Dynamic Extension of ASR Lexicon Using Wikipedia Data

Badr Abdullah, Irina Illina, Dominique Fohr

DYNAMIC EXTENSION OF ASR LEXICON USING WIKIPEDIA DATA

Badr Abdullah[1,2,3], Irina Illina[1,2,3], Dominique Fohr[1,2,3]
[1] Université de Lorraine, LORIA, UMR 7503, Vandoeuvre-lès-Nancy, F-54506, France
[2] Inria, Villers-les-Nancy, F-54600, France
[3] CNRS, LORIA, UMR 7503, Vandoeuvre-les-Nancy, F-54506, France

ABSTRACT

Despite recent progress in developing Large Vocabulary Continuous Speech Recognition Systems (LVCSR), these systems suffer from Out-Of-Vocabulary words (OOV). In many cases, the OOV words are Proper Nouns (PNs). The correct recognition of PNs is essential for broadcast news, audio indexing, etc. In this article, we address the problem of OOV PN retrieval in the framework of broadcast news LVCSR. We focused on dynamic (document dependent) extension of LVCSR lexicon. To retrieve relevant OOV PNs, we propose to use a very large multipurpose text corpus: Wikipedia. This corpus contains a huge number of PNs. These PNs are grouped in semantically similar classes using word embedding. We use a two-step approach: first, we select OOV PN pertinent classes with a multi-class Deep Neural Network (DNN). Secondly, we rank the OOVs of the selected classes. The experiments on French broadcast news show that the Bi-GRU model outperforms other studied models. Speech recognition experiments demonstrate the effectiveness of the proposed methodology.

Index Terms— Automatic speech recognition, out-of-vocabulary words, word embedding, lexicon extension

1. INTRODUCTION

Automatic speech recognition (ASR) systems usually operate on a fixed lexicon and cannot correctly transcribe words that are not in the lexicon. For a system that must recognize speech about current events, this can pose a problem, because new words are continually introduced based on the occurring events. A particular issue is proper nouns (PNs): the names of important people or locations. Recognizing them can be vital to understand the occurring events. Context-independent extension of the LVCSR lexicon based on word frequency statistics does not necessarily improve the LVCSR performance because missing OOV PNs can be rare words and thus are difficult to retrieve.

Methods for OOV retrieval can be categorized in OOV detection approaches and vocabulary selection approaches. Detection based approaches [1], [2] aim to detect/locate OOV words in the LVCSR hypothesis, followed by a search for the matching OOV word. They need careful selection of sub-word units and LM estimation. Recently, in the acoustic-to-word model based on the Connectionist Temporal Classification (CTC) of [3], OOV words are taken into account by character-based CTC. Vocabulary selection based approaches propose a relevant vocabulary for speech recognition based on additional text data [4]. These methods are more dynamic as they propose context specific vocabulary [5], [6], [7]. In the present paper, we study this last class methodology. In our approach, our aim is not to identify OOV regions in the ASR hypothesis but to use the global context of the test document.

In this article, we propose a **document-specific multi-class** approach to **dynamically extend** the LVCSR lexicon, based on the semantic context of the document. For each test document, we will dynamically generate an *extended* lexicon, containing all words of the original lexicon and a list of highly relevant OOV PNs. To retrieve OOV PNs we propose to use a very large text corpus: Wikipedia articles, covering a very large number of topics and containing a huge number of proper names. We call this a *multipurpose* corpus. This corpus is employed to learn a context model that captures relationships between a document and the OOV PNs of the document. As a context model, we propose multi-class DNN approaches. These approaches are capable of learning task specific words and context representations. The retrieved OOV PNs are added to the original lexicon and a second-pass decoding is performed.

In previous works, different methods for OOV PN retrieval have been explored. *Latent Dirichlet Allocation* (LDA) [6] showed that generated unsupervised context representations are not optimal for our task. The Neural Bag-of-Words (NBOW) model [9] and the Neural Bag-of-Weighted-Words (NBOW2) model [10] give very good results. However, in these works, the diachronic text corpus of medium size was used. Moreover, the diachronic text corpus was close in term of topics and date to the test corpus. This is not always possible. Some systems can be deployed in a very large number of tasks. Compared to our previous works [11][12][10], the main contributions of this paper are: (a) the use of a very large multi-topic text corpus to retrieve OOV PNs; (b) the use of multi-class DNN approaches to represent semantic context (CNN and Bi-GRU); (c) the comparison of the proposed models with the *fastText* approach; (d) the evaluation of the proposed approach in terms of PN error rate on audio files. The advantage of our approach is that the LVCSR lexicon is dynamically adapted to each spoken document without

sretraining the other components of the system. The proposed approach can be performed for any language.

The rest of the paper is organized as follows. Section 2 presents the background and describes the proposed approaches. The experiment protocol and the experimental data are described in section 3. The OOV PN retrieval results are discussed in section 4.

2. PROPOSED METHODOLOGY

The general procedure for retrieving and adding OOV proper nouns to the ASR lexicon is as follows (see Figure 1):

(1) Context model is learned from the multipurpose text corpus.

(2) Given a spoken document to be recognized, the ASR is run to obtain an automatic transcription of the document (first-pass hypothesis).

(3) Based on the semantic context of the first-pass hypothesis, a list of likely relevant OOV PNs is produced. This is performed using the context model and our retrieval methodology.

(4) A grapheme-to-phoneme (G2P) conversion is applied to get pronunciations for each word in the proposed list of OOV PNs. OOV PNs and their generated pronunciations are added to the lexicon. The new lexicon is called the *extended lexicon*. Probabilities for the new OOV PNs are added to the language model.

(5) A second-pass ASR decoding is performed using the extended lexicon.

The estimation of the language model probabilities and the generation of word pronunciations for OOV PNs are two large research problems. They are beyond the scope of this article.

For training the context model and retrieving OOV PNs, we propose to use Wikipedia articles: a *multipurpose* corpus. This corpus has several advantages: it has large lexical coverage and a huge number of topics. An interesting feature of Wikipedia is the huge number of proper nouns. There are approximately *1.7* million OOV PNs in the French Wikipedia (see section 3). Extracting a short (compared to the usual size of a lexicon *100K-200K*) list of relevant OOV PNs from *1.7*M possible candidates is not trivial. In this article, we address this problem by partitioning OOV PNs from Wikipedia in classes according to their semantic context. We propose a two-step approach:

(1) Partition OOV PNs in classes and learn the context model in term of OOV PN classes using the multipurpose corpus. Using a semantic model, select the OOV PN classes that could occur in the semantic context of the document to be recognized (first-pass hypothesis).

(2) Rank the OOV PNs of the selected classes based on their relevance to the document to be recognized.

In the following, we will present these two steps.

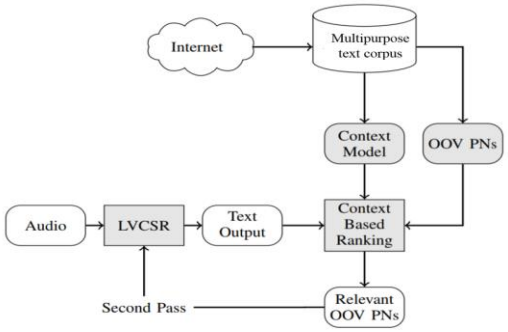

Figure 1. Illustration of our approach for recognition OOV PNs [11].

2.1. Context model and OOV PN class prediction

In order to group OOV PNs from Wikipedia into classes, we propose to apply *K*-means clustering. K-means is a popular clustering algorithm that aims to partition data points into *K* clusters. In our problem, the data points are OOV PN word vectors in some space.

As word vector space, we propose to exploit continuous-space representation proposed in [13][14], *word2vec*. This representation takes into account semantic information of each word. Semantically similar words will be mapped to the same area of the vector space. We apply *K*-means clustering on the word embeddings space to obtain classes. A single Wikipedia document can cover a mixture of topics, and so might contain OOV words from different classes. Mapping a document to one or more classes is a **multi-class** classification problem.

After OOV PN class creation, the context model based on DNN is built. The training of this DNN requires a large text corpus containing a huge number of proper names. In this article we use Wikipedia data. The **input of the DNN** network consists of the **in-vocabulary words** of the document, the output is the class labels of the OOV PNs present in this document. The size of the output layer is equal to the number of classes. Note that the frequency of an OOV class in a document is not taken into account, only its presence. We used two DNN models: CNN and Bi-GRU.

The first layer of the DNN is an embedding layer: the weight of this layer can be initialized with word embedding computed by *word2vec*. For multi-class multi-label classification, we chose sigmoid as the activation of the output layer and binary cross-entropy as the loss function.

The trained context model is used to predict the OOV PN classes that are likely to occur in the ASR transcription. Since we formulate our task as a multi-class classification problem, the top-Q classes that have the highest output values are considered as predicted classes.

2.1.1. Convolutional Neural Network

CNN has been successfully applied for text classification [15]. Since language is temporal in nature, a one-

dimensional convolutional operation can be applied to extract meaningful features from word sequence.

Our network has the following configuration. The input is the sequence of words, represented as one-hot vectors. The first layer is an embedding layer, the second is a convolutional layer and the third layer is a max-over-time pooling layer. The last layer is a dense (fully-connected) layer with a sigmoid as activation function. The last layer has as many neurons as OOV PN classes.

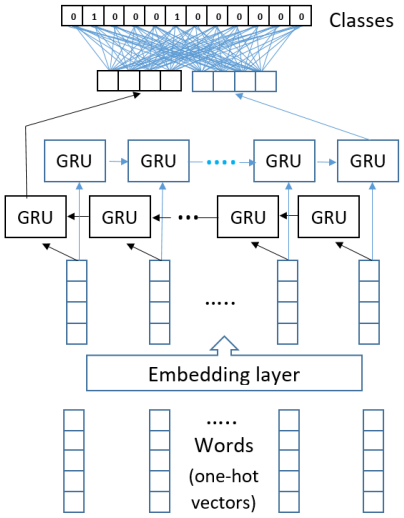

Figure 2. The Bi-GRU model.

2.1.2. Bi-directional Gated Recurrent Unit Network

It was shown that Recurrent Neural Networks suffer from the vanishing gradient problem [16][17]. To address this problem, gated recurrent structures have been proposed: *Gated Recurrent Unit* (GRU) [18]. A GRU is characterized by two gates: reset gate and update gate. Two gates adaptively control how much information from the hidden state should be carried to the next state and how much information from the current input should be taken in the next hidden state. It was shown that processing the sequence in two directions (i.e., forward and backward: Bi-directional GRU) can improve the performance of many sequence modeling applications [19].

Our Bi-GRU has a similar configuration as the CNN (see Figure 2.). The input is the sequence of words, represented as one-hot vectors. The first layer is an embedding layer; the second is a Bi-GRU layer. In the bidirectional recurrent structure, the last hidden states of GRU in the forward direction and backward direction are merged (by concatenation in our study) to form a single vector that represents the word sequence. The last layer is a dense layer with a sigmoid as activation function. The last layer has as many neurons as OOV PN classes.

2.2. OOV PN ranking

The goal of our system is to select a list of OOV PNs that are highly relevant to the spoken document. Since every predicted OOV class consists of many OOV proper nouns, we need a ranking procedure. That is, for each spoken document, we need to retrieve OOV PNs based on their relevance to the semantic content of the document. This could be achieved by quantifying the semantic similarity between the spoken document and each OOV proper noun of the selected classes. To this end, we represent an OOV PN word (*oov)* by its word embedding (v_{oov}) and a document (V_{Doc}) by the average of the word embedding (v_w) of the in-vocabulary words (w) of the document (global semantic context representation):

$$V_{Doc} = \frac{1}{\#w} \sum_{w \in Doc} v_w$$

We use the cosine distance as a measure of semantic similarity between these two vectors:

$$distance(Doc, oov) = 1 - \frac{V_{doc} \cdot v_{oov}}{\|V_{doc}\|^2 . \|v_{oov}\|^2}$$

3. EXPERIMENTAL SETUP

3.1. Data

For the multipurpose and the development text corpuses we used French Wikipedia articles. For the test text and audio corpus, we used French news articles from the *Euronews* channel.

3.1.1. Text corpora

The **multipurpose corpus** is necessary to build word embeddings and to train DNN models to infer OOV PNs relevant to the development and the test corpora. We used French Wikipedia articles. We observed that the length of the articles varies greatly. We decided to remove the articles with less than 10 words (after pre-processing).

We used 1% of the Wikipedia data (about 14K documents) as a **development set** to tune the hyperparameters of our models. The development corpus is used to evaluate the methodology proposed in this paper and to adjust the involved parameters. For the **test corpus**, we used the text articles from the *Euronews* website channel [20], written from January 2014 to June 2014. We selected only the articles containing at least one OOV PN word. The system is evaluated with the adjusted parameters on the test corpus.

From Table 1 we can observe that the Wikipedia corpus contains *1.7M* unique OOV PNs (OOV PN unigrams) with respect of our *96K* LVCSR vocabulary (see section 3.1.5). This number of OOV PNs is very large and it is impossible to add all these words to the LVCSR vocabulary. We assume that the Wikipedia corpus contains a large number of

OOV PNs used in the *Euronews* corpus. The OOV PNs rate in the *Euronews* corpus is *2.6*% (*7.9K* out of *301K*).

3.1.2. Pre-processing of the text data

To train and evaluate proposed models, the text corpus was preprocessed: the *TreeTagger part-of-speech tagging tool* is used to automatically tag PNs. The TreeTagger is reported to perform with an accuracy of 95.7% on French data [21]. The PNs and non-PN words that occur in the lexicon of our LVCSR system are tagged as IV and the remaining PNs are tagged as OOV. For training the DNN models, words in the multipurpose text corpus are lemmatized. Moreover, PNs and non-PNs occurring less than 5 times are discarded. A stop-list of common words and non-content words is applied and a part-of-speech based filter is employed to choose words tagged as PN, noun, adjective, verb and acronym. Other words are discarded. The context models are trained and evaluated with this filtered corpus. For *word2vec* word-embedding this pre-processing is not performed.

Wikipedia corpus (train and development corpora)	
Number of documents	1.4M
Corpus size (word count)	228M
Vocabulary size	2.4M
OOV PN unigrams	1.7M
Total count of OOV PNs	20M
Euronews corpus (test corpus)	
Number of documents	3.1K
Corpus size (word count)	301K
Vocabulary size	23.8K
OOV PN unigrams	3.9K
Total count of OOV PNs	7.9K

Table 1. Statistics about text data after pre-processing. *K* denotes *thousand* and *M* denotes *million.*

3.1.3. Test audio corpus

In order to evaluate the recognition performance of the proposed approaches, we collected a **test audio corpus**. This corpus is composed of *296* French videos from the *Euronews* channel, corresponding to a total duration of 6 hours (corpus size *60K* words). OOV PN rate is 1.3% [22].

3.1.4. OOV PN retrieval performance measures

We used several evaluation measures:
(1) For OOV PN class evaluation, we used the *recall at top-Q* predicted classes (*Recall@Q* in figures) and the *F-measure at top-Q* predicted classes (*F-measure@Q* in figures) [23].These metrics are calculated for each document and averaged over all evaluation documents.
(2) For OOV PN ranking, we computed the *recall* (*Recall*): for each document, the OOV PNs of the best predicted classes are sorted. After this, the recall for top-N words (*operating point*) is computed and averaged over all evaluation documents [24].

3.1.5. Recognition system

The Kaldi-based Automatic Transcription System (KATS) employs context dependent DNN-HMM phone models trained on *200*-hour broadcast news audio files. The *original lexicon* contains about *96K* words. Using the *Pocolm* toolkit [25], bigram and trigram language models are estimated on the "*Le Monde" + Gigaword* corpus and used to produce the word lattice. The original lexicon of *96K* words is augmented by adding the retrieved OOV PN word list as follows. For each development/test file: (1) we create a ranked list of OOV proper nouns; (2) we keep only top-N words of this list; (3) the top-N words are added to the original lexicon and to the language model. Finally, we obtain an *extended lexicon* of (*96K+N*) words for each test file. We perform a second pass speech recognition to recognize the OOV PNs by updating the LVCSR system using the extended lexicon and modified language model.

To update the pronunciation lexicon, automatic G2P converters are used [26], [27]. Up to three pronunciations per new OOV PN are generated using a CRF [9]. Regarding the language model, we added only unigram probability for the new OOV PNs. The mass probability attributed to the new OOV PN unigrams has been subtracted from the probability of the <*unk*> probability [11]. Thus, the total unigram probability adds up to *1*.

3.2. Context model configurations

As training set, the multipurpose train text corpus (Wikipedia) is used. For each model, the parameter set is adjusted on the development text corpus (one part of Wikipedia) and the best parameter sets are given:

- *Word embedding* is used for three different purposes: (1) to initialize the embedding layer of the neural models; (2) to build document embeddings by taking the average of the word embeddings of in-vocabulary of the document; (3) to rank OOV PN list. For computing the word embeddings, we use the skip-gram *word2vec* with *400* dimensions, context window of *20* words, and hierarchical softmax. We only consider words that occur at least *5* times in the training corpus (Wikipedia). Finally, we obtained the embeddings for about *562K* OOV PNs.
- *fastText.* As baseline, the standard text classification tool *fastText* [28] is used. As input, the raw text is used. As output, the class labels are given. Therefore, one document can have several class labels.
- *CNN.* The first layer is composed of an embedding layer whose weights are initialized by the *word2vec* embeddings. We used one convolutional layer followed by a max pooling layer and a ReLU dense layer. A dropout of *0.2* is applied before the output layer. Our preliminary experiments showed that trainable word embeddings slightly improve the performance of CNN. Thus, we set the embedding layer to be trainable.

- *Bi-GRU*. Like the CNN, the first layer is composed of an embedding layer whose weights are initialized by the *word2vec* embeddings. The second layer is a bidirectional GRU with *Adam optimizer* [29], early stopping and *0.2* dropout rate. The standard backpropagation algorithm with stochastic gradient descent is used to train the network. To speed up the training, we used static word embeddings (the first layer is frozen).

4. EXPERIMENTAL RESULTS

4.1. OOV PN class prediction results

The classification consists in finding for each document which OOV PN classes this document could contain. It is multi-class labeling because one document can correspond to several OOV PN classes. The classification is performed using the context models proposed in this paper.

For defining the classes, we used a K-means clustering for grouping OOV PNs into classes. We used word embedding and cos-distance to cluster OOV PN words. We created *1000* classes. It is a good compromise between the number of OOV PNs in the Wikipedia corpus (*1.7M*) and the average number of words per class. A smaller number of classes decreases the model performance.

4.1.1. CNN

CNNs require careful tuning of many hyperparameters. We explored two important design choices: the window size and the number of filters. In these experiments, we keep all hyperparameters of the model fixed and we investigate the impact of various values of a studied parameter.

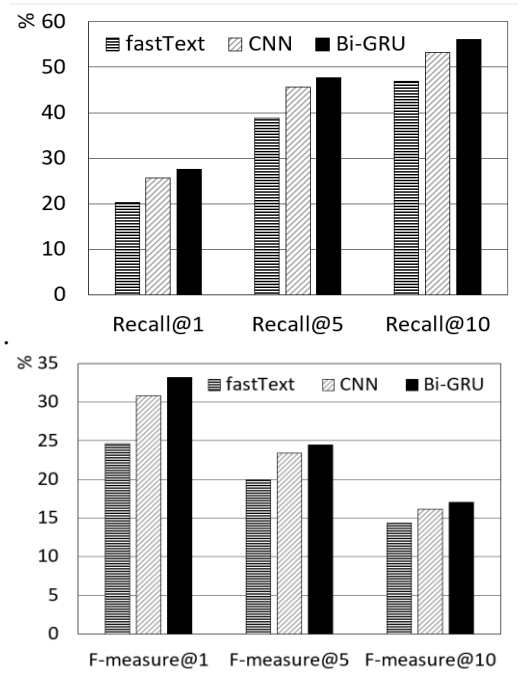

Figure 3. OOV PN classification results in term of recall and F-measure. *Euronews* text test data.

The results of this experiment on the development set (Wikipedia) are shown in Table 2. We cannot observe any trend regarding the change of the window size and the classification performance. Regarding the impact of the number of filters, from Table 2 it can be observed that increasing the number of convolutional filters leads to consistent performance gains. Note that the number of filters represents the dimensionality of the document representation due to the max-pooling operation. CNN with *1000* filters and window size of *3* is used in the following experiments.

4.1.2. Bi-GRU

Since our Wikipedia multipurpose data is large, training many Bi-GRU models is time consuming. We only report the result for *400*-dimensional Bi-GRU and *800*-dimensional Bi-GRU. The dimensionality of the model refers to the size of the hidden state. For example, the *400*-dimensional Bi-GRU consists of two GRUS each with *200* hidden state, one GRU processes the input sequence in the forward direction while the other GRU processes the input sequence in the backward direction. The result of the two GRU layers is concatenated. From Table 2, it can be observed that the *800*-dimensional Bi-GRU outperforms the *400*-dimensional BI-GRU. Bi-GRU shows the best performance compared to the other models investigated in this article. This good performance of Bi-GRU is probably because the hidden states of gated recurrent architectures have a large capacity for encoding sequential information due to the gate mechanism.

Dev set (Wiki)	*Recall@5*	*Recall@10*
Window size	**CNN**	
3	**45.4**	54.5
4	45.1	**54.7**
5	45.0	54.0
6	44.8	54.1
Nbr of filters		
300	45.4	**55.5**
600	46.1	55.4
1000	**46.2**	**55.5**
Dimension	**Bi-GRU**	
400 (2x200)	46.5	56.0
800 (2x400)	**47.2**	**56.7**

Table 2. CNN. The impact of the window size and of the number of filters on the classification results for the CNN. Impact of dimensionality for Bi-GRU. Development set (Wikipedia).

4.1.3. Discussion

Figure 3 summarizes the OOV PN classification results on the *Euronews* test set using the best configurations for each model. We observe that the *fastText* model, used as baseline,

is outperformed by the proposed models. CNN models give slightly inferior results compared to Bi-GRU. The best performance is obtained by the Bi-GRU context model. In terms of F-measure, this model obtains *33.2%* using the best class.

4.2. OOV PN ranking results

In the previous sections, we computed the results at the class level. In this section, for each test document, we assume that we have retrieved one or several OOV PN classes. From these classes, we extract a list of ranked OOV PNs.

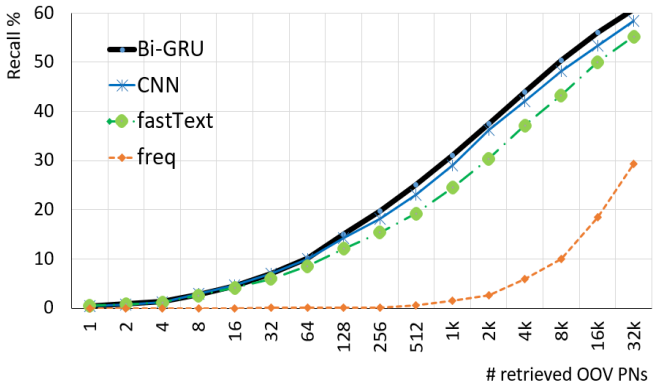

Figure 4. Recall performance of OOV PN retrieval for *Euronews* text test set.

Figure 4 shows the OOV PNs ranking results in terms of recall on the *Euronews* text test set. In Figure 4, the *x*-axis represents the number of OOV proper nouns (in logarithmic scale) that could be added to the LVCSR lexicon before running second-pass decoding. The *y*-axis represents the OOV PN retrieval performance measured by the average recall on the *Euronews* test set. From Figure 4 we can observe that Bi-GRU performs better than the extension of the LVCSR lexicon based on *fastText* and CNN. We can also notice that Bi-GRU, CNN and *fastText* outperform the adding of the most frequent OOV PNs according to occurences on the Wikipedia corpus (method called *freq* in the figure).

Compared to [11], presented results are better in terms of recall. It is important to note, that in [11] a small diachronic corpus close to the test corpus was used to retrieve OOV PNs. In the present approach, the task of retrieval is more difficult because we employed a very large multipurpose corpus (Wikipedia). The advantage of this corpus is that it is multi-topic and covers a very large number of PNs. Thus, it can be used not only for the broadcast news domain but for other domains.

4.3. Speech recognition results

In this section, the list of relevant OOV PNs, that are retrieved according to the proposed methodology, is used for speech recognition. We extract top-4000 OOV PNs from the list of retrieved OOV PNs per test audio document. We construct the extended lexicon (original lexicon and 4K OOV PNs) per document and update the language model per document. We perform a second pass speech recognition with the updated LVCSR system.

Table 3 displays the word error rate (WER) and proper name error rate (PNER) results on the test audio corpus using original LVCSR and LVCSR with extended lexicon (using the Bi-GRU context model and second-pass recognition). PNER is obtained by first aligning the reference and hypothesized word level transcriptions and then calculating substitution, deletion, insertion errors, and the error rate only on the proper name terms. From Table 3, we can see that adding the new PNs into the lexicon and language model had a positive impact on the WER. The WER showed an improvement. This improvement is due to the recognition of OOV PNs and the reduction of insertion and deletion errors. The PNER reduction is larger compared to WER (6% relative improvement). This PNER decrease shows that new proper nouns were appropriately added to the lexicon.

Method	***WER***	***PNER***
LVCSR with original lexicon	20.3	39.9
LVCSR with extended lexicon (using Bi-GRU)	**19.9**	**37.4**

Table 3. WER (%) and PNER (%) performance of OOV PN retrieval for *Euronews* audio test set using second-pass recognition.

5. CONCLUSION

In this work, we address the problem of improving recognition of an ASR system by adding new OOV PNs to the lexicon. We propose to retrieve OOV PNs dynamically for each test file without retraining the other components of the system. For this, a multipurpose corpus is used: Wikipedia. The great advantages of this corpus are a large set of OOV PNs and a multi-topic domain. Thus, this corpus can be used for any domain of speech recognition. The proposed approach can be performed for any language.

We proposed a global semantic context model relying on DNNs. The Bi-GRU model shows the best performance. On the test data, the models we proposed yielded improvements in recall over the method based on *fastText*; in addition, on the French broadcast news transcription, Bi-GRU model led to a decrease in PN error rate. This shows that new proper nouns were appropriately selected.

6. ACKNOWLEDGMENT

This work is funded by the Chist-Era AMIS (Access Multilingual Information opinionS) project. Experiments presented in this paper were partly carried out using the Grid'5000 testbed, supported by a scientific interest group hosted by Inria, CNRS, RENATER and other Universities and organizations (https://www.grid5000.fr).

7. REFERENCES

[1] Qin, L. and Rudnicky, A. "OOV word detection using hybrid models with mixed types of fragments," *Interspeech*, pp. 2450–2453, 2012.

[2] Chen, W., Ananthakrishnan, S., Prasad, R. and Natarajan, P. "Variablespan out-of-vocabulary named entity detection," *Interspeech*, pp. 3761–3765, 2013.

[3] Li, J., Ye, G., Zhao, R., Droppo, J., Gong, Y. "Acoustic-To-Word Model Without OOV", *ASRU*, 2017.

[4] Ming Sun, A., Chen, Y. "Learning OOV through semantic relatedness in spoken dialog systems," *Interspeech*, pp. 1453–1457, 2015.

[5] Maergner, P., Waibel, A., and Lane, I. "Unsupervised vocabulary selection for real-time speech recognition of lectures," *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 4417–4420, 2012.

[6] Oger, S., Linares, G., Bechet, F., Nocera, P. "On-demand new word learning using world wide web", ICASSP, 2008.

[7] Ohtsuki, K., Hiroshima‡ N., Oku, M., and Imamura, A. "Unsupervised vocabulary expansion for automatic transcription of broadcast news", ICASSP 2005.

[8] Blei, D. M., Ng, A.Y., and Jordan, M.I. "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[9] Iyyer, M., Manjunatha, Y., Boyd-Graber, J., and Daume H. "Deep unordered composition rivals syntactic methods for text classification," *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 1681– 1691, 2015.

[10] Sheikh, I. Illina, I. Fohr, D., and Linares, G. "Improved neural bag-ofwords model to retrieve out-of-vocabulary words in speech recognition," *Interspeech*, pp. 675–679, 2016.

[11] Sheikh, I., Fohr, D., Illina, I., Linares, G. "Modelling Semantic Context of OOV Words in Large Vocabulary Continuous Speech Recognition", *IEEE/ACM Transactions on Audio, Speech and Language Processing*, 25 (3), pp.598 – 610, 2017.

[12] Sheikh, I. Illina, I., Fohr, D., and Linares, G. "Document level semantic context for retrieving OOV proper names," *IEEE International Conference on Acoustics, Speech and Signal Processing*, ICASSP, pp. 6050–6054, 2016.

[13] Le, Q., and Mikolov, T. "Distributed representations of sentences and documents", International Conference on Machine Learning, 1188-1196, 2013.

[14] Mikolov, T., Chen, K., Corrado, G., Dean, J. "Efficient estimation of word representations in vector space", *arXiv preprint arXiv:1301.3781,* 2013.

[15] Kim, Y. "Convolutional Neural Networks for Sentence Classification", *EMNLP*,2014.

[16] Bengio, Y., Simard, P., and Frasconi, P. "Learning long-term dependencies with gradient descent is difficult", *IEEE Transactions on Neural Networks*, pp. 157–166, 1994.

[17] Jozefowicz, R., Zaremba, W., and Sutskever, I. "An empirical exploration of recurrent network architectures", *International Conference on Machine Learning*, pp. 2342–2350, 2015.

[18] Cho, K., Merriënboer, B., Gülçehre, C., Bahdanau, Bougares, D. F., Schwenk, H., Bengio, Y. "Learning Phrase Representations using RNN Encoder--Decoder for Statistical Machine Translation", *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing* (EMNLP), 2014.

[19] Graves, A., Mohamed, A., and Hinton, G. "Speech recognition with Deep Recurrent Neural Networks", *IEEE Acoustics, Speech and Signal Processing International Conference*, pp. 6645–6649, 2013.

[20] http://fr.euronews.com/

[21] Allauzen, A. and Bonneau-Maynard, H. "Training and evaluation of pos taggers on the french multitag corpus", *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC)*, 2008.

[22] Sheikh, I. Illina, I., and Fohr, D. "How diachronic text corpora affect context based retrieval of OOV proper names for audio news", *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC)*, pp. 3851–3855, 2016.

[23] Weston, J., Chopra, S., and Adam, K. "#TagSpace: Semantic embeddings from hashtags", *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing* (EMNLP), pp. 1822–1827, 2014.

[24] Manning, C. D., Raghavan, P. and Schutze, H. "Introduction to Information Retrieval", *Cambridge, UK: Cambridge University Press*, 2008.

[25] Povey, D. https://github.com/danpovey/pocolm

[26] Bisani M. and Ney, H. "Joint-sequence models for grapheme-to-phoneme conversion", *Speech Communication*, vol. 50, no. 5, pp. 434 –451, 2008.

[27] Illina, I.,. Fohr, D., and Jouvet, D. "Multiple Pronunciation Generation using Grapheme-to-Phoneme Conversion based on Conditional Random Fields", *XIV International Conference «Speech and Computer» SPECOM*, 2011.

[28] Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. "Enriching Word Vectors with Subword Information", *Transactions of the Association for Computational Linguistics*, vol. 5, Issue 1, 2017.

[29] Diederik, K. and Ba, J. "Adam: A method for stochastic optimization", *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2015.