpretation of meetings, sales events, negotiations and short phone calls – cooperating if need be with a team of freelance colleagues. Micro work elements – such as interpreting (or making) the said phone call, answering spontaneous terminological questions or translating single sentences – smoothly integrate into the macro level of long-term, relatively large-volume language service provision, there is no need for extensive secondary knowledge work in order to familiarise with the respective product, intentions of the persons involved and typical jargon, the background knowledge as a decisive production factor having been acquired (and financed) in the course of the long-term cooperation (Rütten, 2007:101ff). Primary and secondary knowledge work go hand in hand. In this scenario, a company may well draw from the extensive insight the interpreter or translator has and rely on their professional judgement and advice.

2.2 Micro Knowledge Work

Sometimes the cost and effort of recruiting a professional would by far exceed the benefit. This is the case when no context knowledge is required to fulfil the task or when quality simply does not pay off. If a customer lives off selling very cheap products and needs multilingual categorising or key word finding for tons of products just to feed the search engines, then less quality for less money is a business case. They may give thousands of words to a dozen translators and have them translated in no time, saving time and money by not investing in the meaningful translation of a text that, after all, has a very short life-expectancy.

When confidential matters are interpreted, like in medical or legal interpreting, the customer might rather prefer the interpreter not to accumulate vast knowledge about matters discussed or parties involved. Under certain circumstances context knowledge may even be a caveat

when, for example, unbiased and unprejudiced views are required by the customer and an informed interpreter might be prejudiced and render a pre-filtered version of what is being said on the basis of what he or she considers important or unimportant.
In the aforementioned cases, the idea –not only from the customer's, but sometimes also from an interpreter's point of view– might be to use a minimum input of secondary knowledge to fulfil the interpreting task at hand and make no effort to keep and maintain it, just like current assets in a factory, rather than seeing it as a necessary production factor to invest in and benefit from, be it once or repeatedly. In such cases, the benefit of having an informed interpreter who understands allusions and references to past events or persons not present is seemingly outweighed by the cost savings and/or other factors like confidentiality or impartiality. Although it should be mentioned that professional interpreters are bound by professional secrecy anyway, as it is stated for example in the AIIC Code of professional ethics (AIIC 2012).
In all these cases, secondary knowledge work as part of what Julia Böhm (2007) calls "job-specific time input" is deliberately kept to a minimum, thus cutting the usual overall time input for an interpreting assignment roughly by half. However, when this reduction of time input is merely about confidentiality or impartiality, it could be argued that, while leaving out all context information, the purely semantic dimension of preparation, i.e. special legal/medical knowledge, can (or should) still be part of the interpreter's background knowledge, be it that this knowledge is acquired especially for the assignment at hand or has been accumulated in earlier jobs or training. In this case, interpretation is a combination of micro and macro level work.

3 Payment Units

3.1 Macro Payment

When remuneration is based on larger units –like in the case of employees' monthly or annual salaries– the long-term benefit provided to the company or organisation by the employee based on their experience, training, soft skills etc. positively influences the amount being paid (macro payment). The largest usual payment unit for freelance interpreters is a day and for translators an hour (if not paid by the word or line). Without an in-depth survey it is hard to tell whether the knowledge acquired in the long run by the interpreter, as well as the time required for the secondary knowledge work dedicated to a special assignment, are factored in when these fees are calculated. Conference interpreters tend to argue that their daily fees include preparation. However, when analysing the typical cost structures, this often turns out not to be true (AIIC, 2015a).
Generally speaking, the larger the work volume the smaller will be the proportion of secondary knowledge work in relation to the primary task. This is due to a certain scale effect when working on a macro level, for the effort of familiarisation/knowledge acquisition can be allocated to a larger amount of work, i.e. a long and/or repeated assignment or the sum of translation plus interpreting plus any other minor linguistic support like phone calls and emails, provided that these tasks are in a way interrelated. A typical example would be interpreters having been present at meetings and translating documentation beforehand or the minutes afterwards and also interpreting the occasional phone call between the meeting participants. As they are familiar both with the subject matter and with the people involved, they will not have to prepare as much as someone unfamiliar and, more importantly, be able to compensate the loss of visual and contextual information on the phone and read (or hear) between the lines more easily.
Larger work volumes tend to be remunerated in larger payment units. Let's say a two-day interpreting assignment will hardly be paid by the hour, whereas this might be the case for a two-hour job, and a customer might tend to pay a fifty-minute job by the minute. However, if

a small one-off project involving a small amount of micro working units (minutes) is not embedded in a long term, macro-type of cooperation but "informed" interpreting on a macro level is still expected then macro payment will be more appropriate in order to account for the secondary knowledge work required. It does not necessarily have to be in big payment units as long as the preparation effort is factored in. However, this may be easier to factor into bigger payment units, as will be discussed more in detail under 3.2.

3.2 Micro Payment

In translation, payment in small units like words or lines (i.e. characters) has been common practice for a long time. In interpreting, it is becoming increasingly popular at least from the customer side what with Voice over IP and remote interpreting techniques. Crowd sourcing platforms offer a superb technical environment for assigning micro jobs and will be happy to inform crowd workers about their excessive pricing (without knowing their cost base) simply based on a comparison of prices indicated by their competitors. With smaller payment units, the focus may be reduced to mere primary knowledge work with the secondary knowledge work being lost out of sight and thus not being factored in both time-wise and financially. This may be a sensible thing to do for the reasons mentioned above–basically if the job at hand requires low qualification. It may, however, happen accidentally, i.e. when "informed" macro knowledge work is required and the additional effort of macro knowledge work is not assigned to the small payment units.
The idea of working and paying on a macro level while using small payment units may sound contradictory at first but it works perfectly well for many translators provided they do not calculate their fees on the basis of some words being typed away. The same goes for interpreting, which might even be charged by the minute as long as the scope of the calculation is not limited to the mere physical presence of the interpreting person. It may, however, be difficult to calculate if the number of minutes needed is unknown beforehand. If, for example, a price per minute were to be fixed for "over the phone" interpreting, this would have to vary in the extreme according to the number of minutes bought. If an interpreter prepares one hour for an assignment and needs to earn 80 EUR/hour worked in order to cover the costs (AIIC 2015a), then the prices charged for per minute of interpretation would have to be as follows:

preparation minutes	interpreting minutes	total minutes	total price (total minutes *1.33 EUR)	price per minute of interpreting
60	1	61	81.13 €	81.13 €
60	5	65	86.45 €	17.29 €
60	30	90	119.70 €	3.99 €
60	60	120	159.60 €	2.66 €
60	120	180	239.40 €	2.00 €

The principle (and difficulty) of calculating small unit prices and volume discounts while factoring in preparation time becomes quite clear. Any other factors like opportunity costs for a day blocked that might as well have been sold as a complete conference day or an extra pay for the special fatigue of phone interpreting have not been taken into account for the sake of simplicity.

4 The role of software

On the internet, any product and service can be found and recruited much easier than in the pre-internet era. Crowd sourcing and job platforms nowadays first and foremost facilitate the

search for and selection of interpreters and translators for both large and small jobs. This is an advantage in terms of speed and facility of recruitment, but the internet and digitalisation could do much more than that. The nice thing from a consumer perspective is that any odd niche product can be sold from anywhere in the world to anyone in the world. With interpreting jobs becoming ever more specialised (almost anyone speaks enough English to talk about the weather or ask for a spare blanket at the hotel reception), it would be a huge advantage to be able to track down exactly the "niche" interpreter with the right language combination and experience in exactly the required subject area, possibly even not too far away from where the service is needed. Customers and service providers could be matched much more efficiently and precisely than what used to do the yellow pages, taking full advantage of the merits of digitalisation. An interpreter who has heard about, say, otolaryngology before will be able to prepare for an assignment in maybe half the time a completely newbie would, thus provide higher quality interpretation, at much shorter notice if need be, being able to anticipate and make conclusions on the basis of knowledge acquired formerly (a long-term asset). If the internet were properly used to find exactly the right interpreter, the mutual benefit would be much higher than it is at the moment. Interpreters could specialise more easily and customers could benefit from long-term, in-depth cooperation both in terms of pricing and quality.

As to supporting interpreters' knowledge workflow, many applications have been developed recently (Rütten 2015). In contrast to translator's translation memory systems, which provide both financial benefit to the customer and boost efficiency in the translator's workflow, interpreters' knowledge management systems so far do indeed provide support for efficient macro knowledge work, but they are hardly ever used as a basis for long term cooperation with customers. Cloud-based team terminology work has become quite popular recently, but it tends to be a mere workload-sharing exercise in order to be able to keep up with the preparation of extremely dense and technical presentations at all. The resulting group discussions of conceptual or linguistic questions, which lead to better cognitive processing and a common understanding about the subject matter and language to be used within the team of interpreters, often come as a mere side effect. Customers tend to play a very minor role in this exercise.

5 Way forward

Increased price pressure and exchangeability of service providers are one side –the micro dimension– of digitalisation and the internet. Potential benefits of new technologies could be found rather on the macro level, like e.g. better "match-making", long-term cooperation and efficient knowledge management, possibly shared not only among interpreters, but also with customers.

References

AIIC. 2012. "Code of Professional Ethics". *aiic.net* http://aiic.net/page/6724 [last accessed September 27, 2016]

AIIC Germany Profitability Working Group. 2015a. "How to make a living as a conference interpreter: Part 1 – Understanding expenses and fees". *aiic.net.* http://aiic.net/p/7128. [last accessed September 22, 2016]

AIIC Germany Profitability Working Group. 2015b. "How to make a living as a conference interpreter: Part 2 – Best practice for profitability". *aiic.net.* http://aiic.net/p/7130 [last accessed September 22, 2016]

Böhm, Julia. 2007. "Budgeting time and costs for professional conference interpreters: who wants to be a millionaire?". *aiic.net.* http://aiic.net/page/2760 [last accessed September 27, 2016].

Rütten, Anja. 2015. "Summary table of terminology tools for interpreters". www.termtools.dolmetscher-wissen-alles.de [last accessed September 29, 2016].

Rütten, Anja. 2007. Informations- und Wissensmanagement im Konferenzdolmetschen. Sabest 15. Frankfurt: Peter Lang.

How to Configure Statistical Machine Translation with Linked Open Data Resources

Ankit Srivastava, Felix Sasaki, Peter Bourgonje, Julian Moreno-Schneider, Jan Nehring, Georg Rehm
DFKI GmbH, Alt-Moabit 91c, 10559 Berlin, Germany
firstName.lastName@dfki.de

Abstract

In this paper we outline easily implementable procedures to leverage multilingual Linked Open Data (LOD) resources such as the DBpedia in open-source Statistical Machine Translation (SMT) systems such as Moses. Using open standards such as RDF (Resource Description Framework) Schema, NIF (Natural language processing Interchange Format), and SPARQL (SPARQL Protocol and RDF Query Language) queries, we demonstrate the efficacy of translating named entities and thereby improving the quality and consistency of SMT outputs. We also give a brief overview of two funded projects that are actively working on this topic. These are the (1) BMBF funded project DKT (Digitale Kuratierungstechnologien) on digital curation technologies, and (2) EU Horizon 2020 funded project FREME (Open Framework of e-services for Multilingual and Semantic Enrichment of Digital Content). This is a step towards designing a Semantic Web-aware Machine Translation (MT) system and keeping SMT algorithms up-to-date with the current stage of web development (Web 3.0).

1 Introduction

In a 2001 article in the Scientific American (Berners-Lee et al., 2001), Berners-Lee and collaborators first publicised the concept of the Semantic Web, sometimes called Web 3.0. The initial aim of the Semantic Web was to provide standards through which people can publish documents and data, allowing computer programs to combine and link data from many datasets in order to perform a task just like a human. In a nutshell, the Semantic Web is about making links so that a person or a machine can explore the web of data.

The World Wide Web Consortium (W3C) provides standards promoting common data formats and protocols that constitute the basic technology for the Semantic Web. These are:

- Resource Description Framework (RDF): A formalism to represent data on the web as a labelled graph of objects and their relations
- Uniform Resource Identifier (URI): A compact sequence of characters used to identify resources on the web
- Ontologies: Hierarchical vocabularies of types and relations, allowing more efficient storage and use of data by encoding generic facts about objects. RDF Schema is one such formalism or knowledge representation language.

According to W3C,[1] Linked Data lies at the heart of what Semantic Web is about. The collection of Semantic Web technologies (mentioned above and detailed further in Section 2) provides an environment where an application such as a Machine Translation (MT) system can query data and draw inferences using vocabularies linked on the web. In this paper, we describe an algorithm (in Section 3) using these open standards and tools in order to automatically identify named entities, retrieve their translations from linked data ontologies and feed them to a Statistical Machine Translation (SMT) system. We summarise our experimental results on this semantic web-aware SMT in Section 4, followed by a discussion on the limitations of this

[1] http://www.w3.org/standards/semanticweb/data

Proceedings of the 38th Conference Translating and the Computer, pages 138–148,
London, UK, November 17-18, 2016.

approach in Section 5. After giving an overview of two funded projects actively working in this area as well as comparing our approach to previous works in Section 6, we conclude our paper in Section 7.

2 Tools of the Trade

The main goal of this paper is to provide a workable technique for integrating linked open data resources into a machine translation system.

The term Linked Data, coined in 2006, refers to the ability of the Web to link related data as opposed to just linking related documents. It refers to a set of best practices[2] for publishing and linking structured data on the web. Linked Open Data (LOD) typically refers to linked data with open sharing licenses. These links enable both humans and machines to explore the web of data.

In recent years, there has been a tremendous growth (Schmachtenberg et al., 2014) in both the quality and quantity of data available as linked data on the web. This data can describe named entities such as people, organisations, locations, etc. in multiple languages. This fact coupled with the increased move towards the publication of multilingual language resources such as WordNets and Wikipedia using linked data principles (Chiarcos et al., 2011) has led to a significant increase in the availability of Semantic Web information relevant to Natural Language Processing applications including machine translation.

Typical examples of LOD resources include DBpedia Knowledge Base (Auer et al., 2007), FreeBase (Bollacker et al., 2008), BabelNet (Navigli and Ponzetto, 2012), JRC-Names.[3] In our experiments, we focus on DBpedia, but any of the aforementioned resources with a SPARQL endpoint can be plugged in our SMT system.

In the context of SMT, leveraging translations from Linked Data resources can be likened to plugging external knowledge sources such as terminology banks and translation memories. The major difference is that linked data is stored in a different data format (NIF based on RDF) and is accessed using a dedicated query language (SPARQL).

In this section, we describe in brief the enabling technologies, standards, and software used in our experiments to configure SMT with linked data.

2.1 RDF and RDFS

Resource Description Framework (RDF) is a XML-like syntax providing the foundation for representing and processing machine readable data.

RDF is a graph-based model whose basic building block is an entity-attribute-value triple. There are three fundamental concepts of RDF:

- Resources are objects referenced by an identifier or URI
- Properties describe relations between resources
- Statements assert the properties of resources in the form of entity-attribute-value triple, consisting of a resource, a property, and a value. The value can either be a resource or a literal (atomic values such as language codes)

RDFS (RDF Schema) is a primitive ontology language. It is a vocabulary used to define helpful properties (such as rdfs:label for language name) in Resource Description Framework (RDF). An in-depth exposition is provided in The Semantic Web Primer (Antoniou et al., 2012).

[2] http://www.w3.org/DesignIssues/LinkedData.html
[3] https://data.europa.eu/euodp/en/linked-data

2.2 NIF 2.0

NIF 2.0[4] (Natural Language Processing Interchange Format) is an RDF-based format that aims to achieve interoperability between NLP tools such as SMT engines, parsers and annotated language resources such as DBpedia. Its integration with WC standard ITS 2.0[5] (Internationalization Tag Set) makes it attractive to multilingual applications.

The primary use case of this standard is to serve as an input and output format for web services, that enables seamless pipelining or combination of various language processing web services in sequence (Hellmann et al., 2013).

An important characteristic of this standard relevant to NLP is that the atomic unit is a character rather than a word. Thus, if a sentence has 23 characters (including spaces between words), the resource or sentence spans from 0 to 22.

2.3 SPARQL

SPARQL[6]is recursive acronym for SPARQL Protocol and RDF Query Language. It is a query language (like SQL) primarily for linked data, used to retrieve information from RDF-encoded data including NIF. It is a W3C recommended standard. In simple terms, if the data such as a multilingual lexicon is stored as a linked data (NIF standard), then SPARQL is a tool to retrieve information from the linked data such as translations in the required target language.

2.4 DBpedia

DBpedia[7] is a linked open dataset (extracted from Wikipedia) consisting of 4.58 million entities in up to 125 languages and 29.8 million links to external web pages. DBpedia Spotlight[8] is an open-source tool for automatically annotating mentions of DBpedia resources in text. Note that the translations may be prone to error on account of being user generated.

2.5 Moses

Moses[9] (Koehn et al., 2007) is an open-source SMT system used in our experiments as a test bed for Semantic Web-enabled MT. We have employed Phrase-based Statistical Machine Translation system with standard configurations, as specified in Section 4. The translations from the LOD such as DBpedia are inserted in a forced decoding framework, wherein the translation of selected named entities are chosen from DBpedia instead of the Moses decoder.

3 Methodology

Having touched upon the basic building blocks for configuring a SMT system with LOD resources in Section 2, we now describe the framework to interface a Moses-based SMT system with a DBpedia-based LOD resource. The procedure comprises of 6 steps as enumerated below.

1. Convert the text to be translated into a NIF document

2. For each sentence, API call e-NER (Named Entity Recognition) service

3. For each of the named entities (marked in NIF), API call the e-linking service, that is, annotate named entities in the document using DBpedia Spotlight mentioned in Section 2

[4]http://persistence.uni-leipzig.org/nlp2rdf/
[5]http://www.w3.org/TR/its20/
[6]http://www.w3.org/TR/rdf-sparql-query/
[7]http://wiki.dbpedia.org
[8]https://github.com/dbpedia-spotlight/
[9]http://www.statmt.org/moses/

4. For each link (named entity resource identified in the DBpedia LOD), retrieve the translation in target language using a SPARQL query for attribute *rdfs:label* which contains the language identifier

5. Integrate these translations in the Moses decoder. Encode the named entity and its translation in a format compatible with the Moses decoder (enabled with the xml-input feature)

6. Display translated output in the appropriate format

We will illustrate the mechanism behind each step in the framework with the help of an example sentence. Consider translating an English sentence (from the IT-domain) *"MS Paint is a good option."* into German. Note that all the procedures below are carried out by freely available web service API calls, the source code for which can be found at `https://github.com/freme-project` for FREME web services[10] and at `https://github.com/dkt-projekt` for DKT web services.[11] More information about these projects and their tools can be found in Section 6.

Convert into NIF: All the web services for various NLP applications including MT hosted by the DKT and FREME are NIF-enabled. The NIF core technology provides classes and properties to describe the relations between substrings, text, documents, and their URI schemes or identifiers (Hellmann et al., 2013).

Listing 1: Representing a sentence in NIF.

```
<http://freme-project.eu/#char=0,26>
    a       nif:Context , nif:RFC5147String , nif:Sentence ;
    nif:anchorOf            "MS paint is a good option." ;
    nif:beginIndex          "0" ;
    nif:endIndex        "26" ;
    nif:firstWord       <http://freme-project.eu/#char=0,2> ;
    nif:isString        "MS paint is a good option." ;
    nif:lastWord        <http://freme-project.eu/#char=25,26> ;
    nif:refContext          <http://freme-project.eu/#char=0,26> ;
    nif:word                <http://freme-project.eu/#char=9,11> ,
                            <http://freme-project.eu/#char=3,8> ,
                            <http://freme-project.eu/#char=12,13> ,
                            <http://freme-project.eu/#char=19,25> .
```

From Listing 1, we observe how the English sentence (source language) *"MS Paint is a good option."* is assigned a URI including the character spans 0 through 26 (first line). There are various attributes or properties such as all the words, firstWord, and lastWord. However the most important line for our purposes is the whole sentence denoted by *nif:isString*.

Tag the Named Entities and link with DBpedia entries: Herein we have combined steps 2 and 3 mentioned above into one process.

Figure 1 shows a screen-shot of a typical lexical entry on DBpedia for the entity *Paint (software)* linked to the phrase *MS Paint* in our sentence. Figure 2 displays the same entry focusing on the concepts *rdfs:label* and *owl:sameAs* which lists links to the same entity

[10] Of particular interest is the web service named e-entity/dbpedia-spotlight.

[11] Of particular interest are the services DKTBrokerStandalone/nifTools, e-NLP/Sparqler, and e-SMT.

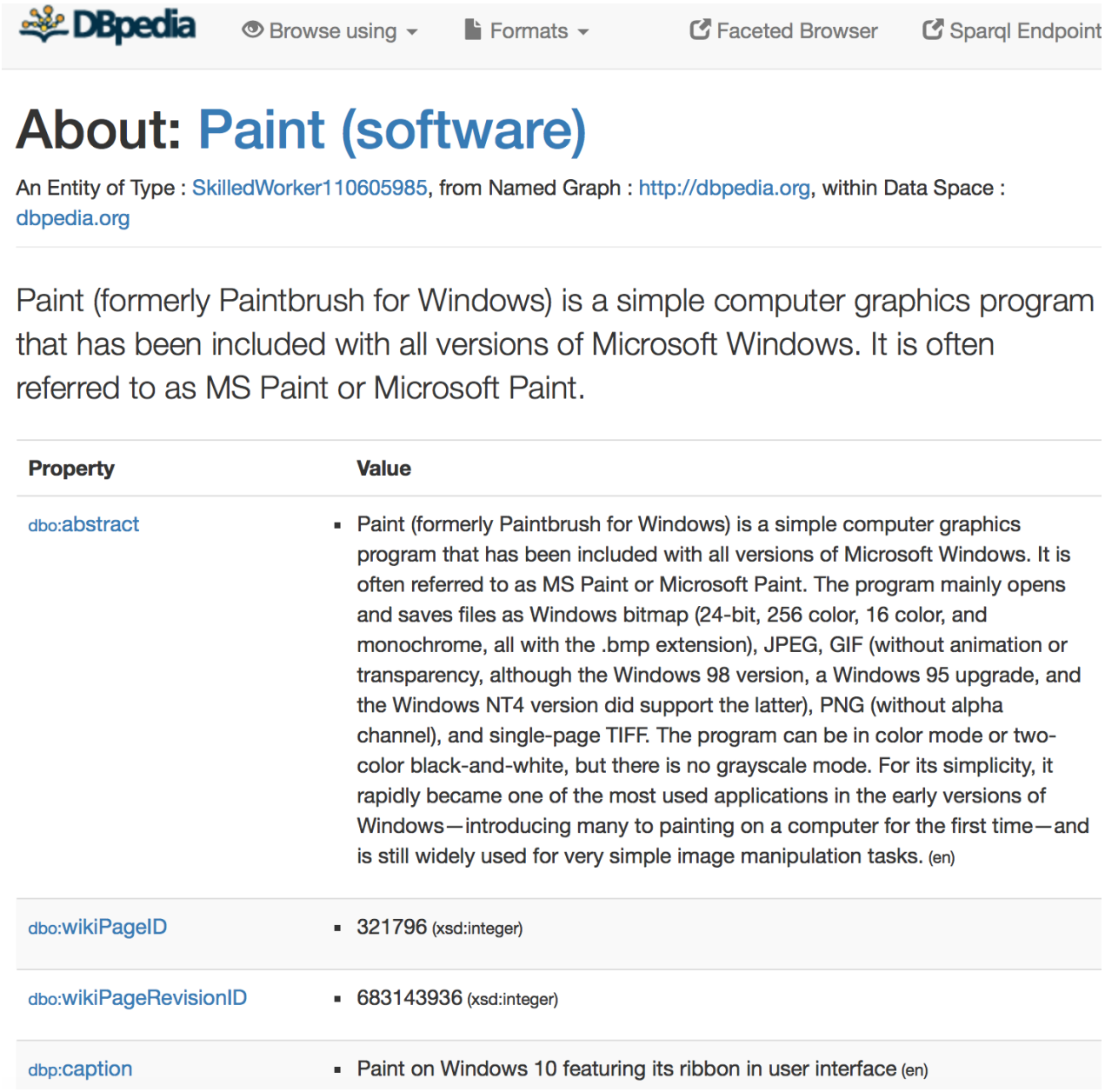


Figure 1: Screen-shot of the DBpedia resource

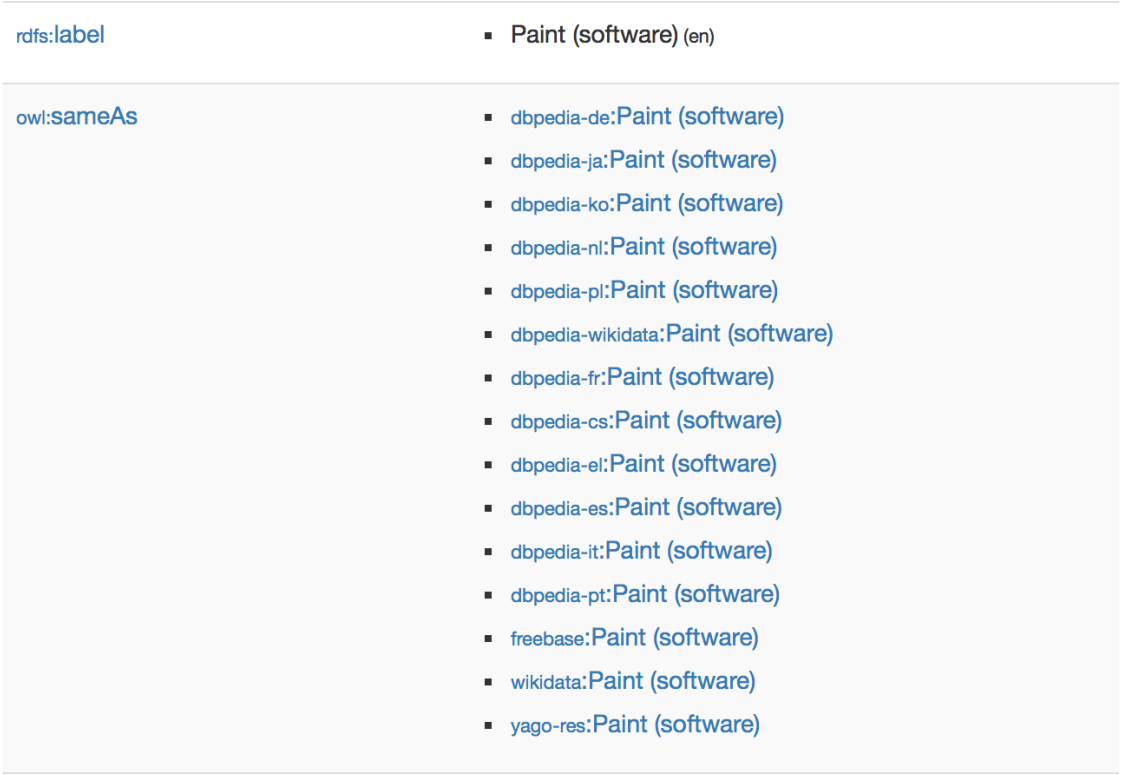


Figure 2: Screen-shot of the DBpedia multilingual entries

(Microsoft Paint) in different languages identified by a 2-digit language code. For example, *de* denotes German language.

The sentence is parsed by the FREME DBpedia-Spotlight web service and all entities or terms which occur in our LOD resource (DBpedia) are annotated with the property *itsrdf:taIdentRef*. A fragment of a NIF document with the disambiguated term and link to DBpedia entry is shown in Listing 2:

Listing 2: Output from FREME NER in the NIF format.

```
<http://freme-project.eu/#char=0,8>
    a                       nif:RFC5147String , nif:Word ;
    nif:anchorOf            "MS-Paint" ;
    nif:beginIndex          "0" ;
    nif:endIndex            "8" ;
    nif:nextWord            <http://freme-project.eu/#char=9,11> ;
    nif:referenceContext    <http://freme-project.eu/#char=0,26> ;
    nif:sentence            <http://freme-project.eu/#char=0,26> ;
    itsrdf:taIdentRef
                <http://dbpedia.org/resource/Paint_(software)> .
```

Query for Target Entity Translation: As stated in Step 4 of the procedure, we use DBpedia SPARQL endpoint available at `https://dbpedia.org/sparql`. This can be directly invoked from inside Java code in the DKT and FREME e-services. Essentially, the DBpedia database is stored as a triple store or a Graph store alluding to the entity-attribute-value triple structure of the RDF data.

The SPARQL query snippet shown in Listing 3 helps us retrieve German *(de)* translations for each annotated named entity or resource (denoted by `http://dbpedia.org/resource/`

`Paint_(software)` in our example), using the attribute *rdfs:label*.

Listing 3: Code Snippet for a SPARQL Query.

```
PREFIX dbpedia: <http://dbpedia.org/resource/>

SELECT distinct *
WHERE {
    <http://dbpedia.org/resource/Paint_(software)>
    rdfs:label ?label
    filter langMatches( lang(?label), "de" )
}
```

Moses Decoder Integration: Once we have identified the entities and obtained their translations from a LOD resource such as DBpedia, the next step is to plug it in a machine translation system such as Moses. Essentially we treat the SMT system as a black box thus making it theoretically possible to substitute any MT system for Moses as per the user requirements.

The Moses decoder is "forced" to use translations for the named entities tagged by the linked data instead of relying on its own translation models and phrase tables. We achieve this by invoking the Moses decoder with the *xml-input*[12] feature turned on, demonstrated by a command-line code snippet in Listing 4. The phrase *MS Paint* has its translation *Microsoft Paint* (retrieved by the SPARQL query) hardcoded before the Moses decoder is initiated.

Listing 4: Code Snippet for a command-line call of Moses.

```
% echo '<np translation="Microsoft Paint">MS Paint</np>
is a good option ." | moses -xml-input exclusive -f moses.ini
```

Display Translated Output: Once we have translated the whole sentence, the procedure is complete, and the translation can either be simply displayed as a *plaintext* string (*"Microsoft Paint ist eine gute wahl."*) or encoded in *NIF* format as shown in Listing 5. The property *itsrfd:target* is associated with linking the translated string along with the target language code (*de*) to the remainder of the NIF document. This format or any other RDF-style (linked data) format is just so that the output can be further chained as input in subsequent NLP applications in a seamless manner.

Listing 5: NIF representation of a sentence and its translation.

```
@prefix xsd:    <http://www.w3.org/2001/XMLSchema#> .
@prefix itsrdf: <http://www.w3.org/2005/11/its/rdf#> .
@prefix nif:    <http://persistence.uni-leipzig.org/
                  nlp2rdf/ontologies/nif-core#> .


<http://dkt.dfki.de/documents/#char=0,26>
    a           nif:RFC5147String , nif:Context , nif:String ;
    nif:beginIndex "0"^^xsd:nonNegativeInteger ;
    nif:endIndex   "26"^^xsd:nonNegativeInteger ;
    nif:isString   "MS paint is a good option." ;
    itsrdf:target  "Microsoft Paint ist eine gute wahl.\n"@de .
```

[12] See Section 4.8.2 in `http://www.statmt.org/moses/manual/manual.pdf`.

4 Experimental Results

Section 3 outlined the core ingredient of this paper, that is, a recipe on how to source translations from linked data ontologies (e.g. DBpedia) into a statistical MT system (e.g. Moses). In this section, we examine the overall benefits, if any, of interfacing a SMT system with Semantic Web resources.

We trained a Moses-based SMT system (Koehn et al., 2007) to translate from English (source language) into German (target language). The set of parallel sentences for training, and the development and test sets for tuning and testing respectively were sourced from the data provided for the WMT 2016 shared task on machine translation of IT domain (Bojar et al., 2016) available at `http://www.statmt.org/wmt16/it-translation-task.html`.

For the purposes of this experiment, we chose this corpora setting, i.e. training a SMT system with large amounts of out-of-domain data (typically European parliamentary proceedings and newswire corpus) augmented with small amounts of domain-specific data (IT-domain corpora such as Libreoffice, Ubuntu, Chromium) in order to translate 1000 IT-domain answers from Batch 3 (the same test set as that employed in the shared task). Batch 1 was used for tuning the translation, language and reordering models (development set). Table 1 outlines the size of the training data.

corpus	entries	words
Chromium browser	63K	551K
Drupal	4.7K	57.4K
Libreoffice help	46.8K	1.1M
Libreoffice UI	35.6K	143.7K
Ubuntu Saucy	182.9K	1.6M
Europarl (mono)	2.2M	54.0M
News (mono)	89M	1.7B
Commoncrawl (parallel)	2.4M	53.6M
Europarl (parallel)	1.9M	50.1M
MultiUN (parallel)	167.6K	5.8M
News Crawl (parallel)	201.3K	5.1M

Table 1: Size of corpora used for SMT.

The motivation was to increase the potential for occurrence of named entities such as technical terms (e.g. Microsoft Paint) in the test data such that we could demonstrate our linked data-aware SMT system. The phrase-based SMT system was trained with standard Moses configuration settings for language model, word alignments, reordering model, explicitly specified in our system description paper for the WMT 2016 IT-domain Shared Task (Avramidis et al., 2016).[13]

Nearly each of the 1000 segments in the test set had at least 1 named entity tagged and annotated. When comparing, a baseline system (translating entirely from the Moses models) with a system whose named entities were translated by linked data resources, a BLEU (Papineni et al., 2002) score improvement of 0.8 (accuracy improved from 34.0 to 34.8) and TER (Snover et al., 2006) score improvement of 2.5 (error reduced from 56.1 to 53.6) was observed. The linked data-aware system identified and correctly translated 12% more terms (named entities) than the baseline SMT system.

One such example of how a SMT system configured with LOD resources benefited and outperformed a baseline SMT system is seen as follows.

[13] http://www.aclweb.org/anthology/W/W16/W16-2329.pdf

SRC (en): MS Paint is a good option.
MT 1 (de): Frau Farbe ist eine gute wahl.
MT 2 (de): Microsoft Paint ist eine gute wahl.

Consider translating the English sentence (the one used to demonstrate our framework in Section 3) into German. MT1 displays the baseline translation where the SMT decoder is unable to disambiguate the term **MS Paint** as a software and not a person. MT2 configured with linked open data gives us the correct translation.

5 Limitations

There are shortcomings and potential pitfalls such as accuracy of user-generated translations in DBpedia and mismatched entity linking which make the case of optimally exploiting linked data resources in SMT system non-trivial.

- Translations are not always accurate because these are user-generated (from Wikipedia entries) and therefore prone to error
- Mismatched Entity Linking. For instance, *MS Paint* only links *MS* to *Microsoft Paint* and leaves the *Paint* unlinked. The result is that *MS* translates to *Microsoft Paint*, while *Paint* is translated separately thereby generating a double translation in the target language. A viable solution is to combine MS and Paint as one entity (pre-processing)
- There is also the issue of how to handle multiple links or translations for a frequently occurring term (entity disambiguation). A possible solution is to pick the top item, or use domain filters (IT-domain versus general in the case of the entity *Paint*).

6 Project Overview and Related Work

6.1 FREME

The project FREME (`http://www.freme-project.eu`) is a two-year European Union Horizon 2020 funded project (started February 2015) aimed towards Open Framework of e-services for Multilingual and Semantic Enrichment of Digital Content. The project involves 8 partners:[14] the DFKI Language Technology Lab, AgroKnow, iMinds, Institute for Applied Informatics, Istituto Superiore Mario Boella, Tilde, VistaTEC, and Wripl.

It essentially hosts a chain of e-services performing diverse NLP applications with the help of interoperability standards such as NIF. The partners lead four business cases around digital content and linked data. With the help of reusable NLP workflows and pipelines, the FREME project provides access to a set NLP and data services demonstrating monetisation of the multilingual data value chain.

More information is available at `https://github.com/freme-project/e-services` and `https://freme-project.github.io/`.

6.2 DKT

The project Digitale Kuratierungstechnologien (DKT: Digital Curation Technologies (`http://digitale-kuratierung.de`)) is a two-year project (started September 2015) funded by the Bundesministeriums fur Bildung und Forschung (German Ministry of Education and Research).

[14] http://www.freme-project.eu/partners/consortium/

The project involves four Berlin-based partner companies (ART+COM AG, Condat AG, 3pc GmbH, and Kreuzwerker GmbH) and the DFKI Language Technology Lab. The project supports digital curation processes carried out by knowledge workers in multiple sectors (museums, television and media, exhibitions, publishers) through robust, precise, and modular language and knowledge technologies. The main goal is to semi-automate the different curation processes (research, annotation, timelining) to make the knowledge workers more time and cost efficient.

More information on the linked data-aware web services is available at `https://github.com/dkt-projekt`.

6.3 Related Work

There have been several approaches in the past that leveraged linked data in SMT systems. Most approaches either use it as an additional knowledge source and training the models on the dictionaries extracted from such resources, or use it in a post-training framework, either forced decoding named entities like our approach or translating unknown words.[15]

McCrae and Cimiano (2013) primarily integrated the dictionary of translations extracted from LOD resources during decoding and creating a new feature for linked data. They essentially let the Moses decoder decide when to chose translations from LOD and when to translate from its phrase tables. In contrast to our approach on forcing translations of all named entities identified by DBpedia, they employ another ontology called Lemon (Lexicon Model for Ontologies[16]) to translate primarily unknown words, that is translations not found by the decoder.

Du et al., (2016) on the other hand leveraged translations from BabelNet dictionaries using both McCrae and Cimiano (2013)'s methods as well as the forced decoding employed by our paper to demonstrate modest improvements in translating English-Polish and English-Chinese data.

It must be noted that the main goal of this paper was to provide a Semantic Web aware method to interface a SMT system with LOD knowledge base via seamless NIF-aware web service API calls. Testing the benefits to a translation system was a secondary outcome. We leave for future work, methods to optimally leverage knowledge from linked data on the Semantic Web and improve SMT system performance, especially in sense disambiguation and translating unknown words.

7 Conclusion

In this paper, we have successfully outlined a procedure to equip an off-the-shelf statistical machine translation system with linked data available on the Semantic Web. With the help of an example, we illustrated a novel machine translation adaptation with the potential for seamless integration into translation and localisation workflows. This is a step towards making MT semantic web-aware.

Acknowledgments

The authors wish to thank the anonymous reviewers for their helpful feedback. The project "Digitale Kuratierungstechnologien (DKT)" is supported by the German Federal Ministry of Education and Research (BMBF), "Unternehmen Region", instrument

[15] While we focus on 2 papers in our related work, a helpful survey of works on Machine Translation using Semantic Web technologies is currently under review and can be found at `http://www.semantic-web-journal.net/content/machine-translation-using-semantic-web-technologies-survey`.

[16] http://lemon-model.net

”Wachstumskern-Potenzial” (No. 03WKP45). More information on the project can be found online at http://www.digitale-kuratierung.de.

References

Antoniou, Grigoris, Paul Groth, Frank van Harmelen, and Rinke Hoekstra. 2012. A Semantic Web Primer. MIT Press, 3rd edition.

Auer, Sren, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. DBpedia: A Nucleus for a Web of Open Data. In *The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference*, pages 722–735.

Avramidis, Eleftherios, Aljoscha Burchardt, Vivien Macketanz, and Ankit Srivastava. 2016. DFKI's system for WMT16 IT-domain task, including analysis of systematic errors. In *Proceedings of the First Conference on Machine Translation*, Berlin, Germany pages 415–422.

Berners-Lee, T., J. Hendler, and O. Lassila. 2001. The Semantic Web. In *Scientific American*, Vol. 284 number 5, pages 34–43, http://www.scientificamerican.com/article.cfm?id=the-semantic-web.

Bojar, Ondrej, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 Conference on Machine Translation. In *Proceedings of the First Conference on Machine Translation at ACL 2016*, Berlin, Germany, pages 131–198.

Bollacker, Kurt, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, pages 1247–1250.

Chiarcos, Christian, Sebastian Hellmann, and Sebastian Nordhoff. 2011. Towards a Linguistic Linked Open Data cloud: The Open Linguistics Working Group. In *TAL*, Vol. 52 number 3, pages 245–275.

Du, Jinhua, Andy Way, and Andrzej Zydron. 2016. Using BabelNet to Improve OOV Coverage in SMT. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portoroz, Slovenia, isbn 978-2-9517408-9-1.

Hellmann, Sebastian, Jens Lehmann, Sren Auer, and Martin Brmmer. 2013. Integrating NLP using Linked Data. In *International Semantic Web Conference*, Sydney, Australia.

Koehn, Philipp, Hieu Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, Wade Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *demonstration session of ACL '07*, Prague, Czech Republic. 177–180.

McCrae, John Philip, and Philipp Cimiano. 2013. Mining Translations from the Web of Open Linked Data. In *Proceedings of the Joint Workshop on NLP, LOD and SWAIE*, Hissar, Bulgaria, pages 8–11.

Navigli, Roberto, and Simone Paolo Ponzetto. 2012. BabelNet: The Automatic Construction, Evaluation and Application of a Wide-Coverage Multilingual Semantic Network. In *Artificial Intelligence*, Vol. 193, pages 217–250.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jung Zhu. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. *ACL 2002, 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, PA. 311–318.

Schmachtenberg, Max, Anja Jentzsch, and Richard Cyganiak. 2014. Linking Open Data Cloud Diagram. http://lod-cloud.net/.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciula, and John Makhoul. 2006. A Study of Translation Edit Rate with targeted Human Annotation. *AMTA 2006, 7th Conference of the Association for Machine Translation in the Americas*, Cambrdge, MA. 223–231.

From CATs to KATs

Félix do Carmo
CLUP—Language Centre of the University of Porto
fcarmo@letras.up.pt

Luís Trigo
LIAAD—Laboratory of Artificial Intelligence and Decision Support
and
INESC-TEC
lptrigo@inescporto.pt

Belinda Maia
CLUP—Language Centre of the University of Porto
bhsmaia@letras.up.pt

Abstract

Current technologies may lead to a revolution to Computer-Aided Translation (CAT) tools. Most of these technologies, which are behind the Machine Translation (MT) comeback, come from the field of Machine Learning. When these technologies are incorporated as extra supports to the tools used by translators, this new generation of tools may be renamed as Knowledge-Assisted Translation (KAT) tools.

We will describe our experience with some of the features that are available in some implementations, but this paper will concentrate on suggesting "Recommended Specifications" for such tools, by resorting to the capacities of Machine Learning methods, complemented by Artificial Intelligence and Augmented Intelligence, to deal with huge volumes of data.

Our starting point is the tasks that translators perform in an interconnected world – clients, and human and machine resources. We will then present some of the Machine Learning features that may be used as supports to the work of translators and post-editors. From domain identification to resource management, there are several areas to study. At the end, zooming into the simpler editing tasks, there are complex theoretical and technological issues that are worth discussing, because they are at the centre of the adaptation that these tools should undergo.

1 Introduction

To analyse the current moment of translation technologies, and to identify what the future may bring us, we need to look beyond the tools that translators use in their daily work, and extend the analysis to autonomous computer translation tools. This is not only due to the fact that we are witnessing an approximation between Computer-Aided Translation (CAT) and Machine Translation (MT), but also because the technologies from both systems may be used together to build new and better tools.

This is not a presentation of an existing system, but a proposal for a change in the views on translation tools. As we hope to demonstrate, the answer to the growing needs for more efficient and easier to use translation tools may be in the technologies that are currently used in MT and in Natural Language Processing (NLP), and which come from Computer Sciences, from such fields as Machine Learning (ML).

We propose that a successful attempt for the integration of MT into CATs must come through the understanding of the different translation processes and procedures translators perform. After the procedural description of the translation process in an industrial context, we present some of the features that may be integrated into a new generation of tools that, instead of trying to build a translated text autonomously, give translators the support they need to build good translations, be it by translating a text from scratch, revising a human translation or by post-editing machine-translated text.

Proceedings of the 38th Conference Translating and the Computer, pages 149–158,
London, UK, November 17-18, 2016.

2 The evolution of computer tools for translation

Traditional translation tools emulate a work desk, with the source text next to the page where the translation is written, both at the centre of the screen and surrounded by references, sources of information and ancillary details. The name most commonly used to identify these packs of applications (Computer-Aided Translation tools, or CATs) clearly conveys the notion that computer assistance is put at the service of translators.

Over the last 30 years or so (the first commercial CAT tools appeared during the 1990s), there have been few revolutions in CAT tool interfaces and editing environments. The biggest one was when the old Trados-style toolbar and the translation unit embedded in a Word document gave way to the more common tabular view. Apart from that, the most important changes in these tools were outside the editing environments, commanded by the different generations of Microsoft Office and the evolution of Web tools, which led to changes in file format filters, tags and formatting standards, among others. CATs have also incorporated project management applications, and adapted to collaborative environments, inside company networks, or on the Web. As a consequence of this, the number of windows and panes in translation tools increased, and translators were called in to play different roles, managing new word counts and file formats, and dealing with new ways to share or protect client and translated content (Austermühl, 2013).

In our professional experience, translators still spend most of their time working in the editor, writing over source language text as they always did, and making decisions based on reference materials that may be managed locally or remotely, which may have different degrees of reliability, according to their specificity, adequacy to domain, language variant, or other features, or which may be too restrictive, like Project Translation Memories that only contain full and fuzzy matches and are useless for a simple concordance search. Two of the most important aids to translators' editing work that have been added to translation tools are Quality Assurance (QA) checks and predictive writing.

Since the 50's, there has been a lot of research in MT, but human translators always seemed to be recognised as necessary to improve its results (Garcia, 2012). However, it is only after the capacities of statistical methods to deal with huge amounts of data were confirmed that widespread MT systems like Google Translate, or Moses (Koehn et al., 2007) were developed. Since 2012, when the first workshop on post-editing was included at a major MT conference (O'Brien et al., 2012), MT research has turned its attention more specifically to the interface between these systems and human translators. So far, this has meant that translators may now receive pre-machine-translated text inserted into the target area in editing tools, mixed with text coming from Translation Memories (TMs), and that they must overwrite that text, performing a specific kind of work known as "post-editing".

2.1 Translation processes and procedures

Let us consider three main processes performed by translators: translation, revision and post-editing.

2.1.1 Translation process

The translation process may be decomposed into four stages of procedures: management, research, writing/editing and revising/checking.

In the translation industry, there is a complex of management procedures usually known as "Project management". We are especially interested in how human and resource management depend on the technical domain of each project.

Before the translation process, translators may read the source text, but especially important before and during the process, is research, which involves clarifying the source text and filling in the gaps in terminology and vocabulary.

However, the most important set of procedures for our analysis is the writing of the translation, especially when it is performed by writing over (or editing) the source text, as is usual in CAT tools.

Finally, there is another set of procedures that involve revising the whole translation and checking different language and formatting levels, from grammatical, spelling and language conventions to terminology and adherence to style guides. We will analyse these points in more detail below.

2.1.2 Revision process

In an industrial environment, revision is a specific process, performed by a person other than the translator. Apart from Mossop's (2007) coursebook, little attention has been given to this process. For now, we would just like to stress that the specifications of this task often imply that the reviser must read the whole source text and the translation, redo all the research that the translator has done, and either validate or override his decisions.

2.1.3 Post-editing process

Post-editing is usually presented as a simple process, with a straightforward definition: "the process whereby humans amend machine-generated translation output to achieve an acceptable final product". (Garcia, 2012). In the sections below, we will try to complement this definition with a more detailed view on the tasks translators perform.

2.2 Supporting features in CAT tools

To translate new sentences or segments, translators write over the source words, and they make translation decisions based on terminology lists and concordance searches in TMs and websites. For repeated segments, they check their context and validate their content, or edit it, if needs be. So, fuzzy matches are the type of segments that specifically require editing (Screen, 2016). CAT tools usually show changes made to the original source segment and translators must edit the translated segment, so as to reproduce these changes. Finally, translators count on the support of QA features and spelling and grammar checkers to do the final verifications before delivery.

The newest generations of CAT tools incorporate more advanced features that support different parts of the translation process. We may now do web searches from within tools such as memoQ and SDL Trados Studio, although in both the only configuration possible is to choose the set of sites to which all word searches will be directed. Other tools, such as Atril's DéjàVu, have had fuzzy composition capabilities for some time. These build translation suggestions for fuzzy matches from fragments of the reference materials, such as terminology databases or MT. For the writing and editing procedures, the most recent innovation is predictive writing, based on suggestions of words, or multi-word units, as the translator is typing along, using sub-segment alignments from the TM or terminology databases.

As for the revision process, work is processed with exactly the same tools as translation. In SDL Trados Studio, for example, there are only slight changes in the interface, from translator to reviser mode: the position of the TM and editing panes, the status of the segments (for version control) and the tracked changes markers, which may be turned off.

2.3 Interactive post-editing tools

Beyond CAT tools, there is a new generation of translation tools that take advantage of MT engines running in the background. In this group of tools, there are different levels of interactivity, but they all exploit the potential of collaborative work.

Although not strictly a translation tool, Google Translate's interface allows for some interactivity that we cannot find elsewhere: users may select chunks of words (created at the

backend) and then call up a list of alternatives and replace or edit the content of such a chunk. However, users cannot resize chunks, move, insert, or delete them (Carmo and Maia, 2015).

CasMaCAT (Alabau et al, 2013) and Lilt (Green et al, 2014) are two of the most recent and advanced interactive translation tools. They both present a very clear interface, and their paradigm for translation work is auto-completion: as the first letters of each word are typed, the system presents translation suggestions. HandyCAT (Hokamp, 2014) presents an open interface that allows researchers to test new interactivity paradigms.

One of the major challenges of interactive and online learning is to balance the complexity of the MT processing at the backend, which deals with major amounts of data and very large search spaces, with the expected interaction by professional translators. In an intrusive architecture, where users' typing habits are interrupted by suggestions, they do not expect to have to correct the same mistakes twice. They want tools that learn, on the fly, from what they have typed. (Moorkens and O'Brien, forthcoming)

2.4 A view on post-editing from Quality Estimation of Machine Translation

Edit distance is a central concept in MT, especially in relation to its evaluation. Metrics like Translation Edit Rate (TER) measure the distance that it takes to transform a translation hypothesis, presented by a MT system, into its reference human translation. This metric measures the number of edits, i.e.: "the insertion, deletion, and substitution of single words, as well as shifts of word sequences" (Snover et al, 2006) that are necessary to make that distance.

Some of the most recent work in this area aims at estimating the level of quality a MT system can achieve, without resorting to a reference translation (Specia et al, 2010). This area of research is known as Quality Estimation (QE), and researchers have been trying to identify the features that may help achieve this objective, by making estimates at the word, phrase, sentence, or even document level. One of the purposes of these tasks is to classify texts in terms of their editing effort, namely in predicting the types of edits that will be necessary after they have been machine-translated (Scarton and Specia, 2016).

Since these four editing micro-tasks - deleting, inserting, moving and replacing words and multi-word units – seem to be at the centre of some of the most advanced attempts of using ML to further improve MT, we posited the hypothesis that post-editing could be defined by them, even if, in each real-life situation, these tasks are executed by translators recursively in the same sentence. If this hypothesis holds true, this would not only help interpret post-edited data and relate it to such tasks as QE, but it would also allow us to design tools that support the editing and post-editing procedures more efficiently.

3 Support by Augmented Intelligence

In the following sections, we will present a few proposals for the integration of existing technologies to support the procedures and tasks that we mention above. The management of the data, bilingual, unstructured, and scattered over various platforms that translators must collect and retrieve for each translation project has to be made in such a way that this data becomes "knowledge". Only then, will we have replaced CAT tools with KAT (Knowledge-Assisted Translation) tools.

Most of the technologies we present below belong to the domain of ML, more specifically in the area of Augmented Intelligence, a discipline which focuses on the connection between artificial and human intelligence (Schmitt, 1998). These tools are open-source or open-source based, which means that they may be easily integrated into a development or prototyping project. "R" (https://cran.r-project.org/) is an environment especially suited for prototyping this type of tools, since it links to several NLP toolkits and libraries.

3.1 Management

Management revolves around the notion of resources. We suggest that in an industry like translation, the main resource, which should be at the centre of all management tasks, is "knowledge".

Information Retrieval techniques may be integrated in the translation workflow to manage human and machine resources based on the content associated with each. This automation is easy to achieve as the translation process is text intensive and there are many ways to connect text with resources (the projects each person has translated, the quality rankings according to technical domain, etc.). Human resources (translators and revisers), TMs and MT models may be treated as individual "documents" that can be retrieved by queries, and then organized/clustered by domains and presented in visual maps. Beyond the traditional ranked list, incoming translation projects may act as queries, returning the answers in a visual map, which provides local and global contexts. Since humans are very efficient in processing visual information and obtaining insights regarding properties and their relationships, this enables an Augmented Intelligence approach, for which we suggest one of the two visual approaches presented below.

The first approach is the Multidimensional Scaling (MDS) representation, which transforms the n-term dimensions that characterize documents/resources in a 2D representation, providing a spatial proximity insight for each.

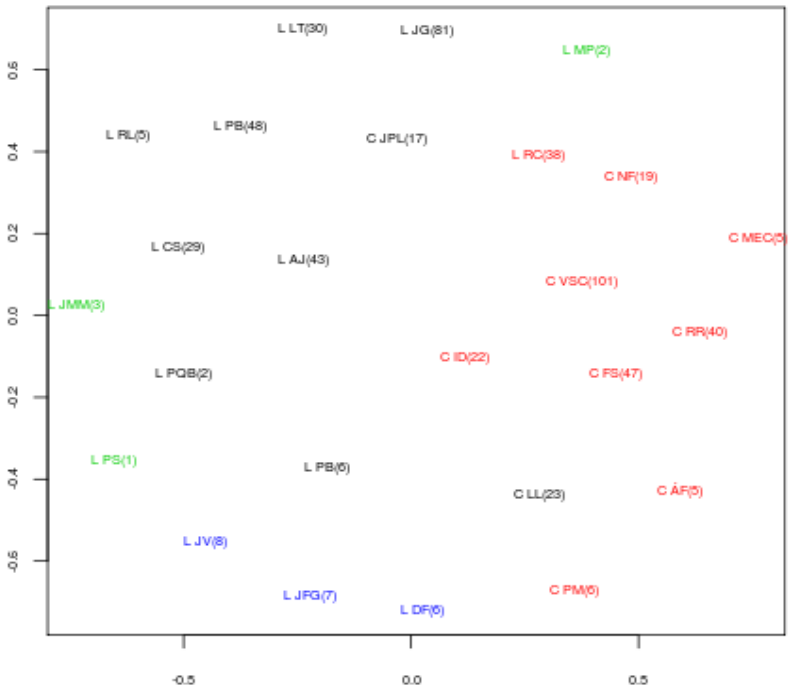


Figure 1: MDS representation of colour-coded clusters of documents/resources

The second option is the graph-based approach, which represents similarities between "documents" as links. Other visual hints may complement these approaches, like the colour, shape and size of the nodes/documents and the thickness of the links.

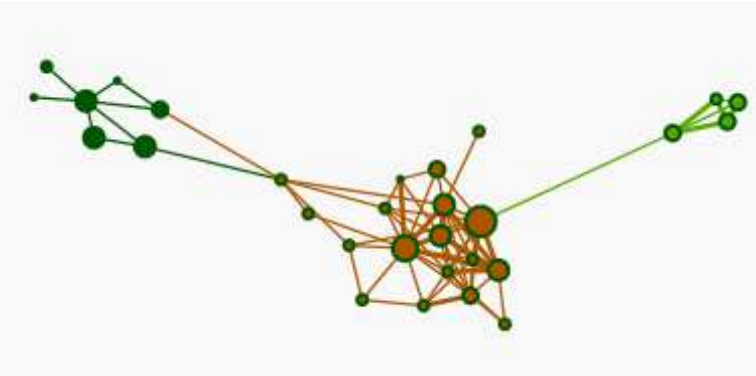


Figure 2: Network of an R&D unit/center

This concept is developed in the Affinity Miner project (Trigo et al., 2015), where it is applied to grasp affinity groups in and between publications of research centres. Document similarity is generated by a simple Bag of Words and vector representation (Feldman and Sanger, 2007) after performing standard text pre-processing. In this project, affinity groups/domains are extracted by community finding algorithms, but in this case we use a text similarity matrix that is broader than co-authorship. The algorithm that was selected was the Walktrap algorithm (Pons, 2005). This technique finds densely connected sub-graphs through random walks, assuming that short random walks tend to stay in the same community.

The importance of each document is highlighted visually by the size of the node corresponding to the number of publications and centrality within the graph. The centrality measure identifies the strongest element in each node, which could be, in our translation management environment, the translator with the highest rank in terms of number of words translated in a technical domain.

The Affinity Miner system can be complemented by resource allocation algorithms like the ones used by schools for timetable generation. The "geneticassigner" algorithm implementation (https://code.google.com/archive/p/geneticassigner/) "takes simple comma separated files with the available places for the different courses and the students lists of options for the courses, and allocates them into the courses with the best distribution it can find, trying its best to assign every student with their options priorities." In our translation project management environment, all we have to do is think of "courses" as "projects" and "students" as "translators" to understand how the system might work.

It is easy to see how a similar system might be used to turn the linguistic knowledge scattered among texts, TMs and other resources into the central pieces of a system that allocates the right contents to the right projects, and to the right people.

3.2 Research

In a knowledge-centred environment, TMs and references are clustered around domains and concepts, as the central piece of the research system. This research system should be closely linked to the translating/editing application.

To enhance the support to translators' research work, we should have specialised search engines, which log and find clusters of word searches, associating the most relevant sites for each search, text and domain, and improving future searches in the same context. The technologies that create this sort of "vertical search engines", specialised by domain, are available, and may even include web-crawlers to add relevant references, similar to those archived.

All this information may be integrated into the local knowledge base of the translator, or the translator company, or it may be shared with other participants in the translation process, such as the reviser, so that he may validate the decisions based on specialised searches.

Named Entity Recognition may be used to build stop lists, for terms that are not subject to translation by the MT engine. A toolkit like OpenNLP (https://opennlp.apache.org/) is appropriate for this task, since it extends this feature recognition to locations, dates and other elements that may be tagged for processing separately by the MT engine.

Another technology that may be used to support the research stage is the one related with terminology and concept extraction within technical domains, namely multi-word terms. A toolkit such as "mwetoolkit" (Ramisch, 2015) may be used us to integrate such features. Some of these tools also use visual representations, based on correlations.

To help us visualise clusters of words, as an improved alternative to traditional searches, we may use the same techniques used for documents, based on similarity on DTM (Document-Term Matrix). Well-documented techniques, such as Latent Semantic Analysis (LSA) may also help identify words associated with concepts.

Some of these technologies are alternatives for the same tasks (such as "community finding" – based on graph connections and "clustering" – based on similarity), but both ultimately enable users to identify networks of terms and concepts in a clear and visual way. So, with the added advantage that they are within easy reach for developers, this makes them very valuable for creating systems that try to extract knowledge from big volumes of data.

3.3 Editing

There are several theoretical and methodological issues to deal with before an editing tool can be designed that includes a support interface for the four editing micro-tasks, but we will not discuss those in this paper. Instead, to exemplify how such a tool might work, we selected a few very simple examples of translations from English into Portuguese, taken from real post-editing projects, to highlight the effect such an interface might have.

SOURCE	**MT SUGGESTION**	**POST-EDITED**
User Name/ID	Nome de utilizador	Nome/**ID** de utilizador
Patient Name/ID	Nome do paciente	Nome/**ID** do paciente
Item Name/ID	Nome do item	Nome/**ID** do item

Table 1: Examples of insertion

In the above example, we may see that the shortest edit to the MT translation suggestion was to insert "/ID" after the word "Nome". If this operation were learnt by the interactive editor application when the translator completes the first segment, the same edit could be automatically applied to the next two strings.

SOURCE	**MT SUGGESTION**	**POST-EDITED**
Acquire - to obtain possession of something	Adquirir - **para** obter a posse de algo	Adquirir - obter a posse de algo
Align - to place something in an orderly position in relation to something else	Alinhar - **para** colocar algo em uma posição ordenada em relação a outra coisa	Alinhar - colocar algo em uma posição ordenada em relação a outra coisa
Allocate - to divide something between different people or projects	Alocar - **para** dividir algo entre diferentes pessoas ou projetos	Alocar - dividir algo entre diferentes pessoas ou projetos

Table 2: Examples of deletion

The same process is visible in the sentences above, but with a "deletion" – the word "para" has to be deleted in the 3 examples, exactly in the same context, which is a fairly easy feature for an online learning tool to learn.

SOURCE	MT SUGGESTION	POST-EDITED
VEC 1 controller pin 7 (BK) wire	**Controlador VEC 1** fio do pino 7 (BK)	Fio do pino 7 (BK) do **Controlador VEC 1**
VEC 1 controller + (RD) wire	**1 Controlador VEC** + (RD)	Fio + (RD) do Controlador VEC 1
VEC 1 controller – (BL) wire	**VEC 1 controlador** - (BL)	Fio - (BL) do Controlador VEC 1

Table 3: Examples of shift

In the above example of a "shift" micro-task, when the post-editor moves the phrase "Controlador VEC 1" to the end of the first segment, it is informing the system that these three words form a unit, which translates "VEC 1 controller". This new sub-segment translation unit may be added to a dictionary of fixed translations that are assigned with a higher match percentage and reused to pre-translate the next segments.

SOURCE	MT SUGGESTION	POST-EDITED
Users must be set up and maintained at the console.	Os utilizadores têm de estar **configurado e mantido** na consola.	Os utilizadores têm de estar **configurados e mantidos** na consola.
Assess - to examine some-thing in order to judge or evaluate it	Avaliar - examinar algo para **juiz** ou avaliar	Avaliar - examinar algo para **ajuizar** ou avaliar
Act - to do something to change a situation	**Ato** - fazer algo para mudar uma situação	**Atuar** - fazer algo para mudar uma situação

Table 4: Examples of replacement

Finally, the "replacement" micro-task might be the solution to the edit effort necessary to correct the translation of each of the above three sentences. In the first one, the highlighted adjectival phrase must be replaced by the corresponding plural form. In the second example, a noun must be replaced by its corresponding verb, and vice versa in the last example. In a simpler implementation, this feature might be linked solely to the alternative translation suggestions in the translation tables of the MT systems (as GT, Lilt or CasMaCAT already show), but, in a more advanced system, the suggestions might be alternative inflected forms, especially important for morphologically-rich languages.

3.4 Revision and checking

The last stage of the translation implies a revision, either done by the translator himself, or by a different reviser. In order to be efficient and effective, this revision should not imply repeating the work done by the translator, especially at the research stage. So, a system that preserved some information from the research process done by the translator (in terms of references consulted, words researched and so on) might be a great help. Revisers would also benefit from computer aids for editing, which contextually showed alternatives to replacements, positions where the translator inserted words, and even edits made by the translator in fuzzy matches.

QA checking features usually generate long lists of false positives (because of very inflexible settings). Revisers should have better QA tools, which allowed them, for example, to exclude issues that are due to intentional actions, such as when the translator creates misaligned translation units due to problems in segmentation, or translates repeated segments differently due to new contexts.

Finally, at this stage, it would also be important to have some way of converting and importing instructions and style guides from clients into these QA checking tools. Accessible

tools such as LanguageTool (https://www.languagetool.org/), an open source tool that has a very simple interface for inserting customisable rules adaptable to style requirements of specific projects, may be easily integrated in a browser-based translation tool.

3.5 Learning and logging

A robust online learning system allows a system to learn as the user is typing words, so that the interactions are context-aware. This depends on how much information a system can log. An interface component that logs each micro-task as a specific event simplifies the logging and learning of the edits.

Bertoldi et al (2014) suggest that an efficient way to support the online learning processing is to work with "local translation models". This is a very appealing suggestion, since not only does it reduce the processing requirements, but it also emulates Project TMs that contain only part of the content of larger TMs. Finally, this type of system architecture also benefits from the local knowledge-base, and at the same time feeds it with new knowledge, thus helping the translator build his own archive of specialised content.

4 Conclusion

In this paper, we have tried to show how an analysis of the tasks and roles translators perform, together with an analysis of the technologies behind MT, may be a good foundation for the development of new and more efficient aids for the translation, revision and post-editing processes. In order for this (r)evolution to be achieved, it is fundamental that translation is handled not just by computers or machines, but by systems that manage knowledge and make it available in an efficient and effective manner.

Acknowledgements

We would like to thank all researchers with whom we have debated the themes presented in this paper, and who gave us such valuable input, especially to Chris Hokamp, who has agreed to adapt HandyCAT to test some of the features we describe.

References

Alabau, Vicent, Ragnar Bonk, Christian Buck, Michael Carl, Francisco Casacuberta, Mercedes García-Martínez, Jesús González-Rubio, Philipp Koehn, Luis A. Leiva, Bartolomé Mesa-Lao, Hervé Saint-Amand, Chara Tsoukala, Germán Sanchis-Trilles and Daniel Ortiz-Martínez. 2013. "Advanced Computer Aided Translation with a Web-Based Workbench". In *Proceedings of the 2nd Workshop on Post-Editing Technologies and Practice (WPTP*), pp. 53-62, 2013

Austermühl, Frank. 2013. "Future (and not-so-future) trends in the teaching of translation technology", *Revista Tradumàtica: tecnologies de la traducció, 2013*, 11 pp. 326 – 337. Universitat Autònoma de Barcelona

Bertoldi, Nicola, Patrick Simianer, Mauro Cettolo, Katharina Wäschle, Marcello Federico, and Stefan Riezler. 2014. "Online adaptation to post-edits for phrase-based statistical machine translation." in *Machine Translation 28*, 3-4 (December 2014), 309-339.

Carmo, Félix do and Belinda Maia. 2015. "Sleeping with the enemy? Or should translators work with Google Translate?" in Sánchez-Gijón, Pilar, Olga Torres-Hostench and Bartolomé Mesa-Lao, Bartolomé (eds.) *Conducting Research in Translation Technologies.* Oxford: Peter Lang.

Feldman, R. and Sanger, J. 2007. *The text mining handbook: advanced approaches in analyzing unstructured dat*a. Cambridge University Press.

Garcia, Ignacio. 2012. "A brief history of postediting and of research on postediting." in *Revista Anglo Saxonica,* vol 3, no 3. pp 291 - 310.

Green, Spence, Jason Chuang, Jeffrey Heer, and Christopher D. Manning. 2014. "Predictive Translation Memory: A Mixed-Initiative System for Human Language Translation". In *UIST.*

Hokamp Chris and Qun Liu. 2015. "HandyCAT". In: Durgar El-Kahlout İ, Özkan M, Sánchez-Martínez F, Ramírez-Sánchez G, Hollowood F, Way A (eds) *Proceedings of the 18th Annual Cnference of the European Association for Machine Translation (EAMT) 2015,* Antalya, p 216.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. "Moses: open source toolkit for statistical machine translation." in *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions (ACL '07).* Association for Computational Linguistics, Stroudsburg, PA, USA, 177-180.

Moorkens, Joss and Sharon O'Brien. (Forthcoming.) "Assessing User Interface Needs of Post-Editors of Machine Translation". in *IATIS Yearbook 2015/6.*

Mossop, B. 2007. *Revising and editing for translators* – 2nd Edition. Manchester: St. Jerome.

O'Brien, Sharon, Michel Simard and Lucia Specia (eds). 2012. *Proceedings of the AMTA 2012 Workshop on Post-Editing Technology and Practice (WPTP 2012).* Association for Machine Translation in the Americas (AMTA), Stroudsburg, PA

Pons, Pascal, & Latapy, Mathieu. 2005. "Computing communities in large networks using random walks." In *J. Graph Algorithms Appl.*, 10(2), 191-218.

Ramisch, Carlos. 2015. *Multiword Expressions Acquisition: A Generic and Open Framework*, Theory and Applications of Natural Language Processing series XIV, Springer.

Scarton, Carolina and Lucia Specia. 2016. *Quality Estimation of Machine Translation: Recent advances, Challenges and Software.* Tutorial presented at PROPOR2016. Tomar, Portugal

Schmitt, Gerhard N. 1998. "Design and construction as computer-augmented intelligence processes". in *CAADRIA '98: Proceedings of the Third Conference on CAAD*, Japan.

Screen, Benjamin. 2016. "What does translation memory do to translation? The effect of translation memory output on specific aspects of the translation process". in *Translation and Interpreting 8 (1)*, pp. 1-18.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., and Makhoul, J. 2006. "A Study of Translation Edit Rate with Targeted Human Annotation". in *7th Conference of the Association for Machine Translation in the Americas*, pages 223–231, Cambridge, Massachusetts.

Specia, Lucia, Dhwaj Raj, and Marco Turchi. 2010. "Machine translation evaluation versus quality estimation". in *Machine Translation 24, 1* (March 2010), 39-50.

Trigo, L., Víta, M., Sarmento, R., & Brazdil, P. 2015. "Retrieval, visualization and validation of affinities between documents". in *Ic3K 2015 - Proceedings of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*, Lisbon.

Automatic Bilingual Corpus Collection from Wikipedia

Mark Unitt
Capita Translation & Interpreting
Huddersfield Road
Delph

Simon Tite
Capita Translation & Interpreting
Huddersfield Road
Delph

Pejman Saeghe
Capita Translation & Interpreting
Huddersfield Road
Delph

{Mark.Unitt, Simon.Tite, Pejman.Saeghe}@capita.co.uk

Abstract

This is a study to combine a number of existing technologies with newly developed tools to create an automatic tool to assist with corpus collection for machine translation. This study aims to combine technologies for domain classification, domain source identification, and comparable file alignment into a unified tool. The unified tool will be used to make the corpora collection process more focused and efficient and enable a wider variety of sources to be used.

1 Introduction

The existing method for bilingual corpus collection is a slow manual process. It focuses on identifying suitable material in the source language using search engines and then locating "matching" material in the target language. If the matching material is absolutely parallel conventional tools such as Hunalign[1] tend to yield satisfactory results. However, for non-parallel resources, their output is far from useable, since these tools have been design for the purpose of aligning parallel sentences. This presents a problem and reduces the pool of material available for creating bilingual corpora.

Our goal is to create topic specific machine translation engines. In order to create these engines, there is need for large quantities of bilingual corpora on the given topic. In this paper we examine different technologies and tools that are already available and combine them with our in-house tools to create an automatic pipeline to generate these bilingual corpora.

2 Section

This section describes the steps involved in generating parallel segments from a set of keywords.

1) Describe the topic: The topic is delineated using keywords. Initially these keywords are identified manually from field experts. This process is then automated using topic modelling techniques (Wallach 2006).

2) Locate source material: Using Wiki API[2] each keyword is searched and a list of wiki pages related to that topic are collected in the source language.

3) Locate matching material: Each of the web pages from the previous step are checked to see if they contain a link to the same article in the target language. Only the pages that have a link to the target language are kept.

4) Download and pre-processing: Source and target articles are downloaded. The text content is extracted and tokenized.

5) Alignment: Pairs of matching pages are aligned using Yalign[3].

[1] http://mokk.bme.hu/en/resources/hunalign/
[2] https://www.mediawiki.org/w/api.php
[3] https://github.com/machinalis/yalign/tree/master

Proceedings of the 38th Conference Translating and the Computer, pages 159–163,
London, UK, November 17-18, 2016.

6) Cleansing: The aligned sentences are passed through a series of filters which use heuristic methods to discover bad alignments.

7) Evaluation: The final result is evaluated with help of linguists.

2.1 Topic classification

This will be a process using domain specific material to generate lists of domain specific terms. The domain specific material will simply be a set of plain text files which are known to have content related to the domain. The basis for this section of the tool will be topic classification technologies, which will be used to provided a list of terms that have been ranked by their compatibility to the domain. Inspiration for this stage of the process came from the very useful tool BootCat (Baroni and Bernardini, 2004). This tool takes a list of terms and generates an output file containing the content of web pages that match the term list, although this is limited to monolingual data.

For the purposes of the proof of concept we limited ourselves to a short initial set of manually generated seed terms saved in plain text file. An enhancement would be to automatically generate these terms using utilities such as NLTK[4] or MALLET[5]. The terms need to be specific to the topic and not be generic or open to a number of interpretations.

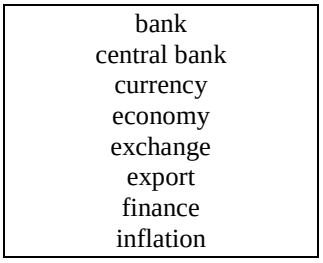


Figure 1Example term list

For a more substantive test a longer list would be extracted from a set of sample documents using MALLET. To simplify the use of MALLET a web based front end was created to control the parameters required by MALLET.

Once the term list is created, it will be converted into a number of multiple word tuples. The terms in the tuples are randomly generated with the limitation that no term is duplicated.

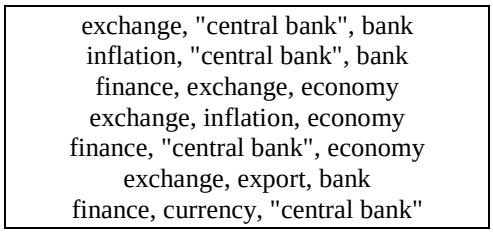


Figure 2 Example tuple list

For the purpose of providing a proof of concept we limited ourselves to seven tuples of three randomly selected terms each.

[4] http://www.nltk.org
[5] http://mallet.cs.umass.edu/topics.php

2.2 Domain source identification

Using the "tuples" generated by the previous process searches were made for web pages that match these tuples. For the initial proof of concept, the search was limited to Wikipedia and used the wiki API. For a wider search a search engine API, such as Google or Microsoft will have to be utilised. The search generated a list of Wikipedia page titles, using the search engine's API the process generated a list of URLs. A search using the seven sample tuples the search generated a candidate list of 3500 page titles. This list was then filtered to remove duplicate entries, reducing the list to 1938 titles.

2.3 Candidate URL pairing

The candidate list of page titles were processed individually to determine if a matching page existed in the target language. For the purposes of this sample exercise Spanish was chosen. Each candidate page title was processed with the Wiki API to determine if an "interlink" to a page with the Spanish language code existed. If such a page, existed it was added to the target page list. In order to test the system this search was limited to the first 50 source language titles and resulted in a target list of 22 page titles.

2.4 Candidate File downloading

Each of the files identified as having a matching target file was downloaded together with its Spanish equivalent. These files were then processed to leave just the main paragraph copy in a plain text format. Additionally any Wikipedia link and reference HTML codes were also removed. Every file was also given an additional two letter language code as a prefix to its filename.

2.5 Comparable file alignment:

The comparable alignment tool identified was Yalign. The concept behind comparable alignment is that each segment in the source file is compared against the segments in the target file. The best matching segment pairs being returned as output.

The content of the paired URLs is aligned using comparable file alignment techniques to produce segment based corpora. This process will divide the content, both source and target, into individual segments using standard text segmentation structures. The segments will then be aligned based on a simple language model and a number of string comparison techniques. The end result will be a comparability rating to each segment combination. The segment combinations that exceed the threshold will be exported for further processing, the remainder will be discarded. This is explained in more detail see (Wolk and Marasek, 2015).

In order to improve the processing of batches of files, some of the file handling routines in Yalign were separated into individual processes. These included the routine to segment the text into individual segments.

Comparing alignment technologies.

Before committing completely to a single choice of comparable alignment tool an alternative was investigated. An alignment tool was written based on the method outlined by (Mohammadi and QasemAghaee, 2010). This tool, known internally as Palign, was used as a comparison to Yalign. The two tools were tested on the same data sets and the results compared. The data was taken from a number of previously translated files. The test "source" files contained a randomised sequence of text segments from a number of test files, whilst the "target" files were those returned by human translators.

Sample	Method	# correctly aligned segments	sample size#	% of segments aligned
1	Yalign	34	84	40.47619
	Palign	23	84	27.38095
2	Yalign	32	85	37.64706
	Palign	10	85	11.76471

Figure 3. Comparative examples of different alignment tools

Although the test was limited these results demonstrated that the Yalign tool would produce the best results.

2.6 Combining the tools into one processing

Each of the components of this process were developed into a standalone module, these modules were then joined into a pipeline. In this set up the output of one module becomes the input of the next module in the pipeline. The output of the individual modules were also recorded for reporting and error checking purposes.

2.7 Corpora evaluation and clean up:

The produced corpora are evaluated both mechanically and through the use of human linguists. Poor quality segments will be discarded. The file format sent to the linguists will be in a simple excel spreadsheet. The linguists will be asked to rate the segments on a basic scale. As there may be very large numbers of segments to be rated, the segments will be divided into batches based on the source URLs. The linguists will be asked to rate the segments at a batch level.

Human evaluation

A set of 800 segment pairs were sent to a human linguist for scoring. The linguist was asked to rate each aligned pair as 1: not acceptable; 2: just acceptable and 3: acceptable. The initial result are as follows:

Rating	# segments	%
1	285	35.54%
2	50	6.23%
3	467	58.23%

Figure 4. Human evaluation of initial alignment

A 36% "noise" ratio is not good enough for the data to be used as a source for building an engine. To improve on this, a series of filters will have to be created to reduce this noise ratio.

Clean up and noise reduction

We have developed some rough filters for removing what appear to be misaligned sentences. These checks currently include

Mismatched segment length:	Translating a sentence from one language into anther will change the length of that sentence, however both the source and translated sentences will generally be of a comparable length. The filter will remove sentences where the relative length exceeds the parameter.
Mismatched numeric tokens:	The filter will count the number of groups of digits and will remove any segments where the number of digit groups differ.
Mismatched non-alphanumeric characters:	This group of characters includes punctuation, brackets and similar characters. The filter will remove those segment pairs where the ratio between the source and target non-alphanumeric characters exceeds a pre-set parameter.

Human Evaluation			**Automatic checks to filter out misaligned segments**					
			Matching Number tokens		**Matching Number tokens and strict length ratio**		**Matching Number tokens length ratio, non-alphanumeric characters**	
Rating	**# segments**	**%**	**# human rated segments left**	**%**	**# human rated segments left**	**%**	**# human rated segments left**	**%**
1	285	35.54%	132	21.85%	72	17.87%	77	16.67%
2	50	6.23%	40	6.62%	8	1.99%	13	2.81%
3	467	58.23%	432	71.52%	323	80.15%	372	80.52%

Figure 5. Effects of different filers

Applying these filters reduces the proportion of "noisy" to a more acceptable level. Further filters may be considered in the future. The goal will be to reduce this noise proportion to ideally below 10% and ultimately below 5%.

2.8 Conclusion

The work done so far has indicated that the concept is workable and will produce aligned material that can be used for MT engine creation. There are areas for improvement and further experimentation. There are for instance a number of parameters within Yalign that need to be explored that have an effect on the number and quality of the outputs provided.

We will also want to conduct a large scale test to generate a sufficiently large output set to be able to generate a machine translation engine. The output from this engine would then be compared against the output of an engine created by more "traditional" methods.

References

H.M. Wallach. 2006. ‘Topic modeling’, Proceedings of the 23rd international conference on Machine learning - ICML ’06, . doi: 10.1145/1143844.1143967

M. Baroni and S. Bernardini. 2004. BootCaT: Bootstrapping corpora and terms from the web. Proceedings of LREC 2004.

K. Wołk and K. Marasek. 2015. "Unsupervised comparable corpora preparation and exploration for bi-lingual translation equivalents".

M. Mohammadi and N. QasemAghaee. 2010. "Building Bilingual Parallel Corpora based on Wikipedia".

Calculating the Percentage Reduction in Translator Effort when using Machine Translation

Andrzej Zydroń
XTM International Ltd.,
UK
azydron@xtm-intl.com

Qun Liu
Dublin City University,
Ireland
qliu@computing.dcu.ie

1 Introduction

At present there is no precise indication of the benefits of using Statistical Machine Translation (SMT) for potential users. The question 'is this going to save me time and/or money' and if so how much, is not addressed in any systematic way. The common answer provided by most SMT service providers is 'well, it depends'. This is far from the answer that users need to make an informed decision about whether to go ahead with SMT.

What is lacking in the industry today is a description of the main factors affecting the quality of SMT output and how you can use them to provide an indication of the savings that SMT will provide. In the end, the decision on whether to use SMT depends on the amount of time saved during translation. This paper provides a clear indication of the savings you can expect, depending on the key factors that affect the quality of the SMT, based on a simple calculation that provides a Percentage Reduction in Translator Effort (PRTE) that can be expected for a given localization project.

2 Translation Cost

Translation forms part of the cost of localization, and it is often all too easy to forget about the other elements of the overall localization process and subsequent costs. In fact translation itself typically accounts for only between 30% to 50% of the overall cost of a localization project, depending on how much automation is involved in the overall localization workflow. The following diagram shows the standard cost model for a manual localization process:

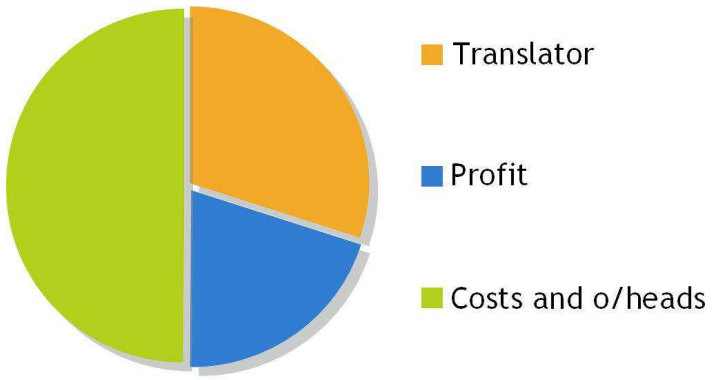


Figure 1: typical localization cost breakdown Prof. Reinhard Schäler ASLIB 2002

As can be seen from the diagram translation itself forms only part of the cost of localization. The other costs, apart from the profit made by the localization service provider, are the management and administrative costs, as well as proofreading, review and correction. An automated translation management system (TMS) can significantly reduce the administrative and management costs of the localization process.

Proceedings of the 38th Conference Translating and the Computer, pages 164–169,
London, UK, November 17-18, 2016.

3 PRTE Calculation

Having put the cost of translation into perspective we can now look at the factors that affect the quality of SMT and consequently the Percentage Reduction in Translator Effort (PRTE).

PRTE can be defined as: The percentage reduction in translator effort by using SMT compared to human translation on its own.

PRTE is the key factor that decides how much savings you can expect to gain from SMT for a given project. The quality of SMT is governed by three major factors:

1. The language closeness (LC): the similarity of the source and target languages in terms of morphology, word order and grammar
2. The amount of training data
3. The relevance of the training data to the current text being translated

If we provide mathematical weightings to these factors we can use them very effectively to provide a calculation of the percentage translator productivity we can expect to achieve using SMT. In order to provide a percentage, we will use a probability type estimation for each factor with a range of 0 to 1, where the value '1' assumes an idealized perfect situation and '0' the opposite.

Let us now consider these factors in detail:

Language Closeness

SMT output is affected by the by the differences between the source and target languages in terms of various aspects, including grammar, word order and morphologies. To put it simply, the closer the two languages are in terms of grammar and word order and morphology, then the better the outcome. To take an extreme case, of say, US English to UK English we can state that the LC is '1.0' as the two variations of English only differ in some spelling instances. Using English as the source again and this time French as the target we can assume a LC value of 0.8, as both languages have similar primitive morphologies and word order. For English to German, we would use a value of 0.6 as the differences in morphology and word order are much more pronounced. For English to Russian or Polish the proposed value would be 0.45 and for English to Japanese it would be 0.25, as there are significant differences in word order and morphology between the two languages.

A good indication of the difference in language models can be found at: http://esl.fis.edu/grammar/langdiff/ - this site provides a comparison for some major languages concerning the difficulties that native speakers of those languages have in learning English. The degree to which these students have issues with learning English is also indicative of the basic differences in grammar and morphology between their native tongue and English and also indicative of the difficulties posed in terms of SMT between English and those languages.

The following table provides an indication of the types of factor where English is the source language. The factors have been arrived at from personal experience and should require further investigation, but they are a good starting point:

Language Closeness factors relative to English	
French	0.800
Spanish	0.775
Portuguese	0.775
Italian	0.760
Dutch	0.750
Swedish	0.700
Danish	0.650
German	0.600
Arabic	0.600
Korean	0.500
Finnish	0.500
Hungarian	0.500
Turkish	0.500
Polish	0.450
Russian	0.450
Czech	0.450
Slovak	0.450
Chinese	0.400
Japanese	0.250

Table 1. LC factors relative to English

If all other factors affecting SMT quality are in an ideal state, then the expected productivity improvement, where the LC is the only factor, then the following graph shows the expected productivity improvement where English is the source language, depending on the target language:

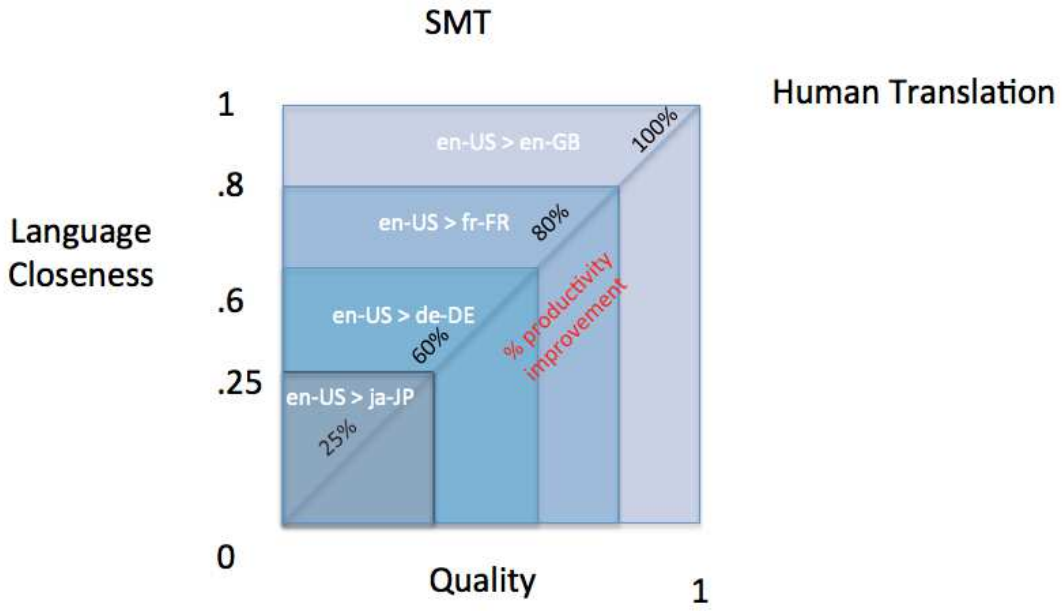


Figure 2: Idealized PRTE for SMT only considering LC factors

Training Set Size Factor (TSSF)

The next key factor regarding SMT quality is the size of the training set. Too small a TSS and there will not be enough data to provide an adequate model for translation. When there is no training data, the TSS should be 0 As the size of data increases the TSS approaches 1, so when the TSS is 1 there is infinite training data. We use the equation below to estimate TSS:

$$TSSF = 1 - 2^{-\frac{Size}{Size'}}$$

Where $Size$ is the actual training data size and $Size'$ is an empirical value which makes TSSF equal 0.5.

What this means is that a training set size of $Size'$ would result in a reduction of the translation effort of 50%. In practical terms this would normally equate to around 50,000 segments, depending on the material being translated. A training set size of 10,000 segments would produce a TSSF of .067 whereas 100,000 segments would result in a TSSF of .75 and 200,000 segments would produce a TSSF of .9375.

The training set size parameters can be adjusted according to the specific requirements of the scenario and how much training data is actually available as opposed to the theoretical optimal amount.

Using the above assumptions, as a very rough rule of thumb normally, you can assume that an optimal training set size of 250,000+ segments would provide a TSS value of approaching 1. Anything less would result in reducing the TSS value roughly by 0.1 for every reduction of 25,000 segments in the training set size.

A constant problem with SMT is the issue of out of vocabulary (OOV) words: these are words that have not been encountered previously in the training set. If the training set size is too small then you can expect a commensurate increase in OOV word instances and therefore more work for the translator.

For the purposes of the PRTE calculation we can assume again a value of between 1 (ideal training set size) and 0 (no training set). Zero would be improbable value (we would not be

able to build a SMT engine with no training data), but we can see that if not enough training data is available it would have significant impact on the quality of the SMT.

Domain Similarity (DMS)

Empirical evidence has shown that the quality if SMT also depends on the quality of the training set. A smaller training set on the same topic domain produces much better results than using a generalized training set. Specific domains have their own vocabulary and phraseology that cannot be rendered with a general SMT engine.

For the purposes of the PRTE calculation we can assume a value between 1 (exactly the same specific domain from data for exactly the same organization) and 0 a completely unrelated specific domain. A generic SMT engine would rate 0.25 where the subject matter being translated related to a highly specific domain with its own detailed terminology.

PRTE Formula

The PRTE formula itself takes all three of the above factors to provide an overall calculation that is easy to implement:

$$PRTE = (LC \times TSSF \times DMS) \times 100\%$$

Figure 3: PRTE formula

This can be represented by a three dimensional graph as follows:

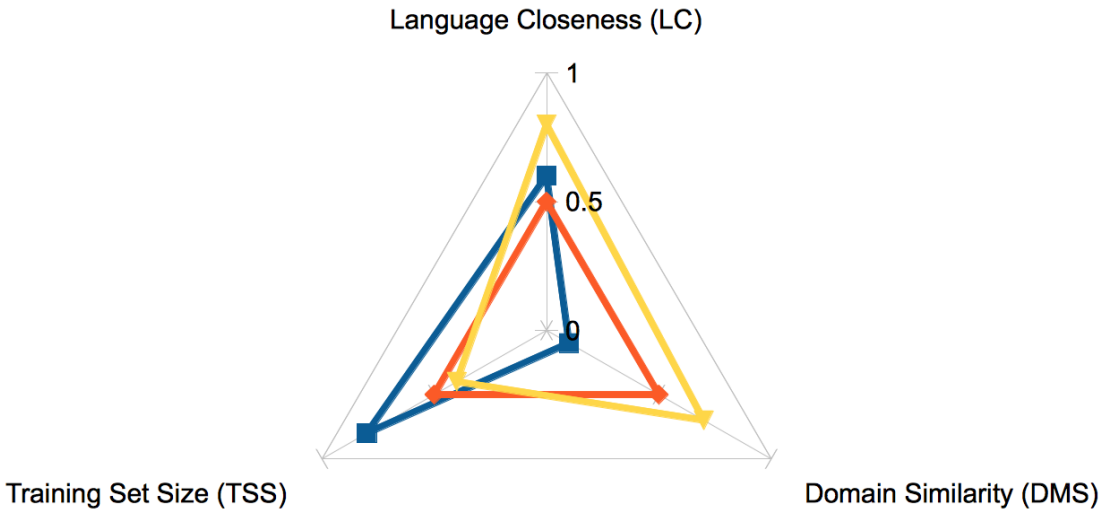


Figure 4: PRTE 3 dimensional graph for various PRTE calculations

To test the validity of the formula we can try some examples:

1. Translating from en-US to en-GB we can assume a LC[1] value of 1. If we have an ideal reference TSSF[2] of 1 and an ideal DMS[3] of 1, we arrive at a PRTE of:

   1x1x1x100 = 100%

1 Language Closeness
2 Training Set Size Factor
3 Domain Similarity

This would mean that the SMT[4] output should require no translator intervention providing a productivity figure of 100%.

2. Translating from en-US to fr-FR we can assume a LC[1] value of 0.8. If we have a slightly less that ideal TSSF[2] of 0.75 but with an ideal DMS[3] of 1, we arrive at a PRTE of:

   0.8x0.75x1x100 = 60%

   This would mean that we should expect an improvement regarding translator productivity of 60% compared with a completely manual human translation.
3. Translating from en-US to ja-JP we can assume a LC[1] value of 0.2. If we have an ideal TSSF[2] value of 1 and an ideal DMS[3] of 1, we arrive at a PRTE value of:

   0.2x1x1x100 = 20%

This would provide an estimated 20% improvement in translator productivity.

4 Conclusion

The PRTE formula is not designed to be a hard and fast assessment of the expected percentage reduction in translator effort, but rather an overall rough estimation of what can be expected. Some of the figures are expected to be at best a 'guess' as regards the DMS and TSS figures. The LC values are also a rough approximation and some SMT systems with an appropriate amount of tuning will be able to provide better values. It also does not take into account the differences between individual SMT engines: some will inevitably be better than others. The amount of manual tuning also needs to be taken into account as it requires the input of highly skilled engineers.

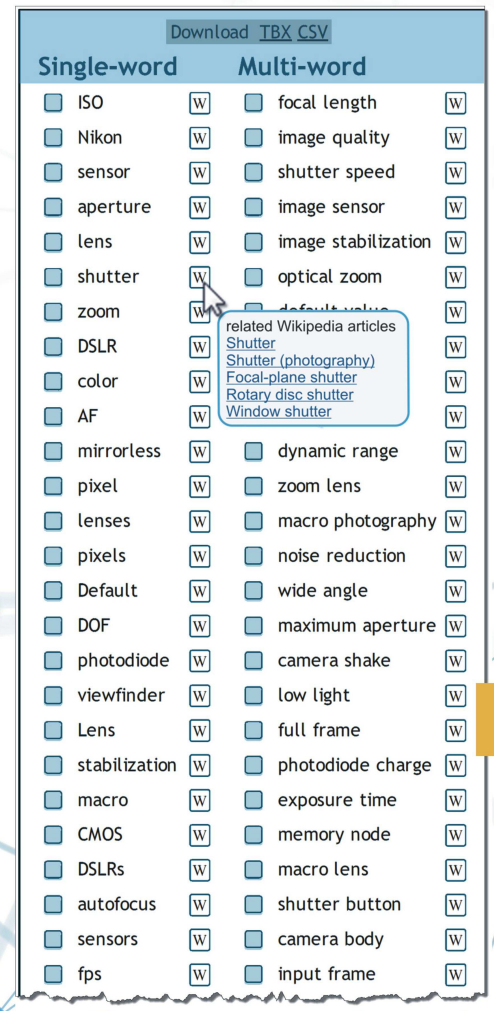
Nevertheless the PRTE formula provides a guide to what is achievable for a given situation and roughly an idea of the returns that can be expected. This is vastly better than nothing, or 'well it depends' which is the current situation.

[4] Statistical Machine Translation

Author Index

next generation

term extraction



Sketch Engine combines statistics and linguistics to deliver the ultimate quality in term extraction.

Sample terms extracted from texts about photography. No manual cleaning or post-processing applied.

Sketch Engine

www.sketchengine.co.uk

matecat™

More Matches for your Translations

Free and open source enterprise-level translation software

From 10% to 20% more matches than any other CAT tool

Increased privacy, no more files via email

A professional tool for language service providers and MT specialists

Collect data
to **set a fair rate**
for post-editor
and improve MT quality

Real-time **progress**
report and quality
control for your
translations

Online adaptation
and quality estimation
for **MT systems**
based on Moses

Connect your Moses MT system via a set of open and easy to use API

Start translating

www.matecat.com

Translated

The MateCat project has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no 287688.

SDL Trados Studio 2017

Designed to make the difference

Transformational technology

AdaptiveMT, upLIFT Fragment Recall and upLIFT Fuzzy Repair

#Trados2017

www.sdl.com/trados2017
www.translationzone.com/translator/trados2017
www.translationzone.com/lsp/trados2017

SDL Trados Studio 2017

Next year's conference:

Translating and the Computer 39

will be organised again by

for 16-17 November 2017
in London (UK)

Please highlight these dates in your diary.

For information on next year's **39th Translating and the Computer** conference, **TC39**, please check

http://asling.org

for how and when to submit proposals for talks, workshops and posters, along with other useful information, as these becomes available.

TC39 will have a special session with a strong focus on **technology tools for interpreters**.