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.

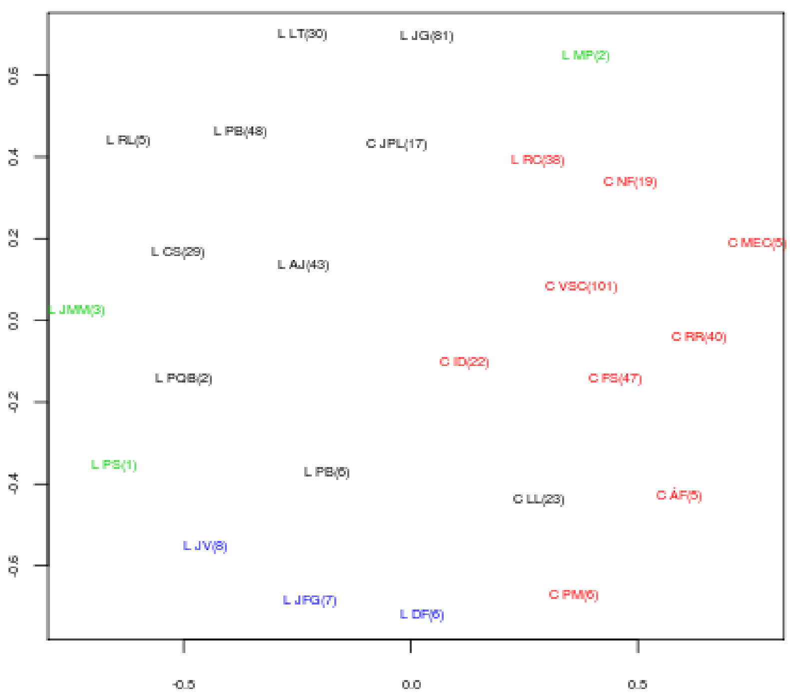

Figure 1: MDS representation of colour-coded clusters of documents/resources

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.

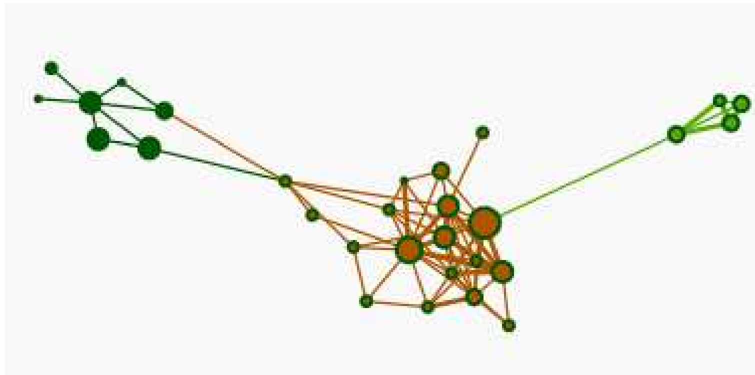

Figure 2: Network of an R&D unit/center

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 something 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.