Black-box integration of heterogeneous bilingual resources into an interactive translation system

Juan Antonio Pérez-Ortiz japerez@dlsi.ua.es
Daniel Torregrosa dtr5@alu.ua.es
Mikel L. Forcada mlf@dlsi.ua.es

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

Abstract

The objective of interactive translation prediction (ITP) is to assist human translators in the translation of texts 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. In this paper, however, we propose a *resource-agnostic* approach in which the suggestions are obtained from any bilingual resource (a machine translation system, a translation memory, a bilingual dictionary, etc.) that provides target-language equivalents for source-language segments. These bilingual resources are considered to be black boxes and do not need to be adapted to the peculiarities of the ITP system. Our evaluation shows that savings of up to 85% can be theoretically achieved in the number of keystrokes when using our novel approach. Preliminary user trials indicate that these benefits can be partly transferred to real-world computer-assisted translation interfaces.

1 Introduction

Translation technologies are frequently used to assist human translators. Common approaches consider machine translation (MT) (Hutchins and Somers, 1992) or translation memories (Somers, 2003) to be systems that produce a first (and usually incorrect) prototype of the translation which is then edited by the human translator in order to produce a target-language text that is adequate for publishing. In both situations, the suggestion may be considered as a source of inspiration by the human translators, who will assemble the final translation by on some occasions accepting and rearranging parts of the proposal, or on others introducing their own words when an appropriate equivalent is not included or is not found in the suggestion. The whole process may be viewed as a *negotiation* between the wordings that form in the translator's mind and wordings that already appear in the suggestion. In both approaches the suggestion is generated once, usually before starting to manually translate every new sentence.

The approach introduced in this paper, however, follows a different path, which is strongly connected to the field of *interactive translation prediction*[1] (ITP), a research field which explores a kind of computer-assisted translation framework whose aim is to interactively provide users with suggestions at every step during the translation process.[2] Most works in the field of ITP have focused on statistical MT systems as the only source of translations considered to obtain the suggestions, but our study aims to determine how bilingual resources of any kind can be accommodated into an interoperable ITP. To obtain the suggestions, the source-language sentence to be translated is split up into many different (and possibly overlapping) word segments of up to a given length, and a translation for each segment is obtained by using a bilingual resource which is able to deliver one or more target-language equivalents for a particular source-language segment. These equivalents will be the source of the proposals which will be offered to the human translator during the translation process. In principle, the nature of these bilingual resources is not restricted: in

[1] The name *interactive translation prediction* has recently been proposed (Alabau et al., 2013) for this research field, which has historically been referred to as *target-text mediated interactive MT* (Foster et al., 1997) or simply *interactive MT* (Barrachina et al., 2009). Despite the traditional term, we consider the recent one to be more suitable for our approach since it is not exclusively based on MT.

[2] The interaction can be compared to that of word completion mechanisms in input text boxes and word processors.

Workshop on Humans and Computer-assisted Translation, pages 57–65,
Gothenburg, Sweden, 26 April 2014.

this paper we shall explore the use of an MT system, but they may also consist of translation memories, dictionaries, catalogues of bilingual phrases, or a combination of heterogeneous resources. As stated above, MT or translation memories 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, thus saving keystrokes, mouse actions[3] and, possibly, time.

The remainder of the paper is organised as follows. After reviewing the state-of-the-art in ITP in Section 2, we outline the main differences between our proposal and those found in literature in Section 3. Our method for generating translation suggestions from bilingual resources is formally presented in Section 4. We then introduce in Section 5 our experimental set-up and show the results of two evaluations: one that is fully automatic and another consisting of a user trial involving human evaluators. Finally, we discuss the results and propose future lines of research 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). These systems observe the current partial translation already typed by the user and, by exploiting an embedded statistical MT engine, propose one or more completions that are compatible with the sentence prefix entered by the user. Various models were considered for the underlying MT system, including alignment templates, phrase-based models, and stochastic finite-state transducers (Barrachina et al., 2009). The proposals offered may range from one or several words, to a completion of the remainder of the target sentence. An automatic best-scenario evaluation with training and evaluation corpora belonging to the same domain (Barrachina et al., 2009) showed that it might theoretically be possible to use only 20–25% of the keystrokes in comparison with the unassisted translation for English–Spanish translation (both directions) and around 45% for English–French and English–German. The results of the user trials (Macklovitch, 2006) showed gains in productivity (measured in number of words translated per hour) of around 15–20%, but despite this, the human translators were not satisfied with the system, principally because they had to correct the same errors in the proposals over and over again (the models in the underlying statistical MT system remained unchanged during the translation process).

A number of projects continued the research where TransType2 had left off. Caitra (Koehn, 2009) is an ITP tool which uses both the phrase table and the decoder of a statistical MT system to generate suggestions; although individual results vary, translators are generally fastest with post-editing and obtain the highest translation performance when combining post-editing and ITP in the same interface (Koehn and Haddow, 2009). Researchers at the Universitat Politècnica de València have also made significant improvements to the TransType2 system such as online learning techniques with which to adaptively generate better proposals from user feedback (Ortiz-Martínez et al., 2011), phrase-table smoothing to cope with segments in the partially typed translation which cannot be generated with the phrases collected during training (Ortiz-Martínez, 2011), or multimodal interfaces (Alabau et al., 2010). The objective of the CASMACAT project (Alabau et al., 2013), which is under active development, is to develop new types of assistance along all these lines. Finally, commercial translation memory systems have also recently started to introduce ITP as one of their basic features (see, for example, SDL Trados AutoSuggest[4]).

3 Innovative nature of our proposal

Common to most of the approaches discussed above is the fact that the underlying translation engine needs to be a glass-box resource, that is, a resource whose behaviour is modified to meet the ITP system needs. The approaches rely on a statistical MT (Koehn, 2010) system which is adapted to provide the list of n-best completions for the remainder of the sentence, given the current sentence prefix already introduced by the user; in order to meet the resulting time constraints, the decoder of the statistical MT system cannot be executed after each keystroke and techniques to compute the search graph once and then reuse it have been proposed (Bender et al., 2005). However, it

[3] In the case of touch devices, other means of interaction (such as gestures) may exist.

[4] http://www.translationzone.com/

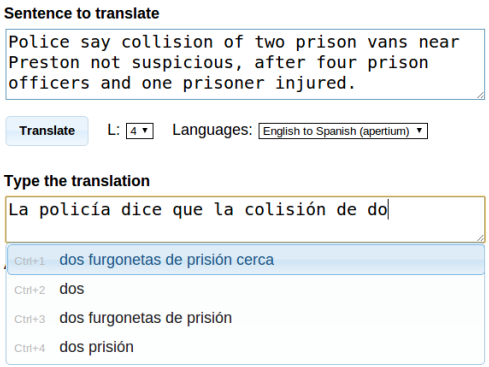

Figure 1: Screenshot of the web interface of our ITP tool showing a translation in progress with some suggestions being offered. The top text box contains the source sentence, whereas users type the translation into the bottom box.

may occur that an ITP system has access to bilingual resources which cannot produce a completion for the rest of the target-language sentence from a given sentence prefix, but are able to supply the translation of a particular source-language segment. This may be owing to either intrinsic reasons inherent to the type of resource being used (for example, a bilingual dictionary can only translate single words or short multi-word units) or extrinsic reasons (for example, an MT system available through a third-party web service cannot be instructed to continue a partial translation).

We propose a black-box treatment of the bilingual resources in contrast to the glass-box approaches found in literature. Unlike them, access to the inner details of the translation system is not necessary; this maintains coupling between the ITP tool and the underlying system to a minimum and provides the opportunity to incorporate additional sources of bilingual information beyond purposely-designed statistical MT systems. Moreover, suggestions are computed once at the start and not after each keystroke, which results in a more effective interaction with the user in execution environments with limited resources.

In this paper, we shall focus on a black-box MT system (Forcada et al., 2011), but we have also begun to explore the integration of other bilingual resources (such as translation memories, dictionaries, catalogues of bilingual phrases, or even a combination of heterogeneous resources). Our system has a web interface similar to that in the projects 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, although this prefix will correspond to the first characters of the word currently being typed and not to the part of the target sentence already entered; 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.

4 Method

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. The resulting segments are then translated by means of a bilingual resource (or combinations thereof). The set of *potential 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. See Table 1 for an example of the set P^S obtained in an English to Spanish translation task when using $L = 3$. We shall represent the i-th suggestion as p_i, its target-language segment as $t(p_i)$ and its corresponding source-language word position as $\sigma(p_i)$. 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.

Let $P^S_C(\hat{w}, j)$ 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 j-th word in the translated sentence T. The elements of $P^S_C(\hat{w}, j)$ are determined by considering only those suggestions in P^S that have the already-typed word prefix as their own prefix:

$$P^S_C(\hat{w}, j) = \{p_i \in P^S : \hat{w} \in \text{Prefix}(t(p_i))\}$$

For example, in the case of the translation of the English sentence in Table 1, if the user types an *M*, the set of compatible suggestions $P^S_C(\text{“M”}, 1)$

Start position	Source segment	Suggestion
1	My	(Mi,1)
1	My tailor	(Mi sastre,1)
1	My tailor is	(Mi sastre es,1)
2	tailor	(sastre,2)
2	tailor is	(sastre es,2)
2	tailor is healthy	(sastre está sano,2)
3	is	(es,3)
3	is healthy	(está sano,3)
4	healthy	(sano,4)

Table 1: Source-language segments and potential suggestions P^S when translating the sentence $S =$ "My tailor is healthy" into Spanish with $L = 3$.

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*. The size of P^S_C is dependent on the value of L, but compatible proposals may also originate from translations of source segments starting at different positions in the input sentence (for example, the set P^S_C after the user types an *s* in the same translation will contain proposals starting with *sastre* and *sano*). More elaborated strategies are consequently necessary to further reduce the number of proposals, since we do not expect users to tolerate lists with more than a few suggestions. In 4.1 we propose the use of a ranking strategy to sort the elements of P^S_C in such a way that it is possible to predict which of them are most suitable to be offered to the user. However, we first elaborate on the issue of compatible suggestions originating from different source positions.

The number of source positions that generate compatible suggestions also depends on the specific word prefix; for example, when users type the letter *d* when translating a long sentence into Spanish, they will probably obtain a significant number of suggestions starting with *de*[5] originating from segments located in different source positions. We measured the number of different positions that provide compatible suggestions when the first characters of the current word are typed during an automatic evaluation of our system (see Section 5); for instance, when translating from English to Spanish, the average is 1.46 after typing *b*, whereas it is 4.73 after typing *d*. Figure 2 shows the average number of different positions for all the letters as users type longer prefixes. Obviously, only suggestions originating from the part of the source sentence currently being translated may be

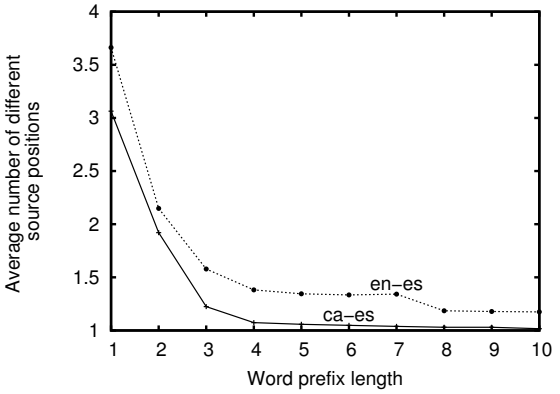

Figure 2: Average number, for all the letters in the alphabet, of different source positions in the source sentence providing compatible suggestions versus length in characters of the typed prefix of the current target word. A system with $L = 4$, $M = \infty$ and no deletion of selected suggestions (see Section 4) was used to obtain the points in this graph. Data is shown for the English–Spanish (*en-es*) and Catalan–Spanish (*es-ca*) corpora used in the automatic experiments (see Section 5).

useful, but this position is difficult to determine unambiguously. The degree of success that can be achieved in this task will be explored in greater depth in future work (see Section 6); a simple approximation is presented in the following section.

4.1 Ranking suggestions

In the absence of a strategy with which to rank the suggestions in $P^S_C(\hat{w}, j)$ which we are currently working on, in this paper we explore a naïve distance-based approach which is based solely on the position j: suggestions p_i whose source position $\sigma(p_i)$ is closer[6] to j are prioritised. For example, in the case of the translation in Table 1, if the user types *Mi s*, suggestions starting with *sastre* will be ranked before those starting with *sano*. This linearity assumption can be seen as a rough attempt to determine the part of the input sentence that is currently being translated; more sophisticated approaches will be considered in future work (see Section 6). However, notice that according to Figure 2, the average number of different source positions of the compatible segments quickly becomes closer to 1 when the length of the word prefix is greater than 2; it is therefore expected that the role played by the distance-based ranker will soon decrease as the user continues typing the

[5]The preposition *de* is notably frequent in Spanish texts.

[6]Ties are broken at random.

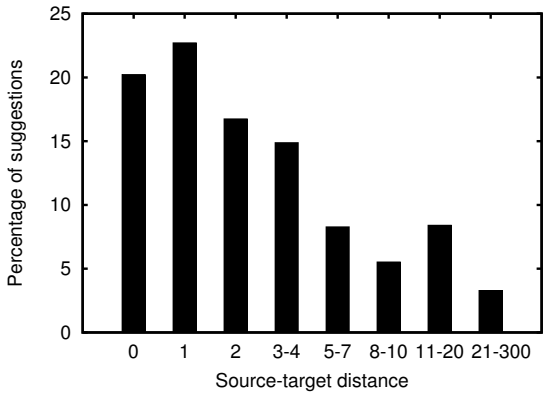

Figure 3: Distribution of the absolute differences (measured in words) between source position of accepted suggestions versus position in the target sentence in which they were selected for the case of Spanish–English translation. $L = 4$, $M = \infty$ and no deletion of selected suggestions (see Section 4) was used to obtain this graph.

current word (although the position of a valid suggestion is far from j, it will probably be the only compatible proposal, and will consequently be selected to be offered).

Translation between closely related languages is often monotonic and most reorderings are local; our distance-based ranking is therefore expected to produce good results for this kind of language pairs. Nevertheless, we cannot in principle expect this ranker to work reasonably well on unrelated languages with different overall grammatical structures (e.g., when translating a language with a verb–subject–object order into another one with a subject–verb–object typology). The graph in Figure 3 represents the distribution of the distances between the source positions of all the accepted suggestions in our automatic Spanish–English evaluation (see Section 5) versus the position in the target sentence of the word for which they were selected. The Pearson correlation coefficient between both positions is very high (0.93), which supports the idea that our naïve distance-based ranking may work reasonably well for the languages used in our experiments.[7]

Let M be the maximum number of suggestions that will eventually be offered to the human translator; the ordered list of *suggestions offered* to the user $P_O^S(\hat{w}, j)$ is made up of a subset of the elements in $P_C^S(\hat{w}, j)$ and restricted so that $|P_O^S(\hat{w}, j)| \leq M$. Note that for the interface to be *friendly*, the value of M should be kept small and, as a result of this, it could easily occur 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, in this paper we propose to restrict the number of proposals originating from a particular source position to two (the longest and the shortest, in this order, which are compatible with the typed word prefix) as long as different compatible suggestions originating from a different position exist. The longest is offered in the hope that it will be correct and will contribute towards saving a lot of keystrokes; however, since the quality of machine translations usually degrades with the length of the input segment (see Figure 4), the shortest is also offered. This must, however, be researched in more depth.

4.2 Deleting dispensable suggestions

Suggestions that have been accepted by the user should not be proposed again. In this work, a selected suggestion p_i will be removed from P^S if no other suggestion p_j with the same target-language text $t(p_i)$ and different source position $\sigma(p_j)$ exists in P^S. In this case, those suggestions obtained from the source position $\sigma(p_i)$ are also removed from P^S. Deleting dispensable suggestions allows other useful suggestions to be selected by the ranker in order to be offered.

5 Experimental setup and results

A fully automatic evaluation and a user trial involving human evaluators were conducted. As previously stated in Section 3, the only bilingual resource considered in this paper is an MT system; in particular, the Spanish to Catalan and English to Spanish rule-based MT systems in the free/open-source platform Apertium[8] (Forcada et al., 2011).

5.1 Evaluation metrics

The performance of our system has been measured by using two metrics: *keystroke ratio* (KSR) and *acceptable suggestion ratio* (ASR). On the one hand, the KSR is the ratio between the number of keystrokes and the length of the translated sen-

[7] Although not shown here, similar results are obtained for the Catalan–Spanish pair.

[8] Revision 44632 of the Apertium repository at http://svn.code.sf.net/p/apertium/svn/trunk/ was used for the engine and linguistic data in these experiments.

tence (Langlais et al., 2000). A lower KSR represents a greater saving in keystrokes. In our experiments, selecting a suggestion has the same cost as pressing one key. On the other hand, the ASR measures the percentage of times that at least one of the suggestions in a non-empty P_O^S is selected. If users frequently receive suggestion lists containing no acceptable proposals, they will stop consulting the list and translate without assistance; it is therefore important to measure the number of times the user is needlessly bothered.

5.2 Automatic evaluation

In order to determine optimal values for the different parameters of our system and to obtain an idea of the best results attainable, a number of automatic tests were conducted. The approach followed is identical to that described by Langlais et al. (2000), in which a parallel corpus with pairs of sentences was used, each pair consisting of a sentence S in the source language and a reference translation T in the target language. In the context of our automatic evaluation, S is used as the input sentence to be translated and T is considered as the target output sentence a user is supposed to have in mind and stick to while *typing*. The longest suggestion in P_O^S which concatenated to the already typed text results in a new prefix of T is always used. If P_O^S contains no suggestions at a particular point, then the system continues *typing* according to T. As the algorithm proceeds in a left-to-right longest-match greedy fashion, there is no guarantee that the best possible results will be obtained, but they will be a good approximation.[9] For example, for $T =$ *Mi coche está averiado*, partial output translation *Mi c*, and $P_O^S(\text{“c”}, 2) = \{$*coche*, *coche es*, *coche está roto*$\}$, our automatic evaluation system will proceed as follows: it will first discard *coche está roto*, because *Mi coche está roto* is not a prefix of T; it will then discard *coche es*, because although *Mi coche es* is a prefix of T, it is not a prefix when a blank is added after it; finally, it will select *coche*, because *Mi coche* followed by a blank is a prefix of T and no longer suggestion that also satisfies these conditions exists.

Two different corpora were used for the automatic evaluation: for English–Spanish (*en-es*), a combination of sentences from all the editions of DGT-TM (Steinberger et al., 2012) released in 2004–2011 (15 250 sentences; 163 196 words in English; 190 448 in Spanish) was used; for Catalan–Spanish (*ca-es*), a collection of news items from El Periódico de Catalunya[10] (15 000 sentences; 307 095 words in Catalan; 294 488 in Spanish) was used.

[9] Note that real users could also decide to select suggestions with small errors and fix them, but neither this nor other behaviours are considered in our automatic evaluation.

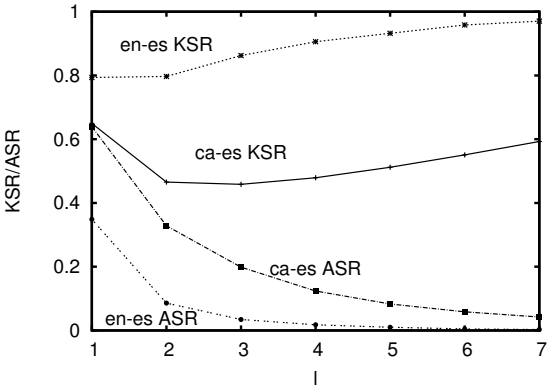

Figure 4: Automatically evaluated KSR versus exact length of the segments l. Longer suggestions are much more useful for Spanish–Catalan (closely related languages) than for English–Spanish: the KSR for $l = 7$ is still a little better than that for $l = 1$ for Catalan–Spanish, but noticeably worse for English–Spanish. ASR quickly degrades as l increases.

5.3 Results of the automatic evaluation

The objective of the automatic evaluation was to estimate the influence of the language pair and the parameters L and M.[11]

Maximum length of segments. We first tested to what extent each different segment length l contributes separately to the KSR. Note that l corresponds in this case to the exact length of the source segments and not to the longest one (as represented by L). $M = \infty$ is used in all the experiments in this section. Figure 4 shows that the KSR becomes worse for greater values of l, which can be explained by the fact that longer machine translations often contain more errors than shorter ones. In the case of Catalan–Spanish, the worst KSR value is for $l = 1$ since adequate suggestions will usually consist of few characters and selecting them will barely contribute to keystroke reduction.

[10] http://www.elperiodico.cat/ca/

[11] 95% confidence intervals of the average values presented in this section were calculated using the Student's t-test. The size of the evaluation corpora signifies that the resulting confidence intervals are so small that they would have been imperceptible on the graphs and have therefore been omitted.

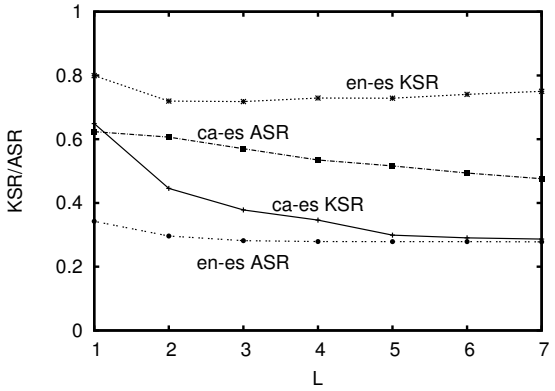

Figure 5: Automatically evaluated KSR/ASR versus maximum length of the segments L. As L increases, the KSR improves, but the ASR is negatively affected.

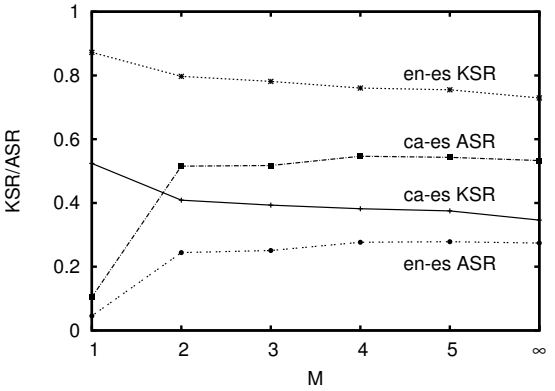

Figure 6: Automatically evaluated KSR/ASR versus maximum number M of suggestions offered. Although the results with $M = 1$ (only one suggestion offered) are considerably worse, for higher values of M they quickly approach the results obtained when no ranker was used and all the compatible suggestions were offered ($M = \infty$).

Combining different segment lengths up to length L provides better values of KSR than using only a fixed value $l = L$ (compare Figures 4 and 5). Figure 5 shows an estimation of the best results our method could attain if all the compatible suggestions in P_C^S were included in P_O^S: values between 0.3 and 0.4 for the Catalan–Spanish KSR and between 0.7 and 0.8 for the English–Spanish KSR. The notable difference may be explained by the fact that Apertium performance is much better (Forcada et al., 2011) for Catalan–Spanish (word error rates of around 15%) than for English–Spanish (word error rates of around 70%).

Maximum number of suggestions offered. We evaluated the influence of the maximum size M of the list of suggestions offered to the user and, hence, the impact of the distance-based ranker. $L = 4$ was used, as this value provides good results for both language pairs (see Figure 5). As expected (see Figure 6), the distance-based ranking strategy works remarkably well (values for KSR and ASR from $M = 4$ are similar to those obtained with $M = \infty$) for closely related languages (Catalan–Spanish), in which translations are usually monotonic and reorderings seldom occur. However, the empirical results also show (see again Figure 6) that it also works well for language pairs (English–Spanish) in which long-distance reorderings exist, at least when compared to the results without ranking ($M = \infty$).

5.4 Human evaluation

A preliminary evaluation of a real use of our system involving 8 human non-professional translators (volunteer computer science students) was also conducted. All the users were Spanish native speakers who understood Catalan, but with no experience with ITP systems. As the results of the automatic evaluation show that the performance of the Apertium English–Spanish MT system negatively affects our ITP system (see Section 5), we decided to focus on the Catalan–Spanish scenario. A set of 10 sentences in Catalan were randomly extracted from the same corpus used in the automatic evaluation. The test was designed to take around 20 minutes. The evaluators were allowed to practise with a couple of sentences before starting the trial. After completing the test, they were surveyed about the usefulness of the system. Our ITP system was used with $L = M = 4$.

5.5 Results of the human evaluation

The users were divided into two groups: users 1–4 translated sentences 1–5 assisted by our ITP tool and sentences 6–10 with no assistance, while users 5–8 translated sentences 1–5 with no assistance and sentences 6–10 assisted by the tool. The KSR and translation times for each user are shown in Table 2. This table also includes KSR′, which is the value of KSR obtained by running our automatic evaluator (see Section 5.2) using the sentences entered by each user as the reference translations T; this can be considered as an approximation to the best result achievable with the ITP tool. All users attained KSRs that were notice-

User	Sentences 1–5			Sentences 6–10		
	KSR	Time	KSR′	KSR	Time	KSR′
#1	**0.49**	**136**	**0.22**	1.11	137	0.23
#2	**0.64**	**144**	**0.15**	1.21	86	0.22
#3	**0.63**	**209**	**0.22**	1.09	112	0.21
#4	**0.37**	**189**	**0.22**	1.22	199	0.18
#5	1.10	145	0.28	**0.37**	**102**	**0.15**
#6	1.24	150	0.27	**0.51**	**154**	**0.17**
#7	1.15	178	0.30	**0.64**	**147**	**0.17**
#8	1.18	118	0.39	**0.58**	**93**	**0.15**

Table 2: KSR, translation times (seconds) and KSR′ (see main text) for each of the users in the evaluation. Values in bold correspond to the translations with assistance from our ITP system.

ably lower than 1 for the assisted translations and slightly higher than 1 when translating without the ITP system; the former, however, are often worse than the KSR values obtained in the automatic evaluation which are around 0.4 for $L = M = 4$ (see Figure 6). Moreover, the values for KSR′ show that even better values for KSR could theoretically be attained for these sentences; note, however, that the reference translations in this case were precisely generated by accepting suggestions generated by Apertium.

The users were surveyed to evaluate the following statements in the range from 1 (complete disagreement) to 5 (complete agreement): *the interface is easy to use*; *I would use a tool like this in future translations*; *I have found the suggestions useful*; and *the tool has allowed me to translate faster*. The median of the responses to the first two questions was 5, whereas the median for the two last questions was 4.5. It was evident that the evaluators perceived that the ITP system had helped them to translate faster, although the time values in Table 2 seem to suggest the opposite. Finally, note that this was a small-scale human evaluation and that sounder results will have to be collected under different conditions by increasing the number of users, sentences and languages in the test.

6 Discussion and future work

The automatic evaluation of our ITP system has provided an estimation of its potential for human translators. Note, however, that this evaluation strategy is based on a greedy algorithm which may not adequately reproduce the way in which a human translator might usually perform the task. 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, 25–65% keystrokes could be saved depending on the language pair. Moreover, 30–55% of the times that a list of suggestions is offered, at least one of the suggestions matches the target sentence.

Our preliminary human tests can be used to discern how well our system could perform, but a more extensive evaluation is needed to explore the influence of parameters, different kinds of users, heterogeneous bilingual resources, new language pairs, particular domains, different interfaces, etc. in greater depth. A comparison with similar tools in literature will also be carried out.

We plan to improve the ranking strategy shown in Section 4.1 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. The former will be used to determine the alignments between the last words introduced by the user and the words in the input sentence;[12] the latter will predict which source words will be translated next, partly by using information from the alignment model.

The ITP system presented in this paper is implemented in Java, except for the web interface, which is written in HTML and JavaScript. The Java code, however, has been designed in such a way that it can be compiled into JavaScript with the help of the Google Web Toolkit framework;[13] and the same code can therefore be executed either on the browser in JavaScript when human translators interact with the tool, or locally in Java when performing the automatic evaluation. The entire code of the application is available[14] under a free software license (GNU Affero General Public License, version 3); this ensures the reproducibility of the experiments and allows our ITP system to be integrated into professional translation tools.

Acknowledgments. This work has been partly funded by the Spanish Ministerio de Economía y Competitividad through project TIN 2012-32615.

[12] On-the-fly, light alignment models have been proposed 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 (Esplà-Gomis et al., 2012); these models would fit nicely into our ITP method.

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

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

References

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

Vicent Alabau, 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. 2013. CASMACAT: An open source workbench for advanced computer aided translation. *Prague Bull. Math. Linguistics*, 100:101–112.

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

Oliver Bender, David Vilar, Richard Zens, and Hermann Ney. 2005. Comparison of generation strategies for interactive machine translation. In *Proceedings of EAMT 2005 (10th Annual Conference of the European Association for Machine Translation*, pages 30–40.

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

Mikel L Forcada, 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. 2011. Apertium: a free/open-source platform for rule-based machine translation. *Machine Translation*, 25(2):127–144.

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

W. John Hutchins and Harold L. Somers. 1992. *An introduction to machine translation*. Academic Press.

Philipp Koehn and Barry Haddow. 2009. Interactive assistance to human translators using statistical machine translation methods. *MT Summit XII*.

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

Philipp Koehn. 2010. *Statistical Machine Translation*. Cambridge University Press.

Philippe Langlais, Sébastien Sauvé, George Foster, Elliott Macklovitch, and Guy Lapalme. 2000. 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)*, page 8.

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

Daniel Ortiz-Martínez, Luis A. Leiva, Vicent Alabau, Ismael García-Varea, and Francisco Casacuberta. 2011. An interactive machine translation system with online learning. In *Proceedings of the ACL-HLT 2011 System Demonstrations*, pages 68–73.

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

Harold L. Somers. 2003. *Computers and Translation: A Translator's Guide*. Benjamins translation library. John Benjamins Publishing Company.

Ralf Steinberger, Andreas Eisele, Szymon Klocek, Spyridon Pilos, and Patrick Schlüter. 2012. DGT-TM: a freely available Translation Memory in 22 languages. In *Language Resources and Evaluation Conference*, pages 454–459.