Document-Level Machine Translation with Word Vector Models

Eva Martínez Garcia, Cristina España-Bonet
TALP Research Center
Univesitat Politècnica de Catalunya
Jordi Girona, 1-3, 08034 Barcelona, Spain
{emartinez,cristinae}@cs.upc.edu

Lluís Màrquez
Arabic Language Technologies
Qatar Computing Research Institute
Tornado Tower, Floor 10
P.O. Box 5825, Doha, Qatar
lmarquez@qf.org.qa

Abstract

In this paper we apply distributional semantic information to document-level machine translation. We train monolingual and bilingual word vector models on large corpora and we evaluate them first in a cross-lingual lexical substitution task and then on the final translation task. For translation, we incorporate the semantic information in a statistical document-level decoder (Docent), by enforcing translation choices that are semantically similar to the context. As expected, the bilingual word vector models are more appropriate for the purpose of translation. The final document-level translator incorporating the semantic model outperforms the basic Docent (without semantics) and also performs slightly over a standard sentence-level SMT system in terms of ULC (the average of a set of standard automatic evaluation metrics for MT). Finally, we also present some manual analysis of the translations of some concrete documents.

1 Introduction

Document-level information is usually lost during the translation process when using Statistical Machine Translation (SMT) sentence-based systems (Hardmeier, 2014; Webber, 2014). Cross-sentence dependencies are totally ignored, as they translate sentence by sentence without taking into account any document context when choosing the best translation. Some simple phenomena like coreferent pronouns outside a sentence cannot be properly translated in this way, which is already important because the correct translation of pronouns in a document confers a high level of coherence to the final translation. Also, discourse connectives are valuable because they mark the flow of the discourse in a text. It is desirable to transfer them to the output translation in order to maintain the characteristics of the discourse. The evolution of the topic through a text is also an important feature to preserve.

All these aspects can be used to improve the translation quality by trying to assure coherence throughout a document. Several recent works go on that direction. Some of them present post-processing approaches making changes into a first translation according to document-level information (Martínez-Garcia et al., 2014a; Xiao et al., 2011). Others introduce the information within the decoder, by, for instance, implementing a topic-based cache approach (Gong et al., 2011; Xiong et al., 2015). The decoding methodology itself can be changed. This is the case of a document-oriented decoder, Docent (Hardmeier et al., 2013), which implements a search in the space of translations of a whole document. This framework allows us to consider features that apply at document level. One of the main goals of this paper is to take advantage of this capability to include semantic information at decoding time.

We present here the usage of a semantic representation based on word embeddings as a language model within a document-oriented decoder. To do this, we trained a word vector model (WVM) using neural networks. As a first approach, a monolingual model is used in analogy with the standard monolingual language models based on n-grams of words instead of vectors. However, to better approach translation, bilingual models are built. These models are avaluated in isolation outside the decoder by means of a cross-lingual evaluation task that resembles a translation environment. Finally, we use these models in a translation task and we observe how the semantic information enclosed in them help to improve translation quality.

The paper is organized as follows. A brief revision of the related work is done in Section 2. In Section 3, we describe our approach of using a bilingual word vector model as a language model. The model is compared to monolingual models and evaluated. We show and discuss the results of our experiments on the full translation task in Section 5. Finally, we draw the conclusions and define several lines of future work in Section 6.

2 Related Work

In the last years, approaches to document-level translation have started to emerge. The earliest ones deal with pronominal anaphora within an SMT system (Hardmeier and Federico, 2010; Nagard and Koehn, 2010). These authors develop models that, with the help of coreference resolution methods, identify links among words in a text and use them for a better translation of pronouns. More recent approaches focus on topic cohesion. (Gong et al., 2011) tackle the problem by making available to the decoder the previous translations at decoding time using a cache system. In this way, one can bias the system towards the lexicon already used. (Xiong et al., 2015) also present a topic-based coherence improvement for an SMT system by trying to preserve the continuity of sentence topics in the translation. To do that, they extract a coherence chain from the source document and, taking this coherence chain as a reference, they predict the target coherence chain by adapting a maximum entropy classifier. Document-level translation can also be seen as the post-process of an already translated document. In (Xiao et al., 2011; Martínez-Garcia et al., 2014a), they study the translation consistency of a document and retranslate source words that have been translated in different ways within a same document. The aim is to incorporate document contexts into an existing SMT system following 3 steps. First, they identify the ambiguous words; then, they obtain a set of consistent translations for each word according to the distribution of the word over the target document; and finally, generate the new translation taking into account the results of the first two steps.

These approaches report improvements in the final translations but, in most of them. the improvements can only be seen through a detailed manual evaluation. When using automatic evaluation metrics like BLEU (Papineni et al., 2002), differences are not significant.

A document-oriented SMT decoder is presented in (Hardmeier et al., 2012; Hardmeier et al., 2013). The decoder is built on top of an open-source phrase-based SMT decoder, Moses (Koehn et al., 2007). The authors present a stochastic local search decoding method for phrase-based SMT systems which allows decoding complete documents. Docent starts from an initial state (translation) given by Moses and this one is modified by the application of a hill climbing strategy to find a (local) maximum of the score function. The score function and some defined change operations are the ones encoding the document-level information. One remarkable characteristic of this decoder, besides the change of perspective in the implementation from sentence-level to document-level, is that it allows the usage of a WVM as a Semantic Space Language Model (SSLM). In this case, the decoder uses the information of the word vector model to evaluate the adequacy of a word inside a translation by calculating the distance among the current word and its context.

In the last years, several distributed word representation models have been introduced. Furthermore, distributed models have been successfully applied to several different NLP tasks. These models are able to capture and combine the semantic information of the text. An efficient implementation of the Context Bag of Words (CBOW) and the Skipgram algorithms is presented in (Mikolov et al., 2013a; Mikolov et al., 2013c; Mikolov et al., 2013d). Within this implementation WVMs are trained using a neural network. These models proved to be robust and powerful to predict semantic relations between words even across languages. They are implemented inside the *word2vec* software package. However, they are not able to handle lexical ambiguity as they conflate word senses of polysemous words into one common representation. This limitation is already discussed in (Mikolov et al., 2013b) and in (Wolf et al., 2014), in which bilingual extensions of the *word2vec* architecture are also proposed. These bilingual extensions of the models consist of a combination of two monolingual models. They combine the source vector model and the target vector model by training a new neural network. This network is able to learn the projection matrix that combines the information of both languages. A new bilingual approach is presented in (Martínez-Garcia et al., 2014b). Also, the resulting models are evaluated in a cross-lingual lexical substitution task as well as measuring their accuracy when capturing words semantic relationships.

Recently, Neural Machine Translation (NMT) has appeared as a powerful alternative to other MT techniques. Its success lies on the excellent results that deep neural networks have achieved in natural language tasks as well as in other areas. In short, NMT systems are build over a trained neural network that is able to output a translation given a source text in the input (Sutskever et al., 2014b; Sutskever et al., 2013; Bahdanau et al., 2014; Cho et al., 2014). However, these systems report some problems when translating unknown or rare words. We are aware of only few works that try to address this problem (Sutskever et al., 2014a; Jean et al., 2014).

Furthermore, there are some works that try to use vector models trained using recurrent neural networks (RNN) to improve decoder outputs. For instance, in (Sundermeyer et al., 2014) they build two kinds of models at word level, one based on word alignments and other one phrase-based. The authors train RNNs to obtain their models and they use them to rerank n-best lists after decoding. They report improvements in BLEU and TER scores in several language pairs, but they are not worried about context issues of a document although they do take into account both sides of the translation: source and target. In (Devlin et al., 2014) they also present joint models that augment the NNLM with a source context window to introduce a new decoding feature. They finally present improvements in BLEU score for Arabic-English language pair and show a new technique to introduce this kind of models inside MT systems in a computationally efficient way. These two last works prove the power of applying NN models as features inside MT systems.

3 Training monolingual and bilingual semantic models

As we explained before, there are several works that use monolingual WVM as language models, or the composition of monoligual models to build bilingual ones. This section shows a methodology to build directly bilingual models.

3.1 Bilingual word vector models

For our experiments we use the two algorithms implemented in the *word2vec* package, Skipgram and CBOW.

The Skipgram model trains a NN to predict the context of a given word. On the other hand, the CBOW algorithm uses a NN to predict a word given a set of its surrounding words, where the order of the words in the history does not inuence the projection.

In order to introduce semantic information in a bilingual scenario, we use a parallel corpus and automatic word alignment to extract a new training corpus of word pairs: $(w_{i,T}|w_{i,S})$. For instance, if the words *house* and *casa* are aligned in a document, we consider the new form *casa|house*.

This approach is different from (Wolf et al., 2014) who build an independent model for each language. With our method, we try to capture simultaneously the semantic information associated to the source word and the information in the target side of the translation. In this way, we hope to better capture the semantic information that is implicitly given by translating a text. To better characterize ambiguous words for MT, for instance, we expect to be able to distinguish among the different meanings that the word *desk* can have when translated in Spanish: *desk|mesa* vs. *desk|mostrador* vs. *desk|escritorio*.

3.2 Settings

The training set for our models is built from parallel corpora in the English-Spanish language pair available in Opus [1] (Tiedemann, 2012; Tiedemann, 2009). These corpora have been automatically aligned and therefore contain the aligment information necessary to build our bilingual models. We chose the one-to-one alignments to avoid noise and duplicities in the final data. Table 1 shows the size of the specific data used: EuropalV7, United Nations, Multilingual United Nations, and Subtitles-2012. Monolingual models are also build with these corpora and therefore are comparable in size. With this corpus, the final training set has 584 million words for English and 759 for Spanish.

[1] http://opus.lingfil.uu.se/

	Corpus	Documents	Sentences	English Tokens	Spanish Tokens
Training	Europarl-v7	–	1,965,734	49,093,806	51,575,748
	UN	–	61,123	5,970,000	6,580,000
	Multi UN	73,047	9,275,905	554,860,000	621,020,000
	Subtitles-2012	46,884	24,929,151	306,600,000	498,190,000
Development	NC-2009	136	2,525	65,595	68,089
Test	NC-2011	110	3,003	65,829	69,889

Table 1: Figures on the corpora used for training, development and test.

For training the models, we set to 600 the dimensionality of our vectors and we used a context window of 5 during the training (2 words before and 2 words after). Previous work (Martínez-Garcia et al., 2014b) and related experiments showed the adequacy of these parameters.

4 Cross-Lingual Lexical Subsitution Task

We evaluate the generated models described in Section 3 in a cross-lingual lexical substitution exercise. In order to do this, first, the content words of the test set which are translated in more than one different way by a baseline translation system are identified (see Section 5 for the description of the baseline system). We call these words ambiguous. The task consists in choosing the adequate translation from the set of ambiguous words. In our case, the correct choice is given by the reference translation of the test set.

To give an example, the word *desk* appears many times in a newswire document about a massive complaining for exaggerated rents. This word has here the meaning of *a service counter or table in a public building, such as a hotel*[2]. The correct translation to that meaning in Spanish would be the word *mostrador* or *ventanilla*. But, we can see that in the output of a SMT system, besides the correct translations, *desk* can appear translated as *mesa* or even as *escritorio* in the same document. If the reference translation contains *mostrador*, only this word will be considered correct in the evaluation.

Once we have identified the words that we want to translate with the vector models, we get their context target words and their aligned source word and look for vector associated to the $sw|tw$ form in our bilingual model. Then, we build a context vector as the sum of the vectors of the surrounding target words and use it to choose among the set of translation options (all the options seen within the document). We choose the best translation as the one that has associated the vector which is the closest to the context vector.

Model	Top 1	Top 5
mono CBOW	47.71%	65.44%
mono Skipgram	47.71%	59.19%
bi CBOW	62.39%	85.49%
bi Skipgram	62.39%	78.36%

Table 2: Evaluation of the *word2vec* vector models. Top 1 and Top 5 accuracies of the monolingual (mono rows) in Spanish and the bilingual (bi rows) English–Spanish models trained using CBOW or Skipgram.

4.1 Results

This task is evaluated on the NewsCommentaries-2011 test set. Table 2 shows the results of the evaluation of our bilingual (*bi*) model in comparison to a monolingual (*mono*) model trained in Spanish. The accuracies show the performance of our models on the ambiguous words. For this test set, we find 8.12% of ambiguous words and, in average, 3.26 options per ambiguous word. We skip some adverbials, common verbs, the prepositions and conjunctions as ambiguous words to avoid noise in the results. In average, the monolingual model has a coverage of 90.97% and the bilingual 87.53% for this test set. Regarding to the ambiguous words, 83.97% of them are known for the bilingual model and a 87.37% for the monolingual.

The two *word2vec* algorithms have the same performance for this task when they suggest only the best option, an accuracy of 47.71% for the monolingual model and 62.39% for the bilingual one. So, bilingual models are encoding significantly more semantic information than monolingual models. It has to be said that here the most frequent translation option achieves a 59.76% of

[2] Definition taken from Collins Concise English Dictionary.

accuracy. So, it is only with bilingual models that we beat the frequentist approach.

Accuracies are significantly improved when more options are taken into account. When looking at the accuracy at Top 5, CBOW achieves 65.44% in the monolingual task and 85.49% in the bilingual one, whereas the Skipgram models have 6 less points in the monolingual case and 13 in the bilingual one. These results indicate that CBOW bilingual models are capturing better the semantics and that considering more than one option can be important in the full translation task.

5 Vector Models for Document-level Translation

We evaluate in this section the use of the word vector models described in Section 3 as language models within a document-level MT system.

5.1 Vector models as Semantic Space Language Models in Docent

The Docent decoder allows us to use a dense word vector model as a semantic language model. This language model implementation tries to reward the word choices that are closer to their context.

In a similar way to the evaluation task explained in Section 3, these models calculate a score for every word in a document translation candidate. This score is calculated as the cosine similarity between the vector representation of the word and the sum of the vectors of the previous 30 words. This parameter makes possible that the context crosses sentence boundaries. The score produced by the semantic space language model is $h(w|h) = \alpha p_{cos}(w|h)$ if w is a known word, and $h(w|h) = \epsilon$ if w is an unknown word, where α is the proportion of content words in the training corpus and ϵ is a small fixed probability, as described in (Hardmeier, 2014).

The assumption is the same here as before, the better the choice, the closer the context vector will be to the vector representation of the evaluated word. The final score for a document translation candidate is an average of the scores of its words.

5.2 Experimental Settings

Our SMT baseline system is based on Moses. The translation system has been trained with the Europarl corpus in its version 7 for the Spanish–English language pair. We used the GIZA++ software (Och and Ney, 2003) to do the word alignments. The language model is an interpolation of several 5-gram language models obtained using SRILM (Stolcke, 2002) with interpolated Kneser-Ney discounting on the target side of the Europarl corpus v7; United Nations; NewsCommentary 2007, 2008, 2009 and 2010; AFP, APW and Xinhua corpora as given by (Specia et al., 2013)[3] The optimization of the weights is done with MERT (Och, 2003) against the BLEU measure on the NewsCommentary corpus of 2009. As in the previous section, our experiments are carried out over the NewsCommentary-2011 test set. We chose the newswire documents as test set because typically they are documents with high consistency and coherence.

Regarding the document-level decoder, we use Docent. The first step in the Docent translation process is the output of our Moses baseline system. We set the initial Docent weights to be the same as the ones obtained with MERT for the Moses baseline. Finally, the word vector models used in the experiments of this section are the ones that we describe and evaluate in Section 3 using the CBOW algorithm.

5.3 Results

Table 3 shows the automatic evaluation obtained with the Asiya toolkit (González et al., 2012) for several lexical metrics (BLEU, NIST, TER, METEOR and ROUGE), a syntactic metric based on the overlap of PoS elements (SP-Op), and an average of a set of 21 lexical and syntactic metrics (ULC), including all the previous measures and many more. The first row shows the results for the Moses baseline system. The second row shows the evaluation of the Docent baseline system working with the baseline Moses output as first step. This Docent system uses only the default features that are equivalent to the ones in the Moses system but without lexical reordering. The last two rows show the evaluation of our extensions for the Docent decoder using both, monolingual vector models as semantic space language models (Docent + monoSSM) and the bilingual ones (Docent + biSSM). The results show only slight differences among the systems. However, these differences reflect the impact of our word embeddings in the translation process and are consistent across metrics. The differences are statistically signifi-

[3] Resources are available in:
http://statmt.org/wmt13/qualityestimationtask.html

system	BLEU	NIST	TER	METEOR	ROUGE	SP-Op	ULC
Moses	28.60	7.54	72.17	23.41	30.20	19.99	77.76
Docent	28.33	7.46	72.83	23.22	30.36	19.38	77.14
Docent + monoSSM	28.48	7.52	72.61	23.28	30.33	19.61	77.49
Docent + biSSM	28.58	7.66	72.56	23.31	30.38	19.78	77.89

Table 3: Automatic evaluation of the systems. See text for the system and metrics definition.

newswire	Moses	Docent	Docent+monoSSM	Docent+biSSM
news79	47.88	48.10	47.07	48.00
news88	24.18	24.60	24.18	23.26
news104	35.53	35.71	35.58	36.00
news107	19.52	19.57	19.58	19.66
news27	14.45	14.22	14.27	14.83
news68	38.91	38.39	38.58	39.73

Table 4: Evaluation of the different systems using BLEU metric on some individual newswire documents extracted from the NewsCommentary-2011 test set.

cant at the 90% confidence level, but not at higher level, between Moses and all Docent systems and, also, between the Docent baseline and both extended Docent systems. We observed that by using boostrap-resampling over BLEU and NIST metrics as described in (Koehn, 2004). We observe that Docent systems have a positive trend in their performance as long as we introduce models with more information (from only monolingual to bilingual).

Looking a little bit closer at each system, we observe that monolingual models do help Docent to find better document translation candidates. They are able to improve 0.15 point in BLEU, which is a lexical metric that is usually not sensible to document-level changes (Martínez-Garcia et al., 2014a) and also they gain 0.41 points in the syntactic metric. In a similar way, bilingual models improve a little bit more the performance over the monolingual models. In particular, they show an improvement of 0.10 in BLEU with respect to the monolingual models and 0.25 points with respect to the Docent baseline system. We observe also a similar behaviour for the rest of the metrics. For instance, regarding to the syntactic metric based on the overlap of PoS elements (SP-Op), bilingual models are able to recover 0.50 points with respect to the Docent baseline system and 0.15 points respect to the system with the monolingual models. For the average metric, ULC, the best system is Docent+biSSM, being 0.13 point over Moses and 0.75 over Docent. However, in general, there is first a slight decrease in translation quality when going from the sentence-based decoder to the document-based one probably due to the fact that Docent is not currently supporting lexicalized reordering.

In summary, we conclude from these results that the semantic information captured by our vector models help the document-level translation decoder. We also observe that bilingual models capture valuable information from the aligned data that came from the first step translation. This behaviour is coherent with the previous evaluation of the models showed in Section 3.

Table 4 shows the BLEU scores for some particular documents with some interesting cases. These results reflect the behaviour of our systems. We found some documents where the Docent systems cannot improve the Moses translation. For instance, the phrase "*House of Bones*" appears in a document about a famous building. Its correct translation is "*Casa de los Huesos*". However, Moses translates it as "*Cámara de huesos*" and Docent systems only suggest a new incorrect option "*Asamblea de huesos*". On the other hand, we find many examples where word vector models are helping. For instance, in the example of *desk* that we mentioned in Section 3, it is translated as *mostrador*, *mesa* and *escritorio* by Moses. Using the Docent baseline, it appears translated as *escritorio* and *mesa*. That shows how Docent is controling the coherence level of the translation. Using the Docent extended with the monolingual model, it appears as *escritorio*, *mesa* and *taquilla*. The word vector language model helps the system to change one translation option for a more correct one. Finally, using the bilingual vector model, we observe the word translated as *mostrador*, *mesa* and *taquilla*, obtaining here 2 good translation instead of only one. This shows how the bilingual information helps to obtain better translations. We observe how monolingual vector models improve the Docent base translation and, at the same time, how the bilingual information helps to improve the translation and even obtain better results than the ones with the Moses baseline.

6 Conclusions

We have presented an evaluation of word vector models trained with neural networks. We test them in a document-level machine translation environment. First, we build monolingual and bilingual models using the *word2vec* package implementations for the CBOW and the Skipgram algorithms. We test the models to see their capability to select a good translation option for a word that appears translated in more than one sense in a first translation of a document. The results of these evaluations show that the CBOW models perform better than the Skipgram one in our test set, achieving at most 85.49% and 78.36% respectively for the bilingual model for the accuracy at Top 5. Also, the bilingual model achieves better results than the monolingual one, with a 65.44% of accuracy for the best monolingual model trained with CBOW against the 85.49% for the bilingual model under the same conditions. These results indicate that WSM can be useful for translation tasks and it is left as future work a wider evaluation of the models considering the variation of all the parameters (context training window, vectors dimensionality, size and quality of the training data, etc.) We also want to use other techniques, like the semisupervised approach described in (Madhyastha et al., 2014), to build new bilingual models in order to compare them with the ones that are presented here.

As a second step of the process, we evaluated our word vector models inside a machine translation system. In particular, we chose the Docent decoder since it works at document-level and allows a fast integration of WVMs as semantic space language models. This option allows us to asses the vector models quality in a specific translation environment. The carried out experiments showed that WVMs models can help the decoder to improve the final translation. Although we only observe a slight improvement in the results in terms of automatic evaluation metrics, the improvement is consistent among metrics and is larger as we introduce more semantic information into the system. That is, we get the best results when using the models with bilingual information.

Summing up, the evaluation has shown the utility of word vector models for translation-related tasks. However, the results also indicate that these systems can be improved. We left as future work the effect that bilingual WVMs obtained with other methods can have in the final translation. Also, we find it interesting to apply these models to a particular document-level phenomenon such as ambiguous words. Developing a specific feature for Docent that scores the adequacy of a translation option for every ambiguous word in a document using word vector models can improve the performance of such models for translation tasks.

Acknowledgments

Supported by the TACARDI project (TIN2012-38523-C02) of the Spanish Ministerio de Economía y Competitividad (MEC). Special thanks to Christian Hardmeier and Jörg Tiedemann for their insightful comments and technical support.

References

Bahdanau, D., K. Cho, and Y. Bengio. 2014. Neural machine translation by jointly learning to align and translate. In *arXiv:1409.0473*.

Cho, K., B. van Merrienboer, D. Bahdanau, and Y. Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. In *Proc. of the 8th SSST*, pages 103–111.

Devlin, J., R. Zbib, Z. Huang, T. Lamar, R. Schwartz, and J. Makhoul. 2014. Fast and robust neural network joint models for statistical machine translation. In *Proc. of the 52nd ACL (Vol 1.)*, pages 1370–1380.

Gong, Z., M. Zhang, and G. Zhou. 2011. Cache-based document-level statistical machine translation. In *Proc. of the 2011 EMNLP*, pages 909–919.

González, M., J. Giménez, and L. Màrquez. 2012. A graphical interface for MT evaluation and error analysis. In *Proc. of the 50th ACL, System Demonstrations*, pages 139–144.

Hardmeier, C. and M. Federico. 2010. Modelling pronominal anaphora in statistical machine translation. In *Proc. of the 7th International Workshop on Spoken Language Translation*, pages 283–289.

Hardmeier, C., J. Nivre, and J. Tiedemann. 2012. Document-wide decoding for phrase-based statistical machine translation. In *Proc. of the EMNLP-CoNLL*, pages 1179–1190.

Hardmeier, C., S. Stymne, J. Tiedemann, and J. Nivre. 2013. Docent: A document-level decoder for phrase-based statistical machine translation. In *Proc. of the 51st ACL*, pages 193–198.

Hardmeier, C. 2014. Discourse in machine translation (PhD Thesis). Uppsala Universitet.

Jean, S., K. Cho, R. Memisevic, and Y. Bengio. 2014. On using very large target vocabulary for neural machine translation. In *arXiv (abs/1412.2007).*

Koehn, P., 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 *Proc. of the 45th ACL*, pages 177–180.

Koehn, Philipp. 2004. Statistical significance tests for machine translation evaluation. In *Proc. of EMNLP 2004*, pages 388–395.

Madhyastha, P. S., X. Carreras, and A. Quattoni. 2014. Learning task-specific bilexical embeddings. In *Proc. of the 25th COLING*, pages 161–171.

Martínez-Garcia, E., C. España-Bonet, and L. Màrquez. 2014a. Document-level machine translation as a re-translation process. In *Procesamiento del Lenguaje Natural, Vol. 53*, pages 103–110. SEPLN.

Martínez-Garcia, E., C. España-Bonet, J. Tiedemann, and L. Màrquez. 2014b. Word's vector representations meet machine translation. In *Proc. of the 8th SSST*, pages 132–134.

Mikolov, T., K. Chen, G. Corrado, and J. Dean. 2013a. Efficient estimation of word representations in vector space. In *Proc. of Workshop at ICLR*. http://code.google.com/p/word2vec.

Mikolov, T., Q. V. Le, and I. Sutskever. 2013b. Exploiting similarities among languages for machine translation. In *arXiv:1309.4168.*

Mikolov, T., I. Sutskever, G. Corrado, and J. Dean. 2013c. Distributed representations of words and phrases and their compositionality. In *Proc. of NIPS*, pages 3111–3119.

Mikolov, T., W. Yih, and G. Zweig. 2013d. Linguistic regularities in continuous space word representations. In *Proc. of NAACL HLT*, pages 746–751.

Nagard, R. Le and P. Koehn. 2010. Aiding pronouns translation with co-reference resolution. In *Proc. of Joint 5th Workshop on Statistical Machine Translation and MetricsMATR*, pages 252–261.

Och, F. and H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics (Vol. 29).*

Och, F. 2003. Minimum error rate training in statistical machine translation. In *Proc. of the ACL.*

Papineni, K., S. Roukos, T. Ward, and W.J. Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proc. of the 40th ACL*, pages 311–318.

Specia, Lucia, Kashif Shah, Jose G. C. De Souza, and Trevor Cohn. 2013. Quest - a translation quality estimation framework. In *Proc. of ACL Demo Session*, pages 79–84.

Stolcke, A. 2002. SRILM – An extensible language modeling toolkit. In *Proc. Intl. Conf. on Spoken Language Processing*, pages 257–286.

Sundermeyer, M., T. Alkhouli, J. Wuebker, and H. Ney. 2014. Translation modeling with bidirectional recurrent neural networks. In *Proc. of the 2014 EMNLP*, pages 14–25. EAMNLP.

Sutskever, I., O. Vinyals, and V. Le Quoc. 2013. Recurrent continuous translation models. In *Proc. of EMNLP*, pages 1700–1709.

Sutskever, I., V. Le Quoc, O. Vinyals, and W.Zaremba. 2014a. Addressing the rareword problem in neural machine translation. In *arXiv:1410.8206.*

Sutskever, I., O. Vinyals, and V. Le Quoc. 2014b. Sequence to sequence learning with neural networks. In *Proc. NIPS 2014*, pages 1422–1430.

Tiedemann, J. 2009. News from opus - a collection of multilingual parallel corpora with tools and interfaces. In *Recent Advances in Natural Language Processing (vol V)*, pages 237–248.

Tiedemann, J. 2012. Parallel data, tools and interfaces in opus. In *Proc. of the 8th LREC*, pages 2214–2218. http://opus.lingfil.uu.se/.

Webber, B. 2014. Discourse for machine translation. In *Proc. of the 28th PACLIC.*

Wolf, L., Y. Hanani, K. Bar, and N. Derschowitz. 2014. Joint word2vec networks for bilingual semantic representations. In *Poster sessions at CICLING.*

Xiao, T., J. Zhu, S. Yao, and H. Zhang. 2011. Document-level consistency verification in machine translation. In *Proc. of MT-Summit XIII*, pages 131–138.

Xiong, D., M. Zhang, and X. Wang. 2015. Topic-based coherence modeling for statistical machine translation. In *IEEE/ACM Transactions on audio, speech and language processing (Vol. 23)*, pages 483 – 493.