An open-source web-based tool for resource-agnostic interactive translation prediction

Daniel Torregrosa, Mikel L. Forcada, Juan Antonio Pérez-Ortiz

Departament de Llenguatges i Sistemes Informàtics, Universitat d'Alacant, Spain

Abstract

We present a web-based open-source tool for interactive translation prediction (ITP) and describe its underlying architecture. ITP systems assist human translators by making context-based computer-generated suggestions as they type. Most of the ITP systems in literature are strongly coupled with a statistical machine translation system that is conveniently adapted to provide the suggestions. Our system, however, follows a *resource-agnostic* approach and suggestions are obtained from any unmodified *black-box* bilingual resource. This paper reviews our ITP method and describes the architecture of Forecat, a web tool, partly based on the recent technology of *web components*, that eases the use of our ITP approach in any web application requiring this kind of translation assistance. We also evaluate the performance of our method when using an unmodified Moses-based statistical machine translation system as the bilingual resource.

1. Introduction

Translation technologies are being increasingly used to assist human translators. Within this context, the objective of interactive translation prediction (ITP) tools (Foster et al., 1997; Barrachina et al., 2009) is to assist human translators in the translation of texts for dissemination by making context-based computer-generated suggestions as they type. Most works in the field of ITP have solely used specifically-adapted statistical machine translation (SMT) systems to obtain the suggestions. On the contrary, the *resource-agnostic* approach considered for the tool described in this paper explores how non-adapted *black-box* bilingual resources of any kind (a machine translation sys-

tem, a translation memory, a bilingual dictionary, etc.) can be accommodated into an interoperable ITP framework.

This paper reviews the main aspects of our method and describes the architecture of Forecat, an HTML5 web tool, based on the recent technology of *web components* (see section 5.2), that eases the use of our ITP approach in any web application requiring this kind of translation assistance. To our knowledge, this is the first ITP tool that has been programmed as a web component.

The remainder of the paper is organised as follows. After reviewing the state-of-the-art in ITP in Section 2, we outline our resouce-agnostic proposal in Section 3. We then present unpublished results for a fully automatic evaluation of our approach in Section 4. After that, the architecture of an open-source web tool that implements our method is discussed in Section 5. Finally, we draw some conclusions in Section 6.

2. Related work

The systems which have most significantly contributed to the field of ITP are those built in the pioneering TransType project (Foster et al., 1997; Langlais et al., 2000), and its continuation, the TransType2 project (Macklovitch, 2006; Barrachina et al., 2009). These systems observe the current partial translation already typed by the user and, by exploiting an embedded SMT engine whose behaviour is modified to meet the needs of the ITP system, propose one or more continuations for the next words (or even the complete remainder of the sentence) that are compatible with the current sentence prefix. An automatic best-scenario evaluation (Barrachina et al., 2009) showed that it might theoretically be possible to use only 20–25% of the keystrokes needed in unassisted translation for English–Spanish translation (both directions) and around 45% for English–French and English–German. The results of user trials (Macklovitch, 2006) showed gains in productivity (measured in number of words translated per hour) of around 15–20%. A number of projects (Koehn, 2009; Ortiz-Martínez, 2011; Alabau et al., 2010, 2013) have recently continued the research where TransType2 left it off. As regards tools, both Caitra (Koehn, 2009) and the CASMACAT Workbench (Alabau et al., 2013) are web applications with an underlying SMT-based ITP system. Caitra consults the internal translation table of the SMT system Moses (Koehn et al., 2007) in order to show the most likely translation options to the human translator. Users can freely type the translation or accept any of the context-based suggestions generated *on the fly*: the most likely completion is directly shown next to the input field; the other translation candidates are also included in the interface so that users may click on any of them in order to incorporate it to the translation. The CASMACAT Workbench

also makes use of inner elements of Moses to offer assistance in the form of ITP, interactive editing with confidence information and adaptive translation models.[1]

Finally, many commercial translation memory systems also include the possibility of using ITP (see, for example, MT AutoSuggest,[2] a plug-in for the SDL Trados Studio tool).

3. A resource-agnostic interactive translation prediction approach

We propose a black-box treatment of the bilingual resources in contrast to the glass-box approaches found in literature. Unlike in the latter, access to the inner details of the translation system is consequently not necessary; this resource-agnostic approach minimises the coupling between the ITP tool and the underlying system and provides the opportunity to incorporate additional sources of bilingual information beyond purposely-designed SMT systems. These resources may include bilingual resources that cannot be adapted to produce a continuation for the remainder of the target-language sentence given a sentence prefix (for instance, because their code cannot be accessed or because they do not provide full sentence translations), but are able to supply the translation of a particular source-language segment. The underlying idea behind our approach is that resources such as machine translation (MT) cannot usually deliver appropriate translations at the sentence level, but their proposals usually contain acceptable segments that do not cover the whole sentence but which can be accepted by the user to assemble a good translation, saving as a result keystrokes, mouse actions or gestures, and, possibly, time. Note that by using the bilingual resources as black boxes that just provide translations, our system could be deprived of additional features that could prove useful if access to the inner details of the resource was possible.

A complete description of the method can be found in the paper by Pérez-Ortiz et al. (2014). What follows is an overview of its most relevant aspects.

Potential suggestions. Our method starts by splitting the source-language sentence S up into all the (possibly overlapping) segments of length $l \in [1, L]$, where L is the maximum source segment length measured in words.[3] The resulting segments are then translated by means of one or more bilingual resources. The set of *potential*

---

[1]Our tool is currently in an early stage of development, but some of these features could be incorporated into it in later stages. Note, however, that one of the major premises behind its development is to keep it as simple as possible so that it can been easily deployed as a standalone web component.

[2]http://www.codingbreeze.com/products/mtautosuggest/autosuggest_overview.htm

[3]Suitable values for L will depend on the bilingual resource: on the one hand, we expect higher values of L to be useful for high-quality MT systems, such as those translating between closely related languages, since adequate translations may stretch to a relatively large number of words; on the other hand, L should be kept small for resources such as dictionaries or low-quality MT systems whose translations quickly deteriorate as the length of the input segment increases.

proposals P^S for sentence S is made up of pairs comprising the translation of each segment and the position in the input sentence of the first word of the corresponding source-language segment. For example, the source-language segments obtained when translating the English sentence $\mathsf{S} = $ "My tailor is healthy" into Spanish with $\mathsf{L} = 3$ are *My*, *My tailor*, *My tailor is*, *tailor*, *tailor is*, *tailor is healthy*, *is*, *is healthy*, and *healthy*; the corresponding set of potential suggestions P^S is made up of (*Mi*, 1), (*Mi sastre*, 1), (*Mi sastre es*, 1), (*sastre*, 2), (*sastre es*, 2), (*sastre está sano*, 2), (*es*, 3), (*está sano*, 3), and (*sano*, 4). We shall represent the i-th suggestion as p_i, its target-language segment as $\mathsf{t}(\mathsf{p}_\mathsf{i})$ and its corresponding source-language word position as $\sigma(\mathsf{p}_\mathsf{i})$.

Compatible suggestions. Let $\mathsf{P}^\mathsf{S}_\mathsf{C}(\hat{w})$ be the subset of P^S including the *compatible suggestions* which can be offered to the user after typing $\hat{w}$ as the prefix of the current word in the translated sentence T. The elements of $\mathsf{P}^\mathsf{S}_\mathsf{C}(\hat{w})$ are determined by considering only those suggestions in P^S that have the already-typed word prefix as their own prefix:

$$\mathsf{P}^\mathsf{S}_\mathsf{C}(\hat{w}) = \{\mathsf{p}_\mathsf{i} \in \mathsf{P}^\mathsf{S} : \hat{w} \in \text{Prefix}(\mathsf{t}(\mathsf{p}_\mathsf{i}))\}$$

For example, in the case of the translation of the previous English sentence, if the user types an *M*, the set of compatible suggestions $\mathsf{P}^\mathsf{S}_\mathsf{C}(M, 1)$ will contain the suggestions with target-language segments *Mi*, *Mi sastre* and *Mi sastre es*, since they are the only proposals in P^S starting with an *M*.

Except for very short sentences, the number of compatible suggestions usually exceeds what users are expected to tolerate. Therefore, adequate strategies are necessary in order to reduce the number of suggestions eventually offered to the user to an appropriate value. The degree of success that can be achieved in this task will be explored in greater depth in future work, but a naïve *distance-based* approach that ranks the suggestions in $\mathsf{P}^\mathsf{S}_\mathsf{C}(\hat{w})$ based solely on the position j of the current word in the target sentence has already provided interesting results (Pérez-Ortiz et al., 2014).

Ranking suggestions. Under the distance-based approach, suggestions p_i whose source position $\sigma(\mathsf{p}_\mathsf{i})$ is closer (in terms of the absolute difference) to the position j in the target sentence of $\hat{w}$ are prioritised.[4] For example, in the case of the translation mentioned above, if the user has just typed *Mi s* and is introducing the second word of the translation, suggestions starting with *sastre* (originated at source position 2) will be ranked before those starting with *sano* (originated at position 4).

Let M be the maximum number of suggestions that will eventually be offered to the human translator. For the small values of M that are acceptable for a user-friendly

---

[4]We cannot in principle expect this ranker to work reasonably well on unrelated languages with very divergent grammatical structures (e.g., when translating a language with a verb–subject–object order into another one with a subject–verb–object order).

interface, it may easily happen that all the suggestions offered are obtained starting at the same source position (that closest to the current target position) although better suggestions from different positions exist. In order to mitigate the impact of this, the *distance-based* ranking is partially relaxed in the experiments in this paper in a similar way as proposed by Pérez-Ortiz et al. (2014): only the longest and the shortest suggestions from each position are in principle chosen; the rest of the suggestions, if any, would only be offered if the number of maximum offered suggestions M is not reached after exhausting the longest and shortest proposals from all compatible positions.

4. Experiments

Although the main purpose of this paper is to introduce the architecture of our ITP tool, in this section we show the results of an automatic evaluation carried on using a black-box phrase-based SMT system as the only bilingual resource. This evaluation will provide an idea of the best results attainable with our method by human translators. The approach followed for the automatic evaluation is identical to that described by Langlais et al. (2000), in which a parallel corpus with pairs of sentences was used. In the context of our automatic evaluation, each source-language sentence S is used as the input sentence to be translated and the corresponding target-language T is considered as the output a user is supposed to have in mind and stick to while *typing*. The longest suggestion in the list of offered suggestions which concatenated to the already typed text results in a new prefix of T is always used. If there are no suggestions at a particular point, then the automatic evaluation system *continues typing* according to T. The performance of our system is measured by using the *keystroke ratio* (KSR), that is, the ratio between the number of keystrokes and the length of the translated sentence (Langlais et al., 2000).[5]

We have evaluated whether the domain of the corpora used to train the Moses-based SMT system (Koehn et al., 2007) affects the KSR. For this, the phrase-based SMT systems have been trained in the standard way, more exactly as described by Haddow and Koehn (2012).[6] Only the best translation proposed by Moses for each segment has been considered in our experiments. Tests have been performed with 3 different systems: one trained and tuned with out-of-domain corpora, another trained with out-of-domain corpora but tuned with in-domain corpora, and a third one trained and tuned with in-domain corpora.

Both L and M are set to 4 following the recommendations from previous automatic evaluations (Pérez-Ortiz et al., 2014). On the one hand, L = 4 represents a good compromise between computational load and usefulness of the suggestions; only marginal

---

[5]A lower KSR represents a greater saving in keystrokes.

[6]The method for the compression of translation phrase tables proposed by Junczys-Dowmunt (2012) has also been used.

performance improvement is attained when incrementing it, with the drawback of having more equivalents to obtain through the bilingual resources, more suggestions to filter out, etc. On the other hand, the performance with $M = 4$ has proved to be close to that obtained considering all the possible candidate suggestions without posing a significant hindrance for the human translators.

Corpora.   From all the pairs considered in the work by Haddow and Koehn (2012)[7], English–Spanish (en–es) and English–Czech (en–cs) were chosen (in both translation directions), as they are, respectively, the best and worst performing pairs in that work. For English–Czech and English–Spanish, we used the most up-to-date version of the corpora used by Haddow and Koehn (2012): the v7 release of the Europarl corpora.[8] and the ACL2013 News Commentary corpora[9]. 2 000 sentences from each corpora were selected as tuning set, and 15 000 from News Commentary were extracted as test. The rest of the corpora was used as training set: in the case of Europarl, 623 913 sentences for English–Czech, and 1 912 074 for English–Spanish; in the case of News Commentary, 122 720 sentences for English–Czech and 155 760 for English–Spanish.

Results.   The KSR values for the automatic evaluation are shown in table 1. As expected, English–Spanish performed better (savings in keystrokes up to 48%) than English–Czech (savings in keystrokes up to 31%) because it is generally easier to translate between English and Spanish, and the available corpora were larger in this case as well. Though statistically significant, the differences between the different in-domain and out-of-domain systems are relatively small. For the purposes of comparison, the rule-based MT system Apertium (Forcada et al., 2011) has been reported (Pérez-Ortiz et al., 2014) to provide a KSR of 0.76 for English–Spanish and 0.70 for Spanish–English when using, in both cases, $L = 4$ and $M = 4$; note, however, that Apertium is an already-built *general-purpose* MT system, which makes it impossible to differentiate between results for in-domain or out-of-domain scenarios.

5. Technical Issues and Implementation

In this section we describe Forecat, an open-source web-based tool that we have implemented to demonstrate the validity of our ITP approach and incorporate its use in real-life applications. Forecat can be used in three different ways as described next.

Firstly, it can be used as a simple web application for computer-assisted translation. Under this perspective, our tool has a web interface similar to that in the projects

---

[7]In their paper, the influence of in-domain and out-of-domain corpora when training SMT systems was evaluated.

[8]http://www.statmt.org/europarl/

[9]http://www.statmt.org/wmt13/translation-task.html

	en–es	es–en	en–cs	cs–en
ep	0.70	0.53	0.78	0.69
ep+nc	0.62	0.52	0.75	0.66
nc	0.62	0.52	0.76	0.64

Table 1. KSR values from the automatic evaluation of our ITP method. In the table, ep *stands for the system trained and tuned with Europarl (out-of-domain);* nc *for the one trained and tuned with News Commentary (in-domain); and* ep+nc *for the one trained with Europarl but tuned with News Commentary. In all cases, the test set consisted of sentences extracted from the News Commentary corpus. For the purposes of comparison, the rule-based MT system Apertium (Forcada et al., 2011) has been reported (Pérez-Ortiz et al., 2014) to provide a KSR of 0.76 for English–Spanish and 0.70 for Spanish–English.*

discussed in Section 2: users freely type the translation of the source sentence, and are offered suggestions *on the fly* in a drop-down list with items based on the current prefix; users may accept these suggestions (using cursor keys, the mouse or specific hot keys) or ignore them and continue typing. A screenshot of the interface is shown in Figure 1. Despite the cognitive load inherent to any predictive interface, the interface is easy and intuitive to use, even for inexperienced users, as can be deduced from the results of a preliminary user trial (Pérez-Ortiz et al., 2014).

Secondly, it can be deployed as a set of web services with an *application programming interface* (API) that provides the basic functionalities for integrating our ITP method in third-party applications. The web API that can be deployed with Forecat has four GET services that use JSON-formatted[10] data. First, the list of available language pairs has to be obtained by the client. Then, the sentence to translate is submitted to the server and the total number of resulting proposals is returned. After that, the list of suggestions to be offered according to the current typed prefix is requested. If the user selects one of the suggestions, the server has to be notified about this by the client through a fourth service.

Finally, we have recently started to build a *web component* from the existing code of Forecat. Web components comply with a number of standards[11] whose objective is to enable fully encapsulated and reusable components for the web.[12] See section 5.2 for additional details about Forecat's web component.

---

[10]http://www.ecma-international.org/publications/standards/Ecma-404.htm

[11]See http://www.w3.org/TR/components-intro/ for more information.

[12]Web components are called upon to dramatically change how developers build web applications by allowing them to declaratively incorporate independent widgets into their applications with a number of possibilities and advantages not possible with today's established technologies.
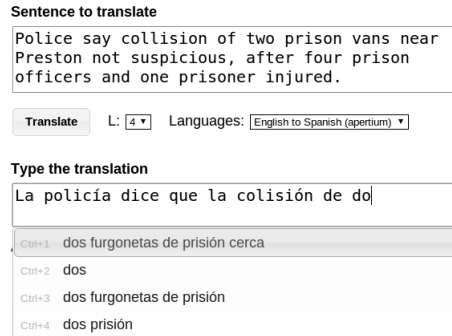
Sentence to translate

Police say collision of two prison vans near
Preston not suspicious, after four prison
officers and one prisoner injured.

Translate   L: 4 ▾   Languages: English to Spanish (apertium) ▾

Type the translation

La policía dice que la colisión de do

Ctrl+1	dos furgonetas de prisión cerca
Ctrl+2	dos
Ctrl+3	dos furgonetas de prisión
Ctrl+4	dos prisión

Figure 1. Screenshot of the interface of our ITP web application showing a translation in progress with some suggestions being offered. The top text box contains the source sentence, whereas users type the translation into the bottom box.

5.1. Programming Languages and Frameworks

Forecat's logic is mostly written in Java with the help of the Google Web Toolkit[13] (GWT), an open-source framework for developing web applications. At the core of the framework is a compiler which translates Java code to JavaScript code which runs flawlessly in current browsers. This allows for a twofold use of the Java code that implements our resource-agnostic ITP method: on the one hand, it can be used locally in Java when performing the automatic evaluation of our approach or when a client calls the corresponding web services; on the other hand, the same code (except for the module responsible of the translation of the segments) can be executed on the browser in JavaScript when human translators interact with the tool either through the web aplication or the web component, thus improving the performance of the tool.

A small part of Forecat, the one dealing with the interface of both the web application or the web component, has been originally written in JavaScript, since we decided not to use GWT for programming the elements of the interface in order to decouple them from the implementation of the method.

5.2. Web Component

We envision Forecat's future as a web component more than as a full web application. This very recent technology allows us to define an encapsulated and interoperable HTML5 custom element (in this case, a translation box) which can easily be imported into any webpage to provide the functionalities of our ITP approach. Forecat code

---

[13]http://www.gwtproject.org/

already includes a working first prototype of an ITP component.[14] For this, it uses Polymer,[15] an open-source Javascript library that simplifies the creation of web components and makes it possible to benefit from them even in those browsers which do not implement the latest version of the underlying standards. The interface of the web component is similar to the bottom box in Figure 1, but the tool works in this case as a standalone widget. The following code shows an example of a simple webpage including a translation box that will offer suggestions as the user translates the sentence *My tailor is healthy* into Spanish, if this pair is available in the component.

```
<!DOCTYPE html>
<html>
<head>
  <script src="bower_components/platform/platform.js"></script>
  <link rel="import" href="elements/translation-box.html">
</head>
<body>
<translation-box id="itp"></translation-box>
<script>
var component = document.querySelector('#itp');
component.addEventListener('languagesReady', function (e) {
  if (contains(e.detail,"en-es")) {
    component.pair= "en-es";
    component.sourceText= "My tailor is healthy.";
  }
});
</script>
</body>
```

The `script` element loads the Polymer library. The next line *imports* our web component which provides the custom element `translation-box` used in the document body. The JavaScript code waits for the `languagesReady` event fired by the component when it is ready to operate and then changes its public attributes `pair` and `sourceText`. The component observes changes in these attributes and then uses their values to obtain the suggestions that will be offered to the translator. The declaration of the component includes a *template* that contains an editable `div` element where the translation will be typed.

---

[14]See the libjs/component directory in the Forecat code.

[15]http://www.polymer-project.org/

5.3. License Choice and Download

Forecat is licensed under version 3 of the GNU Affero General Public License[16] (AGPL). This license is fully compatible with the GNU General Public License (GPL) and equally proposed by the Free Software Foundation, which in fact recommends[17] that "developers consider using the GNU AGPL for any software which will commonly be run over a network". AGPL has been suggested as a means to close a *loophole* in the ordinary GPL which does not force organisations to distribute derivative code when it is only deployed as a web service. The entire code of the application can be downloaded from the Github repository.[18]

6. Conclusions and future work

Resource-agnostic interactive translation prediction (ITP) is a low-cost approach for computer-assisted translation. Forecat, the open-source resource-agnostic ITP tool whose architecture has been discussed in this paper provides a convenient implementation of the ideas behind this approach in the form of a web application, a number of web services, and a web component that can be easily integrated into third-party solutions.

We plan to improve the ranking strategy shown in Section 3 by automatically detecting the part of the input sentence being translated at each moment so that segments that originate in those positions are prioritised. We intend to achieve this by combining word alignment and distortion models. On the one hand, the former will be used to determine the alignments between the last words introduced by the user and the words in the input sentence. On-the-fly, light alignment models have been proposed (Esplà-Gomis et al., 2012) which do not require parallel corpora and are based on the translation of all the possible segments of the sentence with the help of black-box bilingual resources; these models would fit nicely into our ITP method. On the other hand, distortion models, as those proposed by Al-Onaizan and Papineni (2006), will be used to predict which source words will be translated next, partly by using information from the alignment model.

We also plan to explore the impact of simultaneously using different black-box bilingual resources. Different strategies will be evaluated in order to integrate the available resources: combining the findings of the various translation resources into a single suggestion as done by Nirenburg and Frederking (1994) in their multi-engine MT system; using confidence-based measures in order to select the most promising translations as performed by Blatz et al. (2004); or predicting the best candidates for the translation of each particular segment by using only source-language information,

---

[16]http://www.gnu.org/licenses/agpl-3.0.html

[17]http://www.fsf.org/licensing/licenses/

[18]https://github.com/jaspock/forecat/

thus avoiding the need to consult every available resource, as explored by Sánchez-Martınez (2011).

Although there is room for many future improvements, a distance-based ranker, in spite of its simplicity, already provides encouraging results: according to the best results of our automatic experiments, when a maximum of $M = 4$ suggestions are offered and the system selects the longest one that matches the reference translation, around 25–50% keystrokes could be saved depending on the language pair and on the domain of the corpora used to train the SMT system used as bilingual resource.

Acknowledgments

Bibliography

Al-Onaizan, Yaser and Kishore Papineni. Distortion models for statistical machine translation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 529–536. Association for Computational Linguistics, 2006.

Alabau, Vicent, Daniel Ortiz-Martínez, Alberto Sanchis, and Francisco Casacuberta. Multimodal interactive machine translation. In *ICMI-MLMI '10: Proceedings of the 2010 International Conference on Multimodal Interfaces*, 2010.

Alabau, Vicent, Ragnar Bonk, Christian Buck, Michael Carl, Francisco Casacuberta, Mercedes García-Martínez, Jesús González-Rubio, Philipp Koehn, Luis A. Leiva, Bartolomé Mesa-Lao, Daniel Ortiz, Herve Saint-Amand, Germán Sanchis-Trilles, and Chara Tsoukala. CASMACAT: An open source workbench for advanced computer aided translation. *Prague Bull. Math. Linguistics*, 100:101–112, 2013.

Barrachina, Sergio, Oliver Bender, Francisco Casacuberta, Jorge Civera, Elsa Cubel, Shahram Khadivi, Antonio Lagarda, Hermann Ney, Jesús Tomás, Enrique Vidal, and Juan-Miguel Vilar. Statistical approaches to computer-assisted translation. *Computational Linguistics*, 35(1):3–28, 2009.

Blatz, John, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. Confidence estimation for machine translation. In *Proceedings of the 20th International Conference on Computational Linguistics*, COLING '04, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics. doi: 10. 3115/1220355.1220401. URL http://dx.doi.org/10.3115/1220355.1220401.

Esplà-Gomis, Miquel, Felipe Sánchez-Martínez, and Mikel L. Forcada. Using external sources of bilingual information for on-the-fly word alignment. Technical report, Departament de Llenguatges i Sistemes Informàtics, Universitat d'Alacant, 2012.

Forcada, Mikel L, Mireia Ginestí-Rosell, Jacob Nordfalk, Jim O'Regan, Sergio Ortiz-Rojas, Juan Antonio Pérez-Ortiz, Felipe Sánchez-Martínez, Gema Ramírez-Sánchez, and Francis M Tyers. Apertium: a free/open-source platform for rule-based machine translation. *Machine Translation*, 25(2):127–144, 2011.

Foster, George F., Pierre Isabelle, and Pierre Plamondon. Target-text mediated interactive machine translation. *Machine Translation*, 12(1-2):175–194, 1997.

Haddow, Barry and Philipp Koehn. Analysing the effect of out-of-domain data on SMT systems. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, Montreal, Canada, June 2012. Association for Computational Linguistics.

Junczys-Dowmunt, Marcin. Phrasal rank-encoding: Exploiting phrase redundancy and translational relations for phrase table compression. *The Prague Bulletin of Mathematical Linguistics*, 98:63–74, 2012.

Koehn, Philipp. A web-based interactive computer aided translation tool. In *Proceedings of the ACL-IJCNLP 2009 Software Demonstrations*, pages 17–20, 2009.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pages 177–180. Association for Computational Linguistics, 2007.

Langlais, Philippe, Sébastien Sauvé, George Foster, Elliott Macklovitch, and Guy Lapalme. Evaluation of TransType, a computer-aided translation typing system: a comparison of a theoretical-and a user-oriented evaluation procedures. In *Conference on Language Resources and Evaluation (LREC)*, 2000.

Macklovitch, Elliott. TransType2: The last word. In *Proceedings of the 5th International Conference on Languages Resources and Evaluation (LREC 06)*, pages 167–172, 2006.

Nirenburg, Sergei and Robert Frederking. Toward multi-engine machine translation. In *Proceedings of the Workshop on Human Language Technology*, HLT '94, pages 147–151, Stroudsburg, PA, USA, 1994. Association for Computational Linguistics. ISBN 1-55860-357-3. doi: 10.3115/1075812.1075842. URL http://dx.doi.org/10.3115/1075812.1075842.

Ortiz-Martínez, Daniel. *Advances in Fully-Automatic and Interactive Phrase-Based Statistical Machine Translation*. PhD thesis, Universitat Politècnica de València, 2011.

Pérez-Ortiz, Juan Antonio, Daniel Torregrosa, and Mikel Forcada. Black-box integration of heterogeneous bilingual resources into an interactive translation system. In *Proceedings of the EACL 2014 Workshop on Humans and Computer-Assisted Translation*, 2014.

Sánchez-Martınez, Felipe. Choosing the best machine translation system to translate a sentence by using only source-language information. In *Proceedings of the 15th Annual Conference of the European Associtation for Machine Translation*, pages 97–104, 2011.

Address for correspondence:

Juan Antonio Pérez-Ortiz

japerez@dlsi.ua.es

Departament de Llenguatges i Sistemes Informàtics

Universitat d'Alacant

Carretera Sant Vicent del Raspeig s/n

03690 Sant Vicent del Raspeig, Alacant, Spain