UCH-UPV English–Spanish system for WMT10

Francisco Zamora-Martínez
Dep. de Física, Matemáticas y Computación
Universidad CEU-Cardenal Herrera
Alfara del Patriarca (Valencia), Spain
fzamora@dsic.upv.es

Germán Sanchis-Trilles
Instituto Tecnológico de Informática
Universidad Politécnica de Valencia
Valencia, Spain
gsanchis@dsic.upv.es

Abstract

This paper describes the system developed in collabaration between UCH and UPV for the 2010 WMT. For this year's workshop, we present a system for English-Spanish translation. Output N-best lists were rescored via a target Neural Network Language Model, yielding improvements in the final translation quality as measured by BLEU and TER.

1 Introduction

In Statistical Machine Translation (SMT), the goal is to translate a sentence $\mathbf{f}$ from a given source language into an equivalent sentence $\hat{\mathbf{e}}$ from a certain target language. Such statement is typically formalised by means of the so-called log-linear models (Papineni et al., 1998; Och and Ney, 2002) as follows:

$$\hat{\mathbf{e}} = \underset{\mathbf{e}}{\operatorname{argmax}} \sum_{k=1}^{K} \lambda_k h_k(\mathbf{f}, \mathbf{e}) \qquad (1)$$

where $h_k(\mathbf{f}, \mathbf{e})$ is a score function representing an important feature for the translation of $\mathbf{f}$ into $\mathbf{e}$, K is the number of models (or features) and λ_k are the weights of the log-linear combination. Typically, the weights λ_k are optimised during the tuning stage with the use of a development set. Such features typically include the *target language model* $p(\mathbf{e})$, which is one of the core components of an SMT system. In fact, most of the times it is assigned a relatively high weight in the log-linear combination described above. Traditionally, language modelling techniques have been classified into two main groups, the first one including traditional grammars such as context-free grammars, and the second one comprising more statistical, corpus-based models, such as n-gram models. In order to assign a probability to a given word, such models rely on the assumption that such probability depends on the previous *history*, i.e. the $n-1$ preceding words in the utterance. Nowadays, n-gram models have become a "de facto" standard for language modelling in state-of-the-art SMT systems.

In the present work, we present a system which follows a coherent and natural evolution of probabilistic Language Models. Specifically, we propose the use of a continuous space language model trained in the form of a Neural Network Language Model (NN LM).

The use of continuous space representation of language has been successfully applied in recent NN approaches to language modelling (Bengio et al., 2003; Schwenk and Gauvain, 2002; Castro-Bleda and Prat, 2003; Schwenk et al., 2006). However, the use of Neural Network Language Models (NN LMs) (Bengio, 2008) in state-of-the-art SMT systems is not so popular. The only comprehensive work refers to (Schwenk, 2010), where the target LM is presented in the form of a fully-connected Multilayer Perceptron.

The presented system combines a standard, state-of-the-art SMT system with a NN LM via log-linear combination and N-best output rescoring. We chose to participate in the English-Spanish direction.

2 Neural Network Language Models

In SMT the most extended language models are n-grams (Bahl et al., 1983; Jelinek, 1997; Bahl et al., 1983). They compute the probability of each word given the context of the $n-1$ previous words:

$$p(s_1 \dots s_{|S|}) \approx \prod_{i=1}^{|S|} p(s_i | s_{i-n+1} \dots s_{i-1}) \,. \qquad (2)$$

where S is the sequence of words for which we want compute the probability, and $s_i \in S$, from a vocabulary Ω.

Proceedings of the Joint 5th Workshop on Statistical Machine Translation and MetricsMATR, pages 213–217,
Uppsala, Sweden, 11-16 July 2010.

A NN LM is a statistical LM which follows equation (2) as n-grams do, but where the probabilities that appear in that expression are estimated with a NN (Bengio et al., 2003; Castro-Bleda and Prat, 2003; Schwenk, 2007; Bengio, 2008). The model naturally fits under the probabilistic interpretation of the outputs of the NNs: if a NN, in this case a MLP, is trained as a classifier, the outputs associated to each class are estimations of the posterior probabilities of the defined classes (Bishop, 1995).

The training set for a LM is a sequence $s_1 s_2 \dots s_{|S|}$ 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 encoded. A natural representation is a local encoding following a "1-of-$|\Omega|$" scheme. The problem of this encoding for tasks with large vocabularies (as is typically the case) is the huge size of the resulting NN. We have solved this problem following the ideas of (Bengio et al., 2003; Schwenk, 2007), 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 $s_{i-n+1}, \dots, s_{i-1}$ of equation (2). Each word is represented using a local encoding.
- P is the projection layer of the input words, formed by $P_{i-n+1}, \dots, P_{i-1}$ subsets of projection units. The subset of projection units P_j represents the distributed encoding of input word s_j. The weights of this projection layer are linked, that is, the weights from each local encoding of input word s_j to the corresponding subset of projection units P_j are the same for all input words. After training, the codification layer is removed from the network by pre-computing a table of size $|\Omega|$ which serves as a distributed encoding.
- H denotes the hidden layer.
- The output layer O has $|\Omega|$ units, one for each word of the vocabulary.

This n-gram NN LM predicts the posterior probability of each word of the vocabulary given the $n-1$ previous words. A single forward pass of the MLP gives $p(\omega|s_{i-n+1} \dots s_{i-1})$ for every word $\omega \in \Omega$.

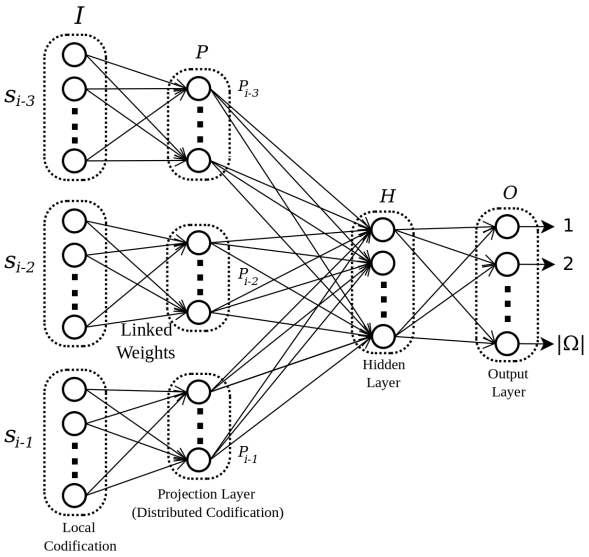

Figure 1: Architecture of the continuous space NN LM during training. The input words are $s_{i-n+1}, \dots, s_{i-1}$ (in this example, the input words are s_{i-3}, s_{i-2}, and s_{i-1} for a 4-gram). I, P, H, and O are the input, projection, hidden, and output layer, respectively, of the MLP.

The major advantage of the connectionist approach is the automatic smoothing performed by the neural network estimators. This smoothing is done via a continuous space representation of the input words. Learning the probability of n-grams, together with their representation in a continous space (Bengio et al., 2003), is an appropriate approximation for large vocabulary tasks. However, one of the drawbacks of such approach is the high computational cost entailed whenever the NN LM is computed directly, with no simplification whatsoever. For this reason, in this paper we will be restricting vocabulary size.

3 Experiments

3.1 Baseline system

For building the baseline SMT system, we used the open-source SMT toolkit Moses (Koehn et al., 2007), in its standard setup. The decoder includes a log-linear model comprising a phrase-based translation model, a language model, a lexicalised distortion model and word and phrase penalties. The weights of the log-linear interpolation were optimised by means of MERT (Och, 2003).

For the baseline LM, we computed a regular n-gram LM with Kneser-Ney smoothing (Kneser

and Ney, 1995) and interpolation by means of the SRILM (Stolcke, 2002) toolkit. Specifically, we trained a 6-gram LM on the larger Spanish corpora available (i.e. UN, News-Shuffled and Europarl), and a 5-gram LM on the News-Commentary corpus. Once these LMs had been built, they were finally interpolated so as to maximise the perplexity of the News-Commentary test set of the 2008 shared task. This was done so according to preliminary investigation.

3.2 NN LM system architecture

The presented systems follow previous works of (Schwenk et al., 2006; Khalilov et al., 2008; Schwenk and Koehn, 2008; Schwenk, 2010) where the use of a NN LM helps achieving better performance in the final system.

The NN LM was incorporated to the baseline system via log-linear combination, adding a new feature to the output N-best list generated by the baseline system (in this case $N = 1\,000$). Specifically, the NN LM was used to compute the log-probability of each sentence within the N-best list. Then, the scores of such list were extended with our new, NN LM-based feature. This being done, we optimised the coefficients of the log-linear interpolation by means of MERT, taking into account the newly introduced feature. Finally the list was re-scored and the best hypothesis was extracted and returned as final output. Figure 2 shows a diagram of the system structure.

3.3 Experimental setup and results

NN LM was trained with the concatenation of the News-shuffled and News-Commentary10 Spanish corpora. Other language resources were discarded due to the large amount of computational resources that would have been needed for training a NN LM with such material. Table 1 shows some statistics of the corpora. In order to reduce the complexity of the model, the vocabulary was restricted to the 20K more frequent words in the concatenation of news corpora. Using this restricted vocabulary implies that 6.4% of the running words of the news-test2008 set, and 7.3% of the running words within the official 2010 test set, will be considered as unknown for our system. In addition, the vocabulary includes a special token for unknown words used for compute probabilities when an unknown word appears, as described in Equation 2.

Table 1: Spanish corpora statistics. NC stands for News-Commentary and UN for United Nations, while $|\Omega|$ stands for vocabulary size, and M/K for millions/thousands of elements.

Set	# Lines	# Words	$\|\Omega\|$
NC	108K	2.96M	67K
News-Shuffled	3.86M	107M	512K
Europarl	1.82M	51M	172K
UN	6.22M	214M	411K
Total	3.96M	110M	521K

A 6-gram NN LM was trained for this task, based in previous works (Khalilov et al., 2008). The distributed encoding input layer consists of 640 units (128 for each word), the hidden layer has 500 units, and the output layer has 20K units, one for each word in the restricted vocabulary. The total number of weights in the network was 10 342 003. The training procedure was conducted by means of the stochastic back-propagation algorithm with weight decay, with a replacement of 300K training samples and 200K validation samples in each training epoch. The training and validation sets were randomly extracted from the concatenation of news corpora. The training set consisted of 102M words (3M sentences) and validation set 8M words (300K sentences). The network needed 129 epochs for achieving convergence, resulting in 38.7M and 25.8M training and validation samples respectively. For training the NN LM we used the April toolkit (España-Boquera et al., 2007; Zamora-Martínez et al., 2009), which implements a pattern recognition and neural networks toolkit. The perplexity achieved by the 6-gram NN LM in the Spanish news-test08 development set was 116, versus 94 obtained with a standard 6-gram language model with interpolation and Kneser-Ney smoothing (Kneser and Ney, 1995).

The number of sentences in the N-best list was set to 1 000 unique output sentences. Results can be seen in Table 2. In order to assess the reliability of such results, we computed pairwise improvement intervals as described in (Koehn, 2004), by means of bootstrapping with 1000 bootstrap iterations and at a 95% confidence level. Such confidence test reported the improvements to be statistically significant.

Four more experiments have done in order to study the influence of the N-best list size in the

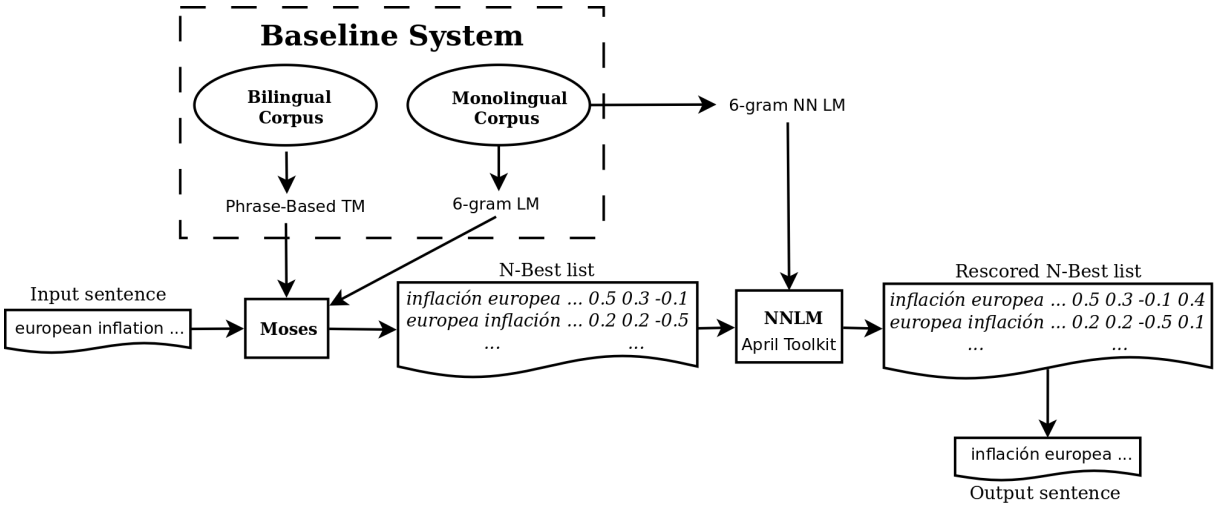

Figure 2: Architecture of the system.

Table 2: English-Spanish translation quality for development and official test set. Results are given in BLEU/TER.

	test08 (dev)	test10 (test)
Baseline	24.8/60.0	26.7/55.1
NN LM	25.2/59.6	27.8/54.0

Table 3: Test set BLEU/TER performance for each N-best list size.

N-best list size	BLEU	TER
200	27.5	54.2
400	27.6	54.2
600	27.7	54.1
800	27.6	54.2
1000	27.8	54.0

performance achieved by the NN LM rescoring. For each N-best list size (200, 400, 600 and 800) the weights of the log-linear interpolation were optimised by means of MERT over the test08 set. Table 3 shows the test results for each N-best list size using the correspondent optimised weights. As it can be seen, the size of the N-best list seems to have an impact on the final translation quality produced. Although in this case the results are not statistically significant for each size step, the final difference (from 27.5 to 27.8) is already significant.

4 Conclusions

In this paper, an improved SMT system by using a NN LM was presented. Specifically, it has been shown that the final translation quality, as measured by BLEU and TER, is improved over the quality obtained with a state-of-the-art SMT system. Such improvements, of 1.1 BLEU points, were found to be statistically significant. The system presented uses a neural network only for computing the language model probabilities. As an immediate future work, we intend to compute the language model by means of a linear interpolation of several neural networks. Another interesting idea is to integrate the NN LM within the decoder itself, instead of performing a subsequent rescoring step. This can be done extending the ideas presented in a previous work (Zamora-Martínez et al., 2009), in which the evaluation of NN LM is significantly sped-up.

Acknowledgments

This paper was partially supported by the EC (FEDER/FSE) and by the Spanish Government (MICINN and MITyC) under the MIPRCV "Consolider Ingenio 2010" program (CSD2007-00018), iTrans2 (TIN2009-14511) project and the erudito.com (TSI-020110-2009-439) project.

References

L. R. Bahl, F. Jelinek, and R. L. Mercer. 1983. A Maximum Likelihood Approach to Continuous Speech Recognition. *IEEE Trans. on Pat. Anal. and Mach. Intel.*, 5(2):179–190.

Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin. 2003. A Neural Probabilistic Language Model. *Journal of Machine Learning Research*, 3(2):1137–1155.

Y. Bengio. 2008. Neural net language models. *Scholarpedia*, 3(1):3881.

C. M. Bishop. 1995. *Neural networks for pattern recognition*. Oxford University Press.

M.J. Castro-Bleda 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.

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 *LNCS*, pages 327–336. Springer.

F. Jelinek. 1997. *Statistical Methods for Speech Recognition*. Language, Speech, and Communication. The MIT Press.

M. Khalilov, J. A. R. Fonollosa, F. Zamora-Martínez, M. J. Castro-Bleda, and S. España-Boquera. 2008. Neural network language models for translation with limited data. In *20th International Conference on Tools with Artificial Intelligence, ICTAI'08*, pages 445–451, november.

R. Kneser and H. Ney. 1995. Improved backing-off for m-gram language modeling. *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, II:181–184, May.

P. Koehn et al. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proc. of the ACL Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic.

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

F. Och and H. Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proc. of ACL'02*, pages 295–302.

F.J. Och. 2003. Minimum Error Rate Training in Statistical Machine Translation. In *Proc. of ACL*, pages 160–167, Sapporo, Japan.

K. Papineni, S. Roukos, and T. Ward. 1998. Maximum likelihood and discriminative training of direct translation models. In *Proc. of ICASSP*, pages 189–192.

H. Schwenk and J. L. Gauvain. 2002. Connectionist language modeling for large vocabulary continuous speech recognition. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP'02)*, pages 765–768, Orlando, Florida (USA), May.

H. Schwenk and P. Koehn. 2008. Large and diverse language models for statistical machine translation. In *International Joint Conference on Natural Language Processing*, pages 661–668.

H. Schwenk, D. Déchelotte, and J. L. Gauvain. 2006. Continuous space language models for statistical machine translation. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 723–730.

H. Schwenk. 2007. Continuous space language models. *Computer Speech and Language*, 21(3):492–518.

H. Schwenk. 2010. Continuous space language models for statistical machine translation. *The Prague Bulletin of Mathematical Linguistics*, 93.

A. Stolcke. 2002. SRILM – an extensible language modeling toolkit. In *Proc. of ICSLP'02*, pages 901–904, September.

F. Zamora-Martínez, M.J. Castro-Bleda, and S. España-Boquera. 2009. Fast Evaluation of Connectionist Language Models. In *International Work-Conference on Artificial Neural Networks*, volume 5517 of *LNCS*, pages 33–40. Springer.