iberSPEECH2014

November 19-21 2014, Las Palmas de Gran Canaria

CONFERENCE PROCEEDINGS

iberSPEECH 2014

VIII Jornadas en Tecnologías del Habla

and IV Iberian SLTech Workshop

Escuela de Ingeniería en Telecomunicación y Electrónica
Universidad de Las Palmas de Gran Canaria
SPAIN

Organized by:

IDeTIC
Instituto para el Desarrollo Tecnológico y la Innovación en Comunicaciones

División de Procesado Digital de Señal

RTTH

Spanish Thematic Network on Speech Technology (RTTH)

Supported by:

UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA

ULPGC 25 ANIVERSARIO 1989-2014

Selected papers published by:

Springer
Lecture Notes in Artificial Intelligence
Lecture Notes in Computer Science

LNAI Vol. 8854

PROCEEDINGS

VII Jornadas en Tecnología del Habla

and

III Iberian SLTech Workshop

November 19-21, 2014
Escuela de Ingeniería enTelecomunicación y Electrónica
Universidad de Las Palmas de Gran Canaria
Las Palmas de Gran Canaria, Spain

Editors: Juan Luis Navarro Mesa
Pedro Quintana Morales
Alfonso Ortega
António Teixeira
Eduardo Hernández Pérez
Antonio Ravelo García
Iván Guerra Moreno
Manuel Medina Molina
Sofía Martín González

Organized by:

IDeTIC

División de Procesado Digital de Señal

RTTH

Spanish Thematic Network on Speech Technology (RTTH)

Supported by:

UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA

ULPGC 25 ANIVERSARIO 1989-2014

ISBN: 978-84-617-2862-6

http://iberspeech2014.ulpgc.es

First steps towards Skipping NNLMs

A. Palacios-Corella[1], F. Zamora-Martínez[2], S. España-Boquera[1], and M.J. Castro-Bleda[1]

[1] Departamento de Sistemas Informáticos y Computación, Universitat Politècnica de València, Valencia, Spain
[2] Departamento de Ciencias Físicas, Matemáticas y de la Computación, Universidad CEU Cardenal Herrera, Alfara del Patriarca (Valencia), Spain

Abstract. Statistical language modeling greatly suffers from the effects of data sparsity which is tackled by means of smoothing techniques. Continuous space language models are able to interpolate unseen word histories but new problems and challenges arise, as a very high computational cost during evaluation of N-gram probabilities, due to the softmax normalization constant. Several approaches to study how to reduce this computational cost have been proposed in the literature. This work tries to improve the use of pre-computed softmax normalization constants tables by including the Skipping N-grams technique into Neural Network Language Models (NN LMs) and describes some experiments conducted on IAM-DB corpus to validate the viability of the proposed technique. The skipping for NN LMs works as regularization, but additionally helps to simplify the use of pre-computation of softmax normalization constants, as will be shown in the preliminary experiments of this paper.

1 Introduction

The estimation of a-priori probabilities of word sequences is one of the main purposes of language models (LMs). They play a key role in many natural language processing applications such as speech or text recognition, machine translation, part-of-speech tagging, or information retrieval. The most commonly used LMs are statistical N-gram models [6, 4], which only consider the $N-1$ previous words to estimate the LM probability for a sequence of words of length $|W|$:

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

The use of N-grams is usually restricted, in practice, to low orders, as is the case of trigrams. Although trigram LMs work well in practice, there are many improvements over this simple model, including higher-order N-grams, smoothing techniques, skipping models, clustering models or cache models [4].

The estimation of these models are based on counting occurrences of word histories in a training corpus. A more recent alternative to the classical "count-based" N-grams are those based on a continuous representation of the lexicon

using connectionist approaches, as is the case of Neural Network Language Models (NN LMs) based on multilayer perceptrons [15, 18, 19].

NN LMs do not require the use of explicit smoothing techniques usually employed in count-based N-grams (e.g., backing-off), but important computational issues appear when using large vocabularies, majorly due to output softmax activation function. Short-list [15] and pre-computation of softmax normalization constants [18, 19] allow to reduce significantly this computational cost.

This paper describes the first steps of this research and its main contribution is to study whether the connectionist skipping N-gram LMs can help to improve NN LMs performance as a regularization method and as a new technique for smoothed NN LMs presented in [18].

2 Related language models

2.1 Skipping N-gram LMs

N-grams suffer from data sparsity making it necessary the use of smoothing techniques. Skipping N-grams [5, 14, 13, 10, 16, 4] can improve the generalization ability of standard smoothing techniques. The idea is that the exact context will have not probably been seen during training, but the chance of having seen a similar context (with gaps that are skipped over) increases as the order of the N-gram does.

Let us explain the idea with an example: suppose that the sentence *"The little boy likes pizza very much"* appears in the training data and we are trying to estimate a 5-gram. The training sentence has contributed to the estimation of p(pizza|the little boy likes). Unfortunately, this sentence cannot help much in the estimation of p(pizza|the little girl likes). The usual technique of backing-off would consist of using lower order N-grams. In this case, we would need to descend until p(pizza|likes). By skipping some words from the past history, the training sentence is useful to estimate p(pizza|the little — likes). For instance, the probability of p(pizza|the little boy eats) would benefit from p(pizza|the little boy —) whereas backing-off would require to descent until unigrams as far as the example sentence is concerned.

Skipping N-grams are not only based on skipping words from the past history but also on the combination of different ways of performing these skips. Each different way of skipping words can be considered a lower order LM in some way and the Skipping N-grams are a mixture of them. For example, the representation of several skipped trigrams (at most two context words are not skipped) may approximate a higher order N-gram using less resources, which explains why some authors have considered Skipping N-grams as a *poor man's* alternative to higher order N-grams. Nevertheless, our emphasis here is that they can also be useful to improve NN LMs probability computation.

2.2 NNLMs

NN LMs are able to learn a continuous representation of the lexicon [15, 18, 19]. A scheme of a NN LM is illustrated in Figure 1 where a multilayer perceptron

(MLP) is used to estimate $p(w_i|w_{i-n+1}\dots w_{i-1})$. There is an output neuron for each word w_i in a vocabulary Ω', a subset of the most frequent words of the task vocabulary Ω, allowing to increase computation of output layer.[3] The input of the NN LM is composed of the sequence $w_{i-n+1},\dots,w_{i-1}$. A local encoding scheme would be a natural representation of the input words, but it is not suitable for large vocabulary tasks due to the huge size of the resulting Neural Network (NN). To overcome this problem, a distributed representation for each word is learned by means of a projection layer. The mapping is learned by backpropagation in the same way as the other weights in the NN. After the projection layer, a hidden layer with non-linear activation function is used and an output layer with the softmax function will represent the N-gram LM probability distribution. The projection layer can be removed from the network after training, since it is much more efficient to replace it by a pre-computed table which stores the distributed encoding of each word. In order to alleviate problems with rare words, the input is restricted to words with frequency greater than a threshold K in training data.

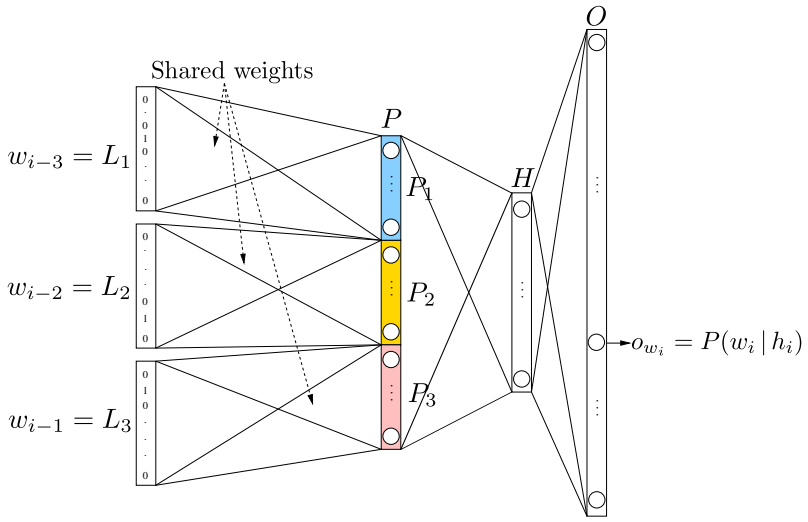

Fig. 1: The architecture of a 4-gram NN LM during training. In this example, the history of word w_i is represented by $h_i = w_{i-1,}, w_{i-2}, w_{i-3}$.

Pre-computation of softmax normalization constants [18] has been proposed as a practical solution to the high computational cost of NN LMs output layer. Following our previous work, in order to compute 4-gram probabilities by using

[3] Relating to the output layer, several techniques exist to reduce its computational cost [8, 15, 18, 2, 12]. Short-list approach [15] is the one used in this work and requires to combine the NN LMs with another LM for words out of the short-list.

NN LMs, lower order NN LMs are needed to compute N-gram probabilities for contexts which softmax normalization constant has not been pre-computed. In this way, the model performance decreases but a significant time speed-up is obtained. One drawback of this approach is that it forces the training of several NN LMs for different N-gram orders.

3 Skipping NNLMs

Since skipping techniques consist of learning several LMs based on different ways of removing words from the past history, a natural way of introducing Skipping NN LMs is by training a sole MLP but replacing random positions in context history by a new special symbol $\langle \text{NONE} \rangle$. There exist 2^{N-1} different ways of replacing some of the $N-1$ words by $\langle \text{NONE} \rangle$. After training, 2^{N-1} models are available, each one with a fixed pattern of skipped positions. The perplexity (PPL) of each possible skipping model will be evaluated in this paper. Additionally, skipping the proper positions, it is possible to convert a 5-gram NN LM into a 4-gram, 3-gram and so on. Softmax normalization constants can be pre-computed for this unique model using LM training data. When a constant is not found, instead of using a totally different model with lower order, it is possible to use the same model but with skipping positions which reduce its order. We have evaluated this approach, a comparison of PPL results of skipped lower order models and *true* lower order NN LMs will be performed in this paper.

It is worth mentioning that the technique described in [11], also coined as continuous skip-ngram model by their authors, is not related to this work despite the similarity in nomenclature: the main purpose of [11] consists in learning a continuous representation of the lexicon of a task in a way that captures the semantic similarity w.r.t. the task. This representation could be used, for instance, in NN LMs. Their continuous skip-ngram models are log-bilinear classifiers which receive a word at the input layer and try to predict the neighboring (past and future) words.

4 Experimental setup

The NN LMs used as baseline for our experiments are fully described in [18, 19]. These models will be extended to construct the Skipping NN LM as described in the previous section. The experiment is conducted on a task based on the transcriptions of the IAM-DB [9] handwritten dataset. The training material to estimate the LMs is taken from LOB [7], Wellington [1] and Brown [3] corpora, which add up to 402K sentences in total. Validation and test sets were taken from IAM-DB corpus (920 and 2781 lines, respectively). Since IAM-DB texts are extracted from LOB corpus, IAM-DB validation and test sentences have been removed from LOB text data. PPL results will be presented for validation and test parts of the IAM-DB.

Backprogation algorithm and L2 regularized cross-entropy loss function are used to train NN LMs and Skipping NN LMs. A projection layer with 256

neurons and a hidden layer with 200 neurons has been used based in previous works. For Skipping NN LMs, the input is stochastically perturbed, introducing the ⟨NONE⟩ symbol in zero or more random positions of the input layer. For every training pattern, the number of skips is sampled from a multinomial distribution. Given a number of skips, the positions of them are uniformly distributed. The multinomial distribution is defined to assign a probability of 50% for no skips at all, whereas the other 50% is distributed over one, two or more skips following a hyperbolic trend (see Table 1).

Table 1: The number of skips is sampled from the following multinomial distributions, given N-gram order.

# skips	bigram	3-gram	4-gram	5-gram
0	50%	50%	50%	50%
1	50%	33%	27%	24%
2	–	16%	13%	12%
3	–	–	9%	8%
4	–	–	–	6%

Once the model is trained, 2^{N-1} different LMs can be build, having each one a different combination of skips or ⟨NONE⟩ tokens in its input layer. Although one of the main benefits of Skipping NN LMs is to emulate lower order N-grams in order to greatly simplify the speed-up technique based on memorizing softmax normalization constants, in this work we focus on: first, investigating these emulation capabilities, and, second, computing the probabilities of the first $N1$ words of a sentence (for higher order N-grams) before the complete word history is available.

5 Experimental results

First, an evaluation of the PPL trend for validation data depending on the none tokens has been performed (see Figure 2). PPL is computed for all of the 2^{N-1} possible skipping positions. To better interpret the results, let us remark that the skipping number is a right-to-left bits mask where "0" indicates no skip and "1" indicates a skip. So, the skip number 7 in a 5-gram refers to the binary representation 1110, meaning that probability of N-gram at position i is computed by using word $i-1$ and three ⟨NONE⟩ tokens. The complete Skipping NN LMs models consider the combination of the different LMs associated to the skipping masks. We have performed this combination, for each N-gram order, using the `compute-best-mix` tool from SRILM toolkit [17]. The obtained results are very similar and competitive with those of standard NN LMs but they are not able to outperform this baseline.

As can be observed in Figure 2, the Skipping NN LMs with a skipping mask without ⟨NONE⟩ performs as well as standard NN LMs while the presence of

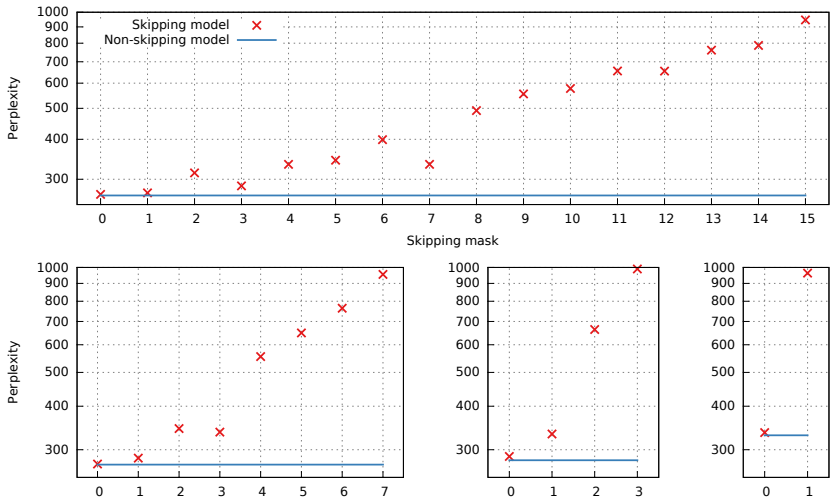

Fig. 2: PPL measured on validation set for Skipping NN LMs varying the skipping mask together with the baseline PPL of the non-skipping NN LM. The vertical axes of the four plots show the PPL and the skipping mask appears on the horizontal axes using the binary notation described in the paper. Upper plot corresponds to 5-grams. Lower plots correspond, from left to right, to 4-grams, trigrams and bigrams.

⟨NONE⟩ downgrades the results and this effect is more pronounced when ⟨NONE⟩ is close to the word to be predicted.

In order to evaluate the future ability of Skipping NN LMs to simplify the work presented at [18], Table 2 shows the PPL obtained by using no skipping NN LMs and Skipping NN LMs where the skipping positions are set to model equal or lower orders. Values from the first row of each table are the PPL values for each one of the NN LMs. The remaining rows contain the values from each of the Skipping NN LMs. The last non-void value of these rows is the PPL for the set without perturbing its input. The previous values are obtained after using skipping configurations that emulate a lower order ngram. For example, the 4-gram value for 5-gram Skipping NN LM is obtained after using the skipping mask that replaces the furthest word from the context by ⟨NONE⟩. To get the trigram value we replace the next word from the context too, and so on.

We can observe that the column values for each table are quite similar. This means that the PPL values for each one of the NN LMs are similar to the ones obtained for the Skipping NN LMs which can compute them using the adequate skipping configuration. Therefore, it is possible to train a single Skipping NN LM to imitate the behaviour of several NN LMs. Backing off for Skipping NN LMs has a simpler training procedure, you need only one model, while for NN LMs you need one model for every N-gram order [18].

Table 2: PPL results for IAM-DB validation (left) and test (right). For skip models the PPL has been computed for 0 skips and for skip combinations which simulate a lower order model.

Model	Ngram order				
	1	2	3	4	5
5gr No skip	–	–	–	–	267
4gr No skip	–	–	–	272	–
3gr No skip	–	–	280	–	–
2gr No skip	–	330	–	–	–
5gr skip	946	334	286	272	269
4gr skip	955	337	284	273	–
3gr skip	990	333	287	–	–
2gr skip	963	336	–	–	–

Model	Ngram order				
	1	2	3	4	5
5gr No skip	–	–	–	–	309
4gr No skip	–	–	–	313	–
3gr No skip	–	–	319	–	–
2gr No skip	–	376	–	–	–
5gr skip	1019	378	326	311	309
4gr skip	1025	383	324	313	–
3gr skip	1045	377	327	–	–
2gr skip	1026	381	–	–	–

6 Conclusions and future work

This work is, to the best of our knowledge, the very first attempt to integrate the well known technique of Skipping N-grams into NN LMs. NN LMs are capable of learning distributed representations which might explain that non additional gain is obtained by including the skipping technique. Coming back to the example of *the little boy* which *likes pizza*, from Section 2.1, it is possible that in a large corpus the contexts of words *boy* and *girl* are similar enough to make it possible for the MLP to learn a similar representation for these words so that the effect of skipping is diminished.

On the other side, the capability of Skipping NN LMs to emulate lower order NN LMs makes them very suitable for greatly simplifying the speed-up technique based on pre-computation of softmax normalization constants [18] since these models rely on lower order models when a constant is not found.

As a future work, we plan to investigate the effect of this technique in larger corpora to give more support to the preliminary results presented here and to study the effect of the new LM in the overall error of a recognition system.

Acknowledgments

This work has been partially supported by the Spanish Government TIN2010-18958.

References

1. Bauer, L.: Manual of Information to Accompany The Wellington Corpus of Written New Zealand English. Tech. rep., Department of Linguistics, Victoria University, Wellington, New Zealand (1993)

2. Bengio, Y., Senecal, J.S.: Adaptive importance sampling to accelerate training of a neural probabilistic language model. IEEE Transactions on Neural Networks 19(4), 713–722 (2008)
3. Francis, W., Kucera, H.: Brown Corpus Manual, Manual of Information to accompany A Standard Corpus of Present-Day Edited American English. Tech. rep., Department of Linguistics, Brown University, Providence, Rhode Island, US (1979)
4. Goodman, J.T.: A Bit of Progress in Language Modeling - Extended Version. Tech. Rep. MSR-TR-2001-72, Microsoft Research, One Microsoft Way Redmond, WA 98052 (2001)
5. Huang, X., Alleva, F., Hon, H.W., Hwang, M.Y., Lee, K.F., Rosenfeld, R.: The SPHINX-II speech recognition system: an overview. Computer Speech and Language 7(2), 137–148 (1993)
6. Jelinek, F.: Statistical Methods for Speech Recognition. Language, Speech, and Communication, The MIT Press (1997)
7. Johansson, S., Atwell, E., Garside, R., Leech, G.: The Tagged LOB Corpus: User's Manual. Tech. rep., Norwegian Computing Centre for the Humanities, Bergen, Norway (1986)
8. Le-Hai, S., Oparin, I., Alexandre, A., Gauvaing, J.L., Franois, Y.: Structured Output Layer Neural Network Language Model. In: Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). vol. 11, pp. 5524–5527 (2011)
9. Marti, U.V., Bunke, H.: The IAM-database: an English sentence database for offline handwriting recognition. International Journal of Document Analysis and Recognition 5, 39–46 (2002)
10. Martin, S., Hamacher, C., Liermann, J., Wessel, F., Ney, H.: Assessment of smoothing methods and complex stochastic language modeling. In: Proc. 6th European Conference on Speech Communications and Technology (Eurospeech). vol. 5, pp. 1939–1942 (1999)
11. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR abs/1301.3781 (2013)
12. Mnih, A., Kavukcuoglu, K.: Learning word embeddings efficiently with noise-contrastive estimation. In: Advances in Neural Information Processing Systems 26, pp. 2265–2273 (2013)
13. Ney, H., Essen, U., Kneser, R.: On structuring probabilistic dependences in stochastic language modeling. Computer Speech and Language 8(4), 1–38 (1994)
14. Rosenfeld, R.: Adaptative statistical language modeling: A maximum entropy approach. Ph.D. thesis, Carnegie Mellon University (1994)
15. Schwenk, H.: Continuous space language models. Computer Speech and Language 21(3), 492–518 (2007)
16. Siu, M., Ostendorf, M.: Variable n-grams and extensions for conversational speech language modeling. IEEE Trans. Speech and Audio Processsing 8(1), 63–75 (2000)
17. Stolcke, A.: SRILM: an extensible language modeling toolkit. In: Proceedings of the International Conference on Spoken Language Processing (ICSLP). pp. 901–904 (2002)
18. Zamora-Martínez, F., Castro-Bleda, M., España-Boquera, S.: Fast Evaluation of Connectionist Language Models. In: International Work-Conference on Artificial Neural Networks, LNCS, vol. 5517, pp. 33–40. Springer (2009)
19. Zamora-Martínez, F., Frinken, V., España-Boquera, S., Castro-Bleda, M., Fischer, A., Bunke, H.: Neural network language models for off-line handwriting recognition. Pattern Recognition 47(4), 1642 – 1652 (2014)