Translation Assistance by Translation of L1 Fragments in an L2 Context

Maarten van Gompel & Antal van den Bosch
Centre for Language Studies
Radboud University Nijmegen
proycon@anaproy.nl

Abstract

In this paper we present new research in translation assistance. We describe a system capable of translating native language (L1) fragments to foreign language (L2) fragments in an L2 context. Practical applications of this research can be framed in the context of second language learning. The type of translation assistance system under investigation here encourages language learners to write in their target language while allowing them to fall back to their native language in case the correct word or expression is not known. These code switches are subsequently translated to L2 given the L2 context. We study the feasibility of exploiting cross-lingual context to obtain high-quality translation suggestions that improve over statistical language modelling and word-sense disambiguation baselines. A classification-based approach is presented that is indeed found to improve significantly over these baselines by making use of a contextual window spanning a small number of neighbouring words.

1 Introduction

Whereas machine translation generally concerns the translation of whole sentences or texts from one language to the other, this study focusses on the translation of native language (henceforth L1) words and phrases, i.e. smaller fragments, in a foreign language (L2) context. Despite the major efforts and improvements, automatic translation does not yet rival human-level quality. Vexing issues are morphology, word-order change and long-distance dependencies. Although there is a morpho-syntactic component in this research, our scope is more constrained; its focus is on the faithful preservation of meaning from L1 to L2, akin to the role of the translation model in Statistical Machine Translation (SMT).

The cross-lingual context in our research question may at first seem artificial, but its design explicitly aims at applications related to computer-aided language learning (Laghos and Panayiotis, 2005; Levy, 1997) and computer-aided translation (Barrachina et al., 2009). Currently, language learners need to refer to a bilingual dictionary when in doubt about a translation of a word or phrase. Yet, this problem arises in a context, not in isolation; the learner may have already translated successfully a part of the text into L2 leading up to the problematic word or phrase. Dictionaries are not the best source to look up context; they may contain example usages, but remain biased towards single words or short expressions.

The proposed application allows code switching and produces context-sensitive suggestions as writing progresses. In this research we test the feasibility of the foundation of this idea.The following examples serve to illustrate the idea and demonstrate what output the proposed translation assistance system would ideally produce. The parts in bold correspond to respectively the inserted fragment and the system translation.

- Input (L1=English,L2=Spanish): *"Hoy vamos a **the swimming pool**."*
  Desired output: *"Hoy vamos a **la piscina**."*
- Input (L1-English, L2=German): *"Das wetter ist wirklich **abominable**."*
  Desired output: *"Das wetter ist wirklich **ekelhaft**."*
- Input (L1=French,L2=English): *"I **rentre à la maison** because I am tired."*
  Desired output: *"I **return home** because I am tired."*
- Input (L1=Dutch, L2=English): *"Workers are facing a massive **aanval op** their employ-*

Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, pages 871–880,
Baltimore, Maryland, USA, June 23-25 2014.

ment and social rights."
Desired output: *"Workers are facing a massive **attack on** their employment and social rights."*

The main research question in this research is how to disambiguate an L1 word or phrase to its L2 translation based on an L2 context, and whether such cross-lingual contextual approaches provide added value compared to baseline models that are not context informed or compared to standard language models.

2 Data preparation

Preparing the data to build training and test data for our intended translation assistance system is not trivial, as the type of interactive translation assistant we aim to develop does not exist yet. We need to generate training and test data that realistically emulates the task. We start with a parallel corpus that is tokenised for both L1 and L2. No further linguistic processing such as part-of-speech tagging or lemmatisation takes place in our experiments; adding this remains open for future research.

The parallel corpus is randomly sampled into two large and equally-sized parts. One is the basis for the training set, and the other is the basis for the test set. The reason for such a large test split shall become apparent soon.

From each of the splits (S), a phrase-translation table is constructed automatically in an unsupervised fashion. This is done using the scripts provided by the Statistical Machine Translation system Moses (Koehn et al., 2007). It invokes GIZA++ (Och and Ney, 2000) to establish statistical word alignments based on the IBM Models and subsequently extracts phrases using the `grow-diag-final` algorithm (Och and Ney, 2003). The result, independent for each set, will be a phrase-translation table (T) that maps phrases in L1 to L2. For each phrase-pair (f_s, f_t) this phrase-translation table holds the computed translation probabilities $P(f_s|f_t)$ and $P(f_t|f_s)$.

Given these phrase-translation tables, we can now extract both training data and test data using the algorithm in Figure 1. In our discourse, the source language (s) corresponds to L1, the fallback language used for by the end-user for inserting fragments, whilst the target language (t) is L2.

Step 4 is effectively a filter: two thresholds can be configured to discard weak alignments,

1. using phrase-translation table T and parallel corpus split S
2. **for** each aligned sentence pair $(sentence_s \in S_s, sentence_t \in S_t)$ in the parallel corpus split (S_s,S_t):
3. **for** each fragment $(f_s \in sentence_s, f_t \in sentence_t)$ where $(f_s, f_t) \in T$:
4. **if** $P(f_s|f_t) \cdot P(f_t|f_s) \geq \lambda_1$ **and** $P(f_s|f_t) \cdot P(f_t|f_s) \geq \lambda_2 \cdot P(f_s|f_{strongest\_t}) \cdot P(f_{strongest\_t}|f_s)$:
5. Output a pair $(sentence'_t, sentence_t)$ where $sentence'_t$ is a copy of t but with fragment f_t substituted by f_s, i.e. the introduction of an L1 word or phrase in an L2 sentence.

Figure 1: Algorithm for extracting training and test data on the basis of a phrase-translation table (T) and subset/split from a parallel corpus (S). The indentation indicates the nesting.

i.e. those with low probabilities, from the phrase-translation table so that only strong couplings make it into the generated set. The parameter λ_1 adds a constraint based on the product of the two conditional probabilities ($P(f_t|f_s) \cdot P(f_s|f_t)$), and sets a threshold that has to be surpassed. A second parameter λ_2 further limits the considered phrase pairs (f_s, f_t) to have the product of their conditional probabilities not not deviate more than a fraction λ_2 from the joint probability for the strongest possible pairing for f_s, the source fragment. $f_{strongest\_t}$ in Figure 1 corresponds to the best scoring translation for a given source fragment f_s. This metric thus effectively prunes weaker alternative translations in the phrase-translation table from being considered if there is a much stronger candidate. Nevertheless, it has to be noted that even with λ_1 and λ_2, the test set will include a certain amount of errors. This is due to the nature of the unsupervised method with which the phrase-translation table is constructed. For our purposes however, the test set suffices to test our hypothesis.

In our experiments, we choose fixed values for these parameters, by manual inspection and judgement of the output. The λ_1 parameter was set to 0.01 and λ_2 to 0.8. Whilst other thresholds may possibly produce cleaner sets, this is hard to evaluate as finding optimal values causes a prohibitive increase in complexity of the search space, and again this is not necessary to test our hypothesis.

The output of the algorithm in Figure 1 is a modified set of sentence pairs $(sentence'_t, sentence_t)$, in which the same sentence pair may be used multiple times with different L1 substitutions for different fragments. The final test set is created by randomly sampling the desired number of test instances.

Note that the training set and test set are constructed on their own respective and independently generated phrase-translation tables. This ensures complete independence of training and test data. Generating test data using the same phrase-translation table as the training data would introduce a bias. The fact that a phrase-translation table needs to be constructed for the test data is also the reason that the parallel corpus split from which the test data is derived has to be large enough, ensuring better quality.

We concede that our current way of testing is a mere approximation of the real-world scenario. An ideal test corpus would consist of L2 sentences with L1 fallback as crafted by L2 language learners with an L1 background. However, such corpora do not exist as yet. Nevertheless, we hope to show that our automated way of test set generation is sufficient to test the feasibility of our core hypothesis that L1 fragments can be translated to L2 using L2 context information.

3 System

We develop a classifier-based system composed of so-called "classifier experts". Numerous classifiers are trained and each is an expert in translating a single word or phrase. In other words, for each word type or phrase type that occurs as a fragment in the training set, and which does not map to just a single translation, a classifier is trained. The classifier maps the L1 word or phrase in its L2 context to its L2 translation. Words or phrases that always map to a single translation are stored in a simple mapping table, as a classifier would have no added value in such cases. The classifiers use the IB1 algorithm (Aha et al., 1991) as implemented in TiMBL (Daelemans et al., 2009).[1] IB1 implements k-nearest neighbour classification. The choice for this algorithm is motivated by the fact that it handles multiple classes with ease, but first and foremost because it has been successfully employed for word sense disambiguation in other studies (Hoste et al., 2002; Decadt et al., 2004), in particular in cross-lingual word sense disambiguation, a task closely resembling our current task (van Gompel and van den Bosch, 2013). It has also been used in machine translation studies in which local source context is used to classify source phrases into target phrases, rather than looking them up in a phrase table (Stroppa et al., 2007; Haque et al., 2011). The idea of local phrase selection with a discriminative machine learning classifier using additional local (source-language) context was introduced in parallel to Stroppa *et al.* (2007) by Carpuat and Wu (2007) and Giménez and Márquez (2007); cf. Haque *et al.* (2011) for an overview of more recent methods.

The feature vector for the classifiers represents a local context of neighbouring words, and optionally also global context keywords in a binary-valued bag-of-words configuration. The local context consists of an X number of L2 words to the left of the L1 fragment, and Y words to the right.

When presented with test data, in which the L1 fragment is explicitly marked, we first check whether there is ambiguity for this L1 fragment and if a direct translation is available in our simple mapping table. If so, we are done quickly and need not rely on context information. If not, we check for the presence of a classifier expert for the offered L1 fragment; only then we can proceed by extracting the desired number of L2 local context words to the immediate left and right of this fragment and adding those to the feature vector. The classifier will return a probability distribution of the most likely translations given the context and we can replace the L1 fragment with the highest scoring L2 translation and present it back to the user.

In addition to local context features, we also experimented with global context features. These are a set of L2 contextual keywords for each L1 word/phrase and its L2 translation occurring in the same sentence, not necessarily in the immediate neighbourhood of the L1 word/phrase. The keywords are selected to be indicative for a specific

[1] `http://ilk.uvt.nl/timbl`

translation. We used the method of extraction by Ng and Lee (1996) and encoded all keywords in a binary bag of words model. The experiments however showed that inclusion of such keywords did not make any noticeable impact on any of the results, so we restrict ourselves to mentioning this negative result.

Our full system, including the scripts for data preparation, training, and evaluation, is implemented in Python and freely available as open-source from `http://github.com/proycon/colibrita/` . Version tag `v0.2.1` is representative for the version used in this research.

3.1 Language Model

We also implement a statistical language model as an optional component of our classifier-based system and also as a baseline to compare our system to. The language model is a trigram-based backoff language model with Kneser-Ney smoothing, computed using SRILM (Stolcke, 2002) and trained on the same training data as the translation model. No additional external data was brought in, to keep the comparison fair.

For any given hypothesis H, results from the L1 to L2 classifier are combined with results from the L2 language model. We do so by normalising the class probability from the classifier ($score_T(H)$), which is our translation model, and the language model ($score_{lm}(H)$), in such a way that the highest classifier score for the alternatives under consideration is always 1.0, and the highest language model score of the sentence is always 1.0. Take $score_T(H)$ and $score_{lm}(H)$ to be log probabilities, the search for the best (most probable) translation hypothesis $\hat{H}$ can then be expressed as:

$$\hat{H} = \arg\max_H (score_T(H) + score_{lm}(H)) \quad (1)$$

If desired, the search can be parametrised with variables λ_3 and λ_4, representing the weights we want to attach to the classifier-based translation model and the language model, respectively. In the current study we simply left both weights set to one, thereby assigning equal importance to translation model and language model.

4 Evaluation

Several automated metrics exist for the evaluation of L2 system output against the L2 reference output in the test set. We first measure absolute accuracy by simply counting all output fragments that exactly match the reference fragments, as a fraction of the total amount of fragments. This measure may be too strict, so we add a more flexible *word accuracy* measure which takes into account partial matches at the word level. If output o is a subset of reference r then a score of $\frac{|o|}{|r|}$ is assigned for that sentence pair. If instead, r is a subset of o, then a score of $\frac{|r|}{|o|}$ will be assigned. A perfect match will result in a score of 1 whereas a complete lack of overlap will be scored 0. The word accuracy for the entire set is then computed by taking the sum of the word accuracies per sentence pair, divided by the total number of sentence pairs.

We also compute a recall metric that measures the number of fragments that the system provided a translation for as a fraction of the total number of fragments in the input, regardless of whether the fragment is translated correctly or not. The system may skip fragments for which it can find no solution at all.

In addition to these, the system's output can be compared against the L2 reference translation(s) using established Machine Translation evaluation metrics. We report on BLEU, NIST, METEOR, and word error rate metrics WER and PER. These scores should generally be much better than the typical MT system performances as only local changes are made to otherwise "perfect" L2 sentences.

5 Baselines

A context-insensitive yet informed baseline was constructed to assess the impact of L2 context information in translating L1 fragments. The baseline selects the most probable L1 fragment per L2 fragment according to the phrase-translation table. This baseline, henceforth referred to as the 'most likely fragment' baseline (MLF) is analogous to the 'most frequent sense'-baseline common in evaluating WSD systems.

A second baseline was constructed by weighing the probabilities from the translation table directly with the L2 language model described earlier. It adds a LM component to the MLF baseline. This LM baseline allows the comparison of classification through L1 fragments in an L2 context, with a more traditional L2 context modelling (i.e. target language modelling) which is also cus-

tomary in MT decoders. Computing this baseline is done in the same fashion as previously illustrated in Equation 1, where $score_T$ then represents the normalised $p(t|s)$ score from the phrase-translation table rather than the class probability from the classifier.

6 Experiments & Results

The data for our experiments were drawn from the Europarl parallel corpus (Koehn, 2005) from which we extracted two sets of $200,000$ sentence pairs each for several language pairs. These were used to form the training and test sets. The final test sets are a randomly sampled $5,000$ sentence pairs from the $200,000$-sentence test split for each language pair.

All input data for the experiments in this section are publicly available[2].

Let us first zoom in to convey a sense of scale on a specific language pair. The actual Europarl training set we generate for English (L1) to Spanish (L2), i.e. English fallback in a Spanish context, consists of $5,608,015$ sentence pairs. This number is much larger than the $200,000$ we mentioned before because single sentence pairs may be reused multiple times with different marked fragments. From this training set of sentence pairs over $100,000$ classifier experts are derived. The eleven largest classifiers are shown in Table 1, along with the number of training instances per classifier. The full table would reveal a Zipfian distribution.

Fragment	Training instances	Translations
the	256,772	la, el, los, las
of	139,273	de, del
and	128,074	y, de, e
to	66,565	a, para, que, de
a	54,306	un, una
is	40,511	es, está, se
for	34,054	para, de, por
this	29,691	este, esta, esto
European	26,543	Europea, Europeo Europeas, Europeos
on	23,147	sobre, en
of the	22,361	de la, de los

Table 1: The top eleven classifier experts for English to Spanish. The eleventh entry is included as an example of a common phrasal fragment

Among the classifier experts are only words and phrases that are ambiguous and may thus map to multiple translations. This implies that such words and phrases must have occurred at least twice in the corpus, though this threshold is made configurable and could have been set higher to limit the number of classifiers. The remaining $246,380$ unambiguous mappings are stored in a separate mapping table.

[2]Download and unpack `http://lst.science.ru.nl/~proycon/colibrita-acl2014-data.zip`

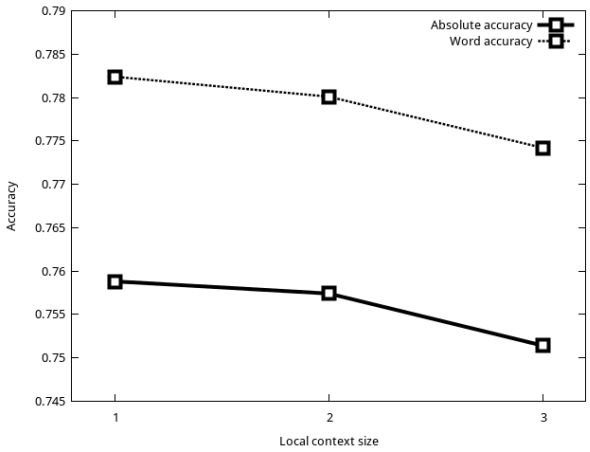

Figure 2: Accuracy for different local context sizes, Europarl English to Spanish

For the classifier-based system, we tested various different feature vector configurations. The first experiment, of which the results are shown in Figure 2, sets a fixed and symmetric local context size across all classifiers, and tests three context widths. Here we observe that a context width of one yields the best results. The BLEU scores, not included in the figure but shown in Table 2, show a similar trend. This trend holds for all the MT metrics.

Table 2 shows the results for English to Spanish in more detail and adds a comparison with the two baseline systems. The various `lXrY` configurations use the same feature vector setup for all classifier experts. Here X indicates the left context size and Y the right context size. The `auto` configuration does not uniformly apply the same feature vector setup to all classifier experts but instead seeks to find the optimal setup per classifier expert. This shall be further discussed in Section 6.1.

As expected, the LM baseline substantially outperforms the context-insensitive MLF baseline. Second, our classifier approach attains a substantially higher accuracy than the LM baseline. Third, we observe that adding the language model to our classifier leads to another significant gain

Configuration	Accuracy	Word Accuracy	BLEU	METEOR	NIST	WER	PER
MLF baseline	0.6164	0.6662	0.972	0.9705	17.0784	1.4465	1.4209
LM baseline	0.7158	0.7434	0.9785	0.9739	17.1573	1.1735	1.1574
l1r1	0.7588	0.7824	0.9801	0.9747	17.1550	1.1625	1.1444
l2r2	0.7574	0.7801	0.9800	0.9746	17.1550	1.1750	1.1569
l3r3	0.7514	0.7742	0.9796	0.9744	17.1445	1.1946	1.1780
l1r1+LM	**0.7810**	**0.7973**	**0.9816**	**0.9754**	**17.1685**	**1.0946**	**1.077**
auto	0.7626	0.7850	0.9803	0.9748	17.1544	1.1594	1.1424
auto+LM	0.7796	0.7966	0.9815	0.9754	17.1664	1.1021	1.0845
l1r0	0.6924	0.7223	0.9757	0.9723	17.1087	1.3415	1.3249
l2r0	0.6960	0.7245	0.9759	0.9724	17.1091	1.3364	1.3193
l2r1	0.7624	0.7849	0.9803	0.9748	17.1558	1.1554	1.1378

Table 2: Europarl results for English to Spanish (i.e English fallback in Spanish context). Recall = 0.9422

(configuration `l1r1+LM` in the results in Table 2). It appears that the classifier approach and the L2 language model are able to complement each other.

Statistical significance on the BLEU scores was tested using pairwise bootstrap sampling (Koehn, 2004). All significance tests were performed with $5,000$ iterations. We compared the outcomes of several key configurations. We first tested `l1r1` against both baselines; both differences are significant at $p < 0.01$ for both. The same significance level was found when comparing `l1r1+LM` against `l1r1`, `auto+LM` against `auto`, as well as the LM baseline against the MLF baseline. Automatic feature selection `auto` was found to perform statistically better than `l1r1`, but only at $p < 0.05$. Conclusions with regard to context width may have to be tempered somewhat, as the performance of the `l1r1` configuration was found to not be significantly better than that of the `l2r2` configuration. However, `l1r1` performs significantly better than `l3r3` at $p < 0.01$, and `l2r2` performs significantly better than `l3r3` at $p < 0.01$.

In Table 3 we present some illustrative examples from the English→Spanish Europarl data. We show the difference between the most-likely-fragment baseline and our system.

Likewise, Table 4 exemplifies small fragments from the `l1r1` configuration compared to the same configuration enriched with a language model. We observe in this data that the language model often has the added power to choose a correct translation that is not the first prediction of the classifier, but one of the weaker alternatives that nevertheless fits better. Though the classifier generally works best in the `l1r1` configuration, i.e. with context size one, the trigram-based language model allows further left-context information to be incorporated that influences the weights of the classifier output, successfully forcing the system to select alternatives. This combination of a classifier with context size one and trigram-based language model proves to be most effective and reaches the best results so far. We have not conducted experiments with language models of other orders.

6.1 Context optimisation

It has been argued that classifier experts in a word sense disambiguation ensemble should be individually optimised (Decadt et al., 2004; van Gompel and van den Bosch, 2013). The latter study on cross-lingual WSD finds a positive impact when conducting feature selection per classifier. This intuitively makes sense; a context of one may seem to be better than any other when uniformly applied to all classifier experts, but it may well be that certain classifiers benefit from different feature selections. We therefore proceed with this line of investigation as well.

Automatic configuration selection was done by performing leave-one-out testing (for small number of instances) or 10-fold-cross validation (for larger number of instances, $n \geq 20$) on the training data per classifier expert. Various configurations were tested. Per classifier expert, the best scoring configuration was selected, referred to as the `auto` configuration in Table 2. The `auto` configuration improves results over the uniformly

Input: Mientras no haya prueba en contrario , la financiación de partidos políticos **European** sólo se justifica , incluso después del tratado de Niza , desde el momento en que concurra a la expresión del sufragio universal , que es la única definición aceptable de un partido político .
MLF baseline: Mientras no haya prueba en contrario , la financiación de partidos políticos **Europea** sólo se justifica , incluso después del tratado de Niza , desde el momento en que concurra a la expresión del sufragio universal , que es la única definición aceptable de un partido político .
l1r1: Mientras no haya prueba en contrario , la financiación de partidos políticos **europeos** sólo se justifica , incluso después del tratado de Niza , desde el momento en que concurra a la expresión del sufragio universal , que es la única definición aceptable de un partido político .

Input: Esta Directiva es nuestra oportunidad **to** marcar una verdadera diferencia , reduciendo la trágica pérdida de vidas en nuestras carreteras .
MLF baseline: Esta Directiva es nuestra oportunidad **a** marcar una verdadera diferencia , reduciendo la trágica pérdida de vidas en nuestras carreteras .
l1r1: Esta Directiva es nuestra oportunidad **para** marcar una verdadera diferencia , reduciendo la trágica pérdida de vidas en nuestras carreteras .

Input: Es la **last** vez que me dirijo a esta Cámara .
MLF baseline: Es la **pasado** vez que me dirijo a esta Cámara .
l1r1: Es la **última** vez que me dirijo a esta Cámara .

Input: Pero el enfoque actual de la Comisión no puede conducir a una buena política ya que es tributario del funcionamiento del mercado y de las normas establecidas por la OMC , el FMI y el Banco Mundial , normas que siguen siendo desfavorables para los **developing countries** .
MLF baseline: Pero el enfoque actual de la Comisión no puede conducir a una buena política ya que es tributario del funcionamiento del mercado y de las normas establecidas por la OMC , el FMI y el Banco Mundial , normas que siguen siendo desfavorables para los **los países en desarrollo** .
l1r1: Pero el enfoque actual de la Comisión no puede conducir a una buena política ya que es tributario del funcionamiento del mercado y de las normas establecidas por la OMC , el FMI y el Banco Mundial , normas que siguen siendo desfavorables para los **países en desarrollo** .

Table 3: Some illustrative examples of MLF-baseline output versus system output, in which system output matches the correct human reference output. The actual fragments concerned are highlighted in bold. The first example shows our system correcting for number agreement, the second a correction in selecting the right preposition, and the third shows that the English word *last* can be translated in different ways, only one of which is correct in this context. The last example shows a phrasal translation, in which the determiner was duplicated in the baseline

applied feature selection. However, if we enable the language model as we do in the `auto+LM` configuration we do not notice an improvement over `l1r1+LM`, surprisingly. We suspect the lack of impact here can be explained by the trigram-based Language Model having less added value when the (left) context size of the classifier is two or three; they are now less complementary.

Table 5 lists what context sizes have been chosen in the automatic feature selection. A context size of one prevails in the vast majority of cases, which is not surprising considering the good results we have already seen with this configuration.

In this study we did not yet conduct optimisation of the classifier parameters. We used the IB1 algorithm with $k = 1$ and the default values of the TiMBL implementation. In earlier work van Gompel and van den Bosch (2013), we reported a decrease in performance due to overfitting when this is done, so we do not expect it to make a positive impact. The second reason for omitting this is more practical in nature; to do this in combination with feature selection would add substantial search complexity, making experiments far more time consuming, even prohibitively so.

66.5%	l1r1
19.9%	l2r2
7.7%	l3r3
3.5%	l4r4
2.4%	l5r5

Table 5: Frequency of automatically selected configurations on English to Spanish Europarl dataset

The bottom lines in Table 2 represent results when all right-context is omitted, emulating a real-time prediction when no right context is available yet. This has a substantial negative impact on re-

Input: Sin ese tipo de protección la gente no aprovechará la oportunidad **to** vivir , viajar y trabajar donde les parezca en la Unión Europea .
l1r1: Sin ese tipo de protección la gente no aprovechará la oportunidad **para** vivir , viajar y trabajar donde les parezca en la Unión Europea .
l1r1+LM: Sin ese tipo de protección la gente no aprovechará la oportunidad **de** vivir , viajar y trabajar donde les parezca en la Unión Europea .

Input: La Comisión también está acometiendo medidas en el ámbito social y **educational** con vistas a mejorar la situación de los niños .
l1r1: La Comisión también está acometiendo medidas en el ámbito social y **educativas** con vistas a mejorar la situación de los niños .
l1r1+LM: La Comisión también está acometiendo medidas en el ámbito social y **educativo** con vistas a mejorar la situación de los niños .

Table 4: Some examples of l1r1 versus the same configuration enriched with a language model.

sults. We experimented with several asymmetric configurations and found that taking two words to the left and one to the right yields even better results than symmetric configurations for this data set. This result is in line with the positive effect of adding the LM to the `l1r1`.

In order to draw accurate conclusions, experiments on a single data set and language pair are not sufficient. We therefore conducted a number of experiments with other language pairs, and present the abridged results in Table 6.

There are some noticeable discrepancies for some experiments in Table 6 when compared to our earlier results in Table 2. We see that the language model baseline for English→French shows the same substantial improvement over the baseline as our English→Spanish results. The same holds for the Chinese→English experiment. However, for English→Dutch and English→Chinese we find that the LM baseline actually performs slightly worse than baseline. Nevertheless, in all these cases, the positive effect of including a Language Model to our classifier-based system again shows. Also, we note that in all cases our system performs better than the two baselines.

Another discrepancy is found in the BLEU scores of the English→Chinese experiments, where we measure an unexpected drop in BLEU score under baseline. However, all other scores do show the expected improvement. The error rate metrics show improvement as well. We therefore attach low importance to this deviation in BLEU here.

In all of the aforementioned experiments, the system produced a single solution for each of the fragments, the one it deemed best, or no solution at all if it could not find any. Alternative evaluation metrics could allow the system to output multiple alternatives. Omission of a solution by definition causes a decrease in recall. In all of our experiments recall is high (well above 90%), mostly because train and test data lie in the same domain and have been generated in the same fashion, lower recall is expected with more real-world data.

7 Discussion and conclusion

In this study we have shown the feasibility of a classifier-based translation assistance system in which L1 fragments are translated in an L2 context, in which the classifier experts are built individually per word or phrase. We have shown that such a translation assistance system scores both above a context-insensitive baseline, as well as an L2 language model baseline.

Furthermore, we found that combining this cross-language context-sensitive technique with an L2 language model boosts results further.

The presence of a one-word right-hand side context proves crucial for good results, which has implications for practical translation assistance application that translate as soon as the user finishes an L1 fragment. Revisiting the translation when right context becomes available would be advisable.

We tested various configurations and conclude that small context sizes work better than larger ones. Automated configuration selection had positive results, yet the system with context size one and an L2 language model component often produces the best results. In static configurations, the failure of a wider context window to be more suc-

Dataset	L1	L2	Configuration	Accuracy	Word Accuracy	BLEU
europarl200k	en	nl	baseline	0.7026	0.7283	0.9771
europarl200k	en	nl	LM baseline	0.6958	0.7195	0.9773
europarl200k	en	nl	l1r1	0.7790	0.7941	0.9814
europarl200k	en	nl	l1r1+LM	**0.7838**	**0.7973**	**0.9818**
europarl200k	en	nl	auto	0.7796	0.7947	0.9815
europarl200k	en	nl	auto+LM	0.7812	0.7954	0.9816
europarl200k	en	fr	baseline	0.5874	0.6403	0.9709
europarl200k	en	fr	LM baseline	0.7054	0.7319	0.9787
europarl200k	en	fr	l1r1	0.7416	0.7698	0.9797
europarl200k	en	fr	l1r1+LM	**0.7680**	**0.7885**	**0.9815**
europarl200k	en	fr	auto	0.7484	0.7737	0.9801
europarl200k	en	fr	auto+LM	0.7654	0.7860	0.9813
iwslt12ted	en	zh	baseline	0.6622	0.7122	**0.6421**
iwslt12ted	en	zh	LM baseline	0.6550	0.6982	0.6416
iwslt12ted	en	zh	l1r1	0.7150	0.7531	0.5736
iwslt12ted	en	zh	l1r1+LM	**0.7296**	**0.7619**	0.5826
iwslt12ted	en	zh	auto	0.7150	0.7519	0.5746
iwslt12ted	en	zh	auto+LM	0.7280	0.7605	0.5833
iwslt12ted	zh	en	baseline	0.5784	0.6167	0.9634
iwslt12ted	zh	en	LM baseline	0.6148	0.6463	0.9656
iwslt12ted	zh	en	l1r1	0.7104	0.7338	0.9709
iwslt12ted	zh	en	l1r1+LM	**0.7270**	**0.7460**	**0.9721**
iwslt12ted	zh	en	auto	0.7078	0.7319	0.9709
iwslt12ted	zh	en	auto+LM	0.7230	0.7428	0.9719

Table 6: Results on different datasets and language pairs. The `iwslt12ted` set is the dataset used in the IWSLT 2012 Evaluation Campaign (Federico et al., 2012), and is formed by a collection of transcriptions of TED talks. Here we used of just over $70,000$ sentences for training. Recall for each of the four datasets is 0.9498 (en-nl), 0.9494 (en-fr), 0.9386 (en-zh), and 0.9366 (zh-en)

cesful may be attributed to the increased sparsity that comes from such an expansion.

The idea of a comprehensive translation assistance system may extend beyond the translation of L1 fragments in an L2 context. There are more NLP components that might play a role if such a system were to find practical application. Word completion or predictive editing (in combination with error correction) would for instance seem an indispensable part of such a system, and can be implemented alongside the technique proposed in this study. A point of more practically-oriented future research is to see how feasible such combinations are and what techniques can be used.

An application of our idea outside the area of translation assistance is post-correction of the output of some MT systems that, as a last-resort heuristic, copy source words or phrases into their output, producing precisely the kind of input our system is trained on. Our classification-based approach may be able to resolve some of these cases operating as an add-on to a regular MT system – or as a independent post-correction system.

Our system allows L1 fragments to be of arbitrary length. If a fragment was not seen during training stage, and is therefore not covered by a classifier expert, then the system will be unable to translate it. Nevertheless, if a longer L1 fragment can be decomposed into subfragments that are known, then some recombination of the translations of said sub-fragments may be a good translation for the whole. We are currently exploring this line of investigation, in which the gap with MT narrows further.

Finally, an important line of future research is the creation of a more representative test set. Lacking an interactive system that actually does what we emulate, we hypothesise that good approximations would be to use gap exercises, or cloze tests, that test specific aspects difficulties in language learning. Similarly, we may use L2 learner corpora with annotations of code-switching points or errors. Here we then assume that places where L2 errors occur may be indicative of places where L2 learners are in some trouble, and might want to fall back to generating L1. By then manually translating gaps or such problematic fragments into L1 we hope to establish a more realistic test set.

References

D. W. Aha, D. Kibler, and M. K. Albert. 1991. Instance-based learning algorithms. *Machine*

Learning, 06(1):37–66, January.

S. Barrachina, O. Bender, F. Casacuberta, J. Civera, E. Cubel, S. Khadivi, A. L. Lagarda, H. Ney, J. Tomás, E. Vidal, and J.M. Vilar. 2009. Statistical approaches to computer-assisted translation. *Computational Linguistics*, 35(1):3–28.

M. Carpuat and D. Wu. 2007. Improving statistical machine translation using word sense disambiguation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 61–72.

W. Daelemans, J. Zavrel, K. van der Sloot, and A. van den Bosch. 2009. TiMBL: Tilburg memory based learner, version 6.2, reference guide. Technical Report ILK 09-01, ILK Research Group, Tilburg University.

B. Decadt, V. Hoste, W. Daelemans, and A. van den Bosch. 2004. GAMBL, genetic algorithm optimization of memory-based WSD. In R. Mihalcea and P. Edmonds, editors, *Proceedings of the Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text (Senseval-3)*, pages 108–112, New Brunswick, NJ. ACL.

M. Federico, M. Cettolo, L. Bentivogli, M. Paul, and S. Stüker. 2012. Overview of the IWSLT 2012 evaluation campaign. In *Proceedings of the seventh International Workshop on Spoken Language Translation (IWSLT)*, pages 12–33.

J. Giménez and L. Màrquez. 2007. Context-aware discriminative phrase selection for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 159–166, Prague, Czech Republic, June. Association for Computational Linguistics.

R. Haque, S. Kumar Naskar, A. van den Bosch, and A. Way. 2011. Integrating source-language context into phrase-based statistical machine translation. *Machine Translation*, 25(3):239–285, September.

V. Hoste, I. Hendrickx, W. Daelemans, and A. van den Bosch. 2002. Parameter optimization for machine learning of word sense disambiguation. *Natural Language Engineering*, 8(4):311–325.

P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic, June. Association for Computational Linguistics.

P. Koehn. 2004. Statistical significance tests for machine translation evaluation. In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 388–395, Barcelona, Spain, July. Association for Computational Linguistics.

P. Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *In Proceedings of the Machine Translation Summit X ([MT]'05).*, pages 79–86.

A. Laghos and Z. Panayiotis. 2005. Computer assisted/aided language learning. pages 331–336.

M. Levy. 1997. *Computer-assisted language learning: Context and conceptualization.* Oxford: Clarendon Press.

H. Tou Ng and H. Beng Lee. 1996. Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach. In *ACL*, pages 40–47.

F.J. Och and H. Ney. 2000. Giza++: Training of statistical translation models. Technical report, RWTH Aachen, University of Technology.

F. J. Och and H. Ney. 2003. A systematic comparison of various statistical alignment models. *Comput. Linguist.*, 29(1):19–51, March.

A. Stolcke. 2002. Srilm - an extensible language modeling toolkit. In John H. L. Hansen and Bryan L. Pellom, editors, *7th International Conference on Spoken Language Processing, ICSLP2002 - INTERSPEECH 2002, Denver, Colorado, USA, September 16-20, 2002*. ISCA.

N. Stroppa, A. van den Bosch, and A. Way. 2007. Exploiting source similarity for SMT using context-informed features. In A. Way and B. Gawronska, editors, *Proceedings of the 11th International Conference on Theoretical Issues in Machine Translation (TMI 2007)*, pages 231–240, Skövde, Sweden.

M. van Gompel and A. van den Bosch. 2013. WSD2: Parameter optimisation for memory-based cross-lingual word-sense disambiguation. In *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval 2013), in conjunction with the Second Joint Conference on Lexical and Computational Semantics*.