

Arabic-English translation improvement by target-side neural network language modeling

Maxim Khalilov, José A. R. Fonollosa

F. Zamora-Martínez, María J. Castro-Bleda, S. España-Boquera

Centre de Recerca TALP, Universitat Politècnica de Catalunya
Barcelona, Spain

Dep. de Lenguajes y Sistemas Informáticos, Universidad Politécnica de Valencia
Valencia, Spain

Abstract

The quality of translation, produced by Statistical Machine Translation (SMT) systems, crucially depends on generalization, provided by the statistical models involved into translation process. In this study we present the n -gram based translation system (i.e. the UPC SMT system), enhanced with a continuous space language model (LM), estimated with a neural network (NN). In the framework of the study, we use NN LM on the rescoring step, reevaluating the N -best list of complete translations hypothesis. Different word history length included in the model (n -gram order) and distinct continuous space representation (i.e. including words, appearing in the training corpus more than k times) are considered in the paper. We report result for an Arabic-English translation task, improving Arabic-English translation accuracy by better target language model representation in contrast with the state-of-the-art approach. The experimental results are evaluated by means of automatic evaluation metrics correlated with fluency and adequacy of the generated translations.

1. Introduction

Language modeling is an essential step in many Natural Language Processing applications, and, particularly in the Statistical Machine Translation (SMT) task.

The most widespread and popular technique for language model estimation in the state-of-the-art SMT systems is the so-called n -gram approach, which assign high probability to frequent sequences of words by considering the history of only $n-1$ preceding words in the utterance. On the contrast, the approach presented in the paper can be considered as a coherent and natural evolution of the probabilistic Language Model (LM): we propose to use a continuous LM trained in the form of a neural network (NN). The idea of continuous space representation of language is not new, some successful attempts of NN model application to language modeling has been recently made (Xu and Jelinek, 2004; Bengio et al., 2003; Castro and Prat, 2003). However, the use of NN LM in the state-of-the-art SMT systems is not so popular, due to its high computational cost. The only comprehensive work refers to (Schwenk et al., 2006b; Schwenk et al., 2006a), where the target LM is presented in the form of fully-connected multi-layer perceptron.

Our work addresses the improvement of Arabic to English translation which is considered as a complex translation task since high dissimilarity between the source and the target languages. Classical Arabic, like Modern Standard Arabic, is a VSO (verb-subject-object) pro-drop language with rich templatic morphology where words are made up of roots and affixes and clitics agglutinate to words, while English follows the SVO (subject-verb-object) word order and is a non-pro-drop language with less affluent morphology, but with a higher number of irregular verbs.

As mentioned above, NN language models extremely demand for memory resources and computational time. Considering this limitation and be aimed to operate on a fair experimentation field demonstrating a pure impact gener-

ated by the target NN LM, we use a 30 K phrases extraction from the NIST 2008¹ corpus, belonging to the news domain (*Newswire*), that can be characterized as extremely limited amount of training data (about 700 K of tokens in the English part and 640 K in the Arabic part).

The basic idea of the study lies in improving of the Arabic-English translation quality, circumventing difficulties imposed by complex structure of the Arabic language. Instead of dealing directly with Arabic we improve the target language (English) model representation, hopefully improving translation fluency without impairing of adequacy.

The article is structured as follows: in Section 2 we describe the NN LM and give a brief explanation of the training algorithm. In Section 3 we outline the n -gram-based SMT system, Section 4 presents our experimental setup and obtained results, while Section 5 concludes the article with the leading discussions.

2. Neural network language model

Our approach to the widely-used statistical language models based on n -grams consists on using neural networks. A NN LM is a statistical LM which follows the same equation as n -grams:

$$p(w_1 \dots w_{|W|}) \approx \prod_{i=1}^{|W|} p(w_i | w_{i-n+1} \dots w_{i-1}) \quad (1)$$

and where the probabilities that appear in that expression are estimated with a NN. The model naturally fits under the probabilistic interpretation of the outputs of the NNs: if a NN is trained as a classifier, the outputs associated to each class are estimations of the posterior probabilities of the defined classes. The demonstration of this assertion can be found in a number of places, for example, in Bishop (1995).

¹<http://www.nist.gov/speech/tests/mt/2008/>

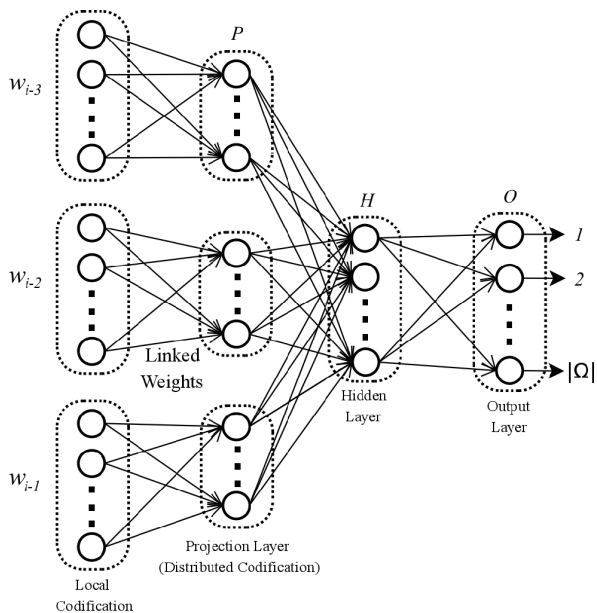


Figure 1: Architecture of the continuous space NN LM. The input words are $w_{i-n+1}, \dots, w_{i-1}$ and P , H and O are the projection, hidden and output layer, respectively.

The training set for a LM is a sequence $w_1 w_2 \dots w_{|W|}$ of words from a vocabulary Ω . In order to train a NN to predict the next word given a history of length $n-1$, each input word must be codified. A natural representation is a local codification following a “1-of- $|\Omega|$ ” scheme. The problem of this codification for tasks with large vocabularies (as is the case) is the huge size of the resulting NN. We have solved this problem following the ideas of Bengio et al. (2003), learning a distributed representation for each word.

Figure 1 illustrates the architecture of the feed-forward NN used to estimate the NN LM. The input is composed of words $w_{i-n+1}, \dots, w_{i-1}$ of equation (1). Each word is represented using a local codification. P is the projection layer of the input words, formed by $P_{i-n+1}, \dots, P_{i-1}$ projection units, which represent the distributed codification of each input word. H denotes the hidden layer and the output layer O has $|\Omega|$ units, one for each word of the vocabulary. This NN, trained as a classifier, predicts the posterior probability of each word of the vocabulary given the history, i.e., $p(w_i | w_{i-n+1} \dots w_{i-1})$.

In order to achieve a good configuration (topology and parameters) for each NN LM in the translation task, exhaustive scanning using a tuning set was performed. The activation function for the hidden layers was the *hyperbolic tangent* function and the *softmax* function for the output units. Best configurations used a projection layer of 32 units for each word. As an example of the huge sizes of the NNs used, the best experiment (see Table 3) used a 4-gram NN LM and a vocabulary of 4 908 tokens (words with less than $k=10$ occurrences were discarded). This NN had 157 088 weights, replicated $n-1$ times at the projection layer, and 325 228 weights at the hidden and output layers.

3. UPC n-gram SMT system

A detailed description of the architecture of the n -gram UPC translation system, which was used in the work, can be found in Crego et al. (2006) and in Mariño et al. (2006). Using *noisy-channel* and *maximum entropy* approaches, it is possible to combine additional feature models in the determination of the translation hypothesis, as shown below:

$$\hat{e}_1^I = \arg \max_{e_1^I} \left\{ \sum_{m=1}^M \lambda_m h_m(e_1^I, f_1^J) \right\} \quad (2)$$

where the feature functions h_m refer to the system models, namely *bilingual translation model*, *target LM* and additional feature models (a *word penalty model*, a *Part-of-Speech (POS) target LM*, a *source-to-target* and a *target-to-source* lexicon models (Och et al., 2004)); the set of λ_m refers to the weights corresponding to these models.

The n -gram translation system is based on a bilingual model, constituting of bilingual units (called tuples). This model approximates the joint probability between the source and the target languages under consideration. The procedure of tuples extraction from a word-to-word alignment (performed with GIZA++ (Och and Ney, 2000)) according to certain constraints is explained in detail in Mariño et al. (2006). In this way the context used in the translation model is bilingual, it not only takes the target sentence into account, but both languages linked in tuples. The translation model can be seen here as a LM, where the language is composed by tuples.

The decoder (called MARIE), an open source tool², implementing a beam search strategy based on dynamic programming and allowing for distortion capabilities was used in the translation system. It takes into account all the feature functions described above, along with the bilingual n -gram translation model. It allows for histogram and threshold pruning and hypothesis recombination.

N -gram-based translation system is highly sensitive to the difference in word order between source and target languages, because of this reason extended monotone distortion model based on automatically extracted reordering patterns was introduced, as presented in Crego and Mariño (2006).

Given the development set and references, the log-linear combination of weights was adjusted using a *simplex* (Nelder and Mead, 1965) optimization method (with the optimization criteria of the highest hybrid metric, based on BLEU (Papineni et al., 2002) and NIST scores (Dodington, 2002)) and an N -best re-ranking just as described in <http://www.statmt.org/jhuws/>. This strategy allows for a faster and more efficient adjustment of model weights by means of a double-loop optimization, which provides significant reduction of the number of translations that should be carried out.

3.1. Arabic data preprocessing

We used a similar approach to that shown in Habash and Sadat (2006), we used the MADA+TOKAN system for disambiguation and tokenization. For disambiguation only

²<http://gps-tsc.upc.es/veu/soft/soft/marie/>

diacritic unigram statistics were employed. For tokenization we used the D3 scheme with -TAGBIES option. The scheme splits the following set of clitics: w+, f+, b+, k+, l+, Al+ and pronominal clitics. The -TAGBIES option produces Bies POS tags on all taggable tokens.

3.2. Rescoring

The NN LM was used on the rescoring/reranking step. Firstly the N -best list of possible translations is generated from the output lattice of the MARIE decoder ($N=1\ 000$). On the second step, the N -best translation hypotheses are reevaluated by adding additional features to the baseline (i.e. NN LM) and discriminatively reranking the translation hypothesis according to the log-linear approach. It allows to obtain a better generalization ability. This feature should be able to better distinguish between higher and lower quality translations.

4. Experiments

The experiment results were obtained on the 30 K phrases extraction from the NIST'07 corpus. Automatic evaluation conditions were case-sensitive with tokenized punctuation marks. The development and test sets were provided with four reference translations and contain 489 and 500 sentences, respectively. A brief statistics of the training corpus in use can be found in table 1.

	Arabic	English
Sentences	30 K	30 K
Words	839.59 K	936.39 K
Average sentence length	27.99	31.22
Vocabulary	43.39 K	33.07 K

Table 1: Basic statistics of the training corpus (30 K extraction from the NIST'08 Arabic-English corpus).

4.1. Baseline

MARIE decoder was used to generate a word lattice which is used afterward to extract the 1 000-best list. The optimization criteria was $100*\text{BLEU} + 4*\text{NIST}$, following the point from (Chen et al., 2005). A target language LM was generated using the SRI Language Modeling Toolkit³ (Stolcke, 2002) on the basis of the considering vocabulary (see Table 1). The following LM configuration was chosen based on the lower perplexity principle (perplexity values were estimated on the concatenation of the development set references): a 4-gram model with *unmodified Kneser-Ney back-off* discounting and counts post-modification after discount estimation (*-kn-modify-counts-at-end* option). This LM was implicitly integrated into the SMT system as a feature function and was considered as a reference baseline. In case of baseline system we did not perform any additional rescoring, instead of it, we extracted the single-best list corresponding to the highest cost that was considered as a translation.

4.2. NN LM experiments

The NN LM models were trained with the April toolkit (España-Boquera et al., 2007), developed for neural networks and pattern recognition tasks in the investigation group of Valencia. Target NN LMs were trained on exactly the same training data as the baseline LM. We considered two key parameters of the continuous NN LM: (a) *word frequency threshold k*: words with less than k occurrences were discarded; (b) *n-gram order*: 3-gram, 4-gram and mixed 3-and 4-gram models were tested.

For the mixed 3-and 4-gram models, several coefficients to combine both models were tested on the tuning set, as shown in equation (3). The best performance was achieved with $\alpha = 0.5$, that corresponds to an equally weighted linear combination of the models:

$$p(w_1 \dots w_{|W|}) \approx \prod_{i=1}^{|W|} \alpha p(w_i | w_{i-2} w_{i-1}) + (1-\alpha) p(w_i | w_{i-3} w_{i-2} w_{i-1}) \quad (3)$$

1 000-best lists generated from the word lattice were rescored with the NN LM. In order to avoid problems of possibly imperfect optimization, different start points were tried and the best set of weights due to the $100*\text{BLEU} + 4*\text{NIST}$ criteria was chosen.

In order to reduce the time needed for the rescoring, we made an attempt to decrease the size of the models, extracting the part related to the distributed codification into an auxiliary table (associating a unique code to each word), and on the other hand maintaining the structure of the net that calculate the LM without codification part. It allowed to reduce the number of the weights by up to half.

4.3. Results

Table 2 shows the perplexity values, obtained on the concatenation of all the references of the development corpus applied to the LMs estimated using SRI LM toolkit and NN techniques. The vocabulary of the $k=10$ systems is 4 908 words, while the $k=12$ models include 4 398 words. BLEU, NIST and METEOR scores, as well as the baseline system translation quality comparative results, are reported in Table 3. *Baseline* refers to the SMT systems with the SRI target LM, including all vocabulary words (see subsection 4.1.).

Our previous experience shows that, at least for small translation tasks with a lack of training material, target LM perplexity reduction leads to a notable improvement in translation quality. As can be observed, considerable improvements were obtained by using a reranking of the 1 000-best list with NN LM: for the major part of the considered NN LMs configurations, BLEU score is higher than for the baseline configuration. Statistical significance threshold⁴ lies on the level of 27.40 and 22.05 BLEU points for the development and test sets, respectively. Consequently, incorporating of NN LM technique to the n -gram-based SMT

³<http://www.speech.sri.com/projects/srilm/>

⁴We used a bootstrap resampling method as described in (Koehn, 2004) for the 95% confidence interval and 1 000 set resamples.

k	3-gram		4-gram		3- and 4-grams mixture	
	10	12	10	12	10	12
SRI LM	106.55	103.17	111.49	108.19	–	–
NN LM	97.75	93.14	98.11	92.32	94.89	89.73

Table 2: Perplexity reduction effect caused by the NN LM integration into the SMT system.

	Development corpus			Test corpus		
	BLEU	NIST	METEOR	BLEU	NIST	METEOR
baseline	27.00	7.29	53.83	21.77	6.65	49.89
$k=10$						
3-gram	27.43	7.27	53.82	21.91	6.61	49.62
4-gram	27.54	7.36	54.21	22.26	6.72	50.18
3- and 4-grams mixture	27.34	7.14	53.26	21.54	6.49	49.24
$k=12$						
3-gram	27.37	7.32	54.10	21.73	6.63	49.68
4-gram	27.47	7.31	53.98	21.91	6.65	49.82
3- and 4-grams mixture	27.47	7.35	54.33	22.07	6.71	50.20

Table 3: Comparative evaluation scores.

system allows gaining up 0.5 BLEU point for the development set and the same value for the test set. The results difference is statistically significant for two of six considered system configurations in terms of BLEU score, while METEOR and NIST metric values vary slightly, never exceeding the statistical significance thresholds.

5. Discussion and conclusions

NN LM shows very quite evident reduction in terms of perplexity (7.5 - 14.8 %), that allows to be beneficial to translation quality for Arabic to English translation task, even considering exclusively the most frequent words from the training corpus. The NN LM was introduced to the n -gram SMT system as a feature function and was used on the reranking step to rescore the N -best list of translation hypotheses.

Of the technique studied, we have found that system configuration providing the better BLEU score corresponds to the 4-gram LMs, while the technique of higher and lower order n -gram mixture seems to be promising and in the future we would like to apply this approach to other language pairs and to larger corpora. Surprisingly we have not achieved better system performance moving up the word frequency threshold from 10 to 12, that can be probably explained by high sparseness of the search space.

The idea of correlation of automatic evaluation metrics with the subjective human evaluation metrics assessed in translation quality is introduced in (Paul, 2006): fluency correlates better with BLEU and adequacy correlates best with METEOR, while the NIST metric has only moderate correlation to both subjective human evaluation metrics. Our work demonstrates the potential for NN LM application in the SMT to improve translation fluency, while adequacy keeps invariable.

Acknowledgements

Work partially supported by the Spanish Ministerio de Educación y Ciencia (TIN2006-12767), by the Spanish Government under grant TEC2006-13964-C03 (AVIVAVOZ project), and by the Spanish Government and FEDER under contract TIN2005-08660-C04-02 (EDECAN project).

6. References

- Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research*, 3(2):1137–1155.
- C. M. Bishop. 1995. *Neural networks for pattern recognition*. Oxford University Press.
- M. José Castro and F. Prat. 2003. New Directions in Connectionist Language Modeling. In *Computational Methods in Neural Modeling*, volume 2686 of *LNCS*, pages 598–605. Springer-Verlag.
- B. Chen, R. Cattoni, N. Bertoldi, M. Cettolo, and M. Federico. 2005. The ITC-irst SMT system for IWSLT-2005. In *Proceedings of IWSLT 2005*, page 98–104.
- J. M. Crego and J. B. Mariño. 2006. Improving statistical mt by coupling reordering and decoding. *Machine Translation*, 20(3):199–215.
- J. Crego, A. de Gispert, P. Lambert, M. Khalilov, M. Costajussà, J. Mariño, R. Banchs, and J.A.R. Fonollosa. 2006. The TALP Ngram-based SMT System for IWSLT 2006. In *Proceedings of IWSLT 2006*, pages 116–122.
- G. Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the ARPA Workshop on Human Language Technology (HLT)*, pages 138 – 145.
- S. España-Boquera, F. Zamora-Martínez, M.J. Castro-Bleda, and J. Gorbe-Moya. 2007. Efficient BP Algorithms for General Feedforward Neural Networks. In

- Bio-inspired Modeling of Cognitive Tasks*, volume 4527 of *LNCSS*, pages 327–336. Springer.
- N. Habash and F. Sadat. 2006. Arabic preprocessing schemes for statistical machine translation. In *Proceedings of the Human Language Technology Conference of the NAACL*, pages 49–52.
- P. Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP) 2004*, pages 388–395.
- J. B. Mariño, R. E. Banchs, J. M. Crego, A. de Gispert, P. Lambert, J. A. R. Fonollosa, and M. R. Costa-jussà. 2006. N-gram based machine translation. *Computational Linguistics*, 32(4):527–549, December.
- J.A. Nelder and R. Mead. 1965. A simplex method for function minimization. *The Computer organization*, 7:308–313.
- F. J. Och and H. Ney. 2000. Improved statistical alignment models. In *Proceedings of the 38th Ann. Meeting of the ACL*, pages 440 – 447, October.
- F. J. Och, D. Gildea, S. Khudanpur, A. Sarkar, K. Yamada, A. Fraser, S. Kumar, L. Shen, D. Smith, K. Eng, V. Jain, Z. Jin, and D. Radev. 2004. A Smorgasbord of Features for Statistical Machine Translation. In *Proceedings of HLTNAACL04*, pages 161–168.
- K. Papineni, S. Roukos, T. Ward, and W. Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the ACL 2002*, pages 311–318.
- M. Paul. 2006. Overview of the IWSLT 2006 Evaluation Campaign. In *Proceedings of IWSLT06*, pages 1–15.
- H. Schwenk, M. R. Costa-jussà, and J. A. R. Fonollosa. 2006a. Continuous space language models for the IWSLT 2006 task. In *Proceedings of IWSLT 2006*, pages 166–173.
- H. Schwenk, D. Déchelotte, and J. L. Gauvain. 2006b. Continuous space language models for statistical machine translation. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 723–730.
- A. Stolcke. 2002. SRILM: an extensible language modeling toolkit. In *Proceedings of the Int. Conf. on Spoken Language Processing*, pages 901–904.
- P. Xu and F. Jelinek. 2004. Random forest in language modeling. In *Proceedings of EMNLP 2004*, pages 325–332.