What is best for Spoken Language Understanding: Small but Task-dependant Embeddings or Huge but Out-of-domain Embeddings?

Sahar Ghannay, Antoine Neuraz, Sophie Rosset

WHAT IS BEST FOR SPOKEN LANGUAGE UNDERSTANDING: SMALL BUT TASK-DEPENDANT EMBEDDINGS OR HUGE BUT OUT-OF-DOMAIN EMBEDDINGS?

Sahar Ghannay[1], *Antoine Neuraz*[1,2], *Sophie Rosset*[1]

[1] Université Paris-Saclay, CNRS, LIMSI, 91400, Orsay, France
[2] Department of Biomedical Informatics, Hôpital Necker-Enfants Malades, APHP INSERM UMRS 1138, Team 22, Paris Descartes, Université Sorbonne Paris Cité

ABSTRACT

Word embeddings are shown to be a great asset for several Natural Language and Speech Processing tasks. While they are already evaluated on various NLP tasks, their evaluation on spoken or natural language understanding (SLU) is less studied. The goal of this study is two-fold: firstly, it focuses on semantic evaluation of common word embeddings approaches for SLU task; secondly, it investigates the use of two different data sets to train the embeddings: small and task-dependent corpus or huge and out-of-domain corpus. Experiments are carried out on 5 benchmark corpora (ATIS, SNIPS, SNIPS70, M2M, MEDIA), on which a relevance ranking was proposed in the literature. Interestingly, the performance of the embeddings is independent of the difficulty of the corpora. Moreover, the embeddings trained on huge and out-of-domain corpus yields to better results than the ones trained on small and task-dependent corpus.

Index Terms— spoken language understanding, word embeddings

1. INTRODUCTION

A textual or spoken task-oriented dialogue system involves several modules, which typically include natural language understanding, speech recognition, generation, and a dialogue manager. In this paper, we are interested in natural/spoken language understanding (SLU). The objective of SLU is to produce a semantic analysis and an formalization of the user's utterance. SLU is often divided into 3 sub-tasks: domain classification, intent classification, and slot-filling [1]. The latter can also be considered as a concept detection task [2]. Over the past five years, most of the work focused on neural architectures [3, 4, 5]. While preliminary studies [6] suggested that neural architecture could benefit from embeddings pre-trained on huge corpora, most of the work did not use such pre-trained embeddings (see for example [4]). More recently, Korpusik et al. [7] investigate the transfer ability for SLU task of a pre-trained BERT representation [8]. They demonstrate state-of-the-art performance on different benchmark corpora including ATIS, restaurant queries, and written and spoken meal descriptions.

Word embeddings were shown to be a great asset for several Natural Language Processing [9, 10] and Speech Processing tasks [11, 4]. They were evaluated on different tasks such as POS tagging, chunking, analogical reasoning, similarity [12, 13, 14, 15] and ASR error detection [11]. However, their evaluation on spoken or natural language understanding is less studied. In this paper, we focus on the semantic evaluation of common word embeddings approaches for such a task. A comprehensive study covering a wide range of evaluation criteria and popular embeddings approaches was proposed by [16]. They used three different evaluation criteria: word relatedness, coherence (i.e. groups of words in small neighborhood in the embeddings space are mutually related) and downstream performance. Depending on the criterion, the ranking of the embedding methods varied. Similarly, in a previous study focusing on embeddings trained on labeled or unlabeled data [13], we showed that depending on the downstream task (i.e. analogical reasoning or similarity) the results were not consistent. For that reason, we investigated and evaluated several approaches to combine word embeddings to advantage of their complementarity, and create more versatile word embeddings.

This study focuses on semantic evaluation of common word embeddings approaches for SLU task, with the aim of building a fast, robust, efficient and simple SLU system. We focus on SLU task as it is defined by [1], specially on concept detection. We propose to use a simple BiLSTM architecture composed of two hidden layers, which is enriched only with word embeddings, no additional features being used. Thus, we compare context independent (GloVe, Skip-gram, FastText, CBOW) and contextual (ELMo) word embeddings approaches. Then, we investigate the use of two different data sets to train the embeddings: small and task-dependent corpus or huge and out-of-domain corpus. Experiments are carried out on 5 benchmark corpora (ATIS, SNIPS, SNIPS70, M2M, MEDIA), on which a ranking from the most ambiguous to the almost-solved one was proposed by [17].

This work has been partially funded by the LIHLITH project (ANR-17-CHR2-0001-03), and supported by ERA-Net CHIST-ERA, and the "Agence Nationale pour la Recherche" (ANR, France).

2. WORD EMBEDDINGS DESCRIPTION

Word embeddings are real-valued vector representations of words, that corresponds to projections in a continuous space of words supposed to preserve the semantic and syntactic similarities between them. During the last few years, many approaches have been proposed to build word embeddings. These approaches can differ in the type of architecture used and the training time. In this study we focus on the evaluation of the popular word embeddings approaches, which fall into two categories: context independent (Skip-gram, CBOW, GloVe, FastText) and contextual (ELMo), described in the next sections.

2.1. Word2vec

Word2vec approach, proposed in [18], is composed of a two-layer neural net that processes text. Its input is a text corpus and its output is a set of feature vectors for words in that corpus. It can be obtained using two methods: Skip-gram and continuous bag of words (CBOW), described as follows.

Skip-gram: This architecture is trained using the negative-sampling procedure. It takes as input the target word w_i and outputs the context words C. The context is not limited to the immediate context, and training instances can be created by skipping a constant number of words in its context, for instance, w_{i-3}, w_{i-4}, w_{i+3}, w_{i+4}, hence the name *Skip-gram*.

CBOW: This architecture takes as input the preceding and the following words $C = w_{i-2}, w_{i-1}, w_{i+1}, w_{i+2}$ of the target word w_i, and outputs the resulting word w_i. The non-linear hidden layer is removed, and the projection layer is shared for all contextual words C, which are projected on the same position. Moreover, the order of words in the context does not influence the projection; this is the reason of why this architecture is called bag of word model in the literature. The CBOW model learns vectors to predict a word given its context C by averaging the contextual word vectors and then running a log-linear classifier on the averaged vector to get the resulting word w_i.

2.2. GloVe

This approach is introduced by [14], and relies on constructing a global co-occurrence matrix X of words, by processing the corpus using a sliding context window. Here, each element X_{ij} represents the number of times the word j appears in the context of word i. The model is based on the global co-occurrence matrix X instead of the actual corpus, thus the name GloVe, for Global Vectors.

2.3. FastText

The FastText approach was proposed in [15], and is based on the Skip-gram model. This model learns word representations by taking into account the morphology which is modeled by considering sub-word units and representing the word as a bag of character n-grams. Thus, each word is represented as the sum of its character n-gram representations. This approach allows to compute word representations for words that did not appear in the training data, which is not the case for other approaches.

2.4. ELMo

ELMo (Embeddings from Language Models) is a deep contextualized word representation that models both complex characteristics of word use (e.g., syntax and semantics), and how these uses vary across linguistic contexts (i.e., to model polysemy) [19]. They are computed on top of two-layer bidirectional language model (biLM) with character convolutions, as a linear function of the internal network states. ELMo embeddings differ from previous word embeddings approaches in that each token is assigned a representation that is a function of the entire input sentence. In addition, ELMo can learn a linear combination of the vectors stacked above each input word for each end task, which improves performance over just using the top LSTM layer.

3. EXPERIMENTS: DATA AND RESULTS

In this section, we present the data used in our work, the experimental setup and then the results.

3.1. Data

We use 5 standard benchmark corpora (ATIS, SNIPS, SNIPS70, M2M, MEDIA), corresponding to different domains and applications. The Air Travel Information System (ATIS) corpus concerns flight information [20]. The MEDIA corpus is about hotel reservation and information [2]. The M2M corpus contains dialogues for restaurant and movie ticket booking [21]. The SNIPS corpus is a multi-domain dialogue corpus collected by the SNIPS company [22]. It is composed of 7 in-house tasks (intents) such as Weather information, restaurant booking, managing playlist, etc. The SNIPS70 corpus is a sub-part of the SNIPS corpus, in which the training set is limited to 70 queries per intent randomly chosen. MEDIA data are in French while the other corpora are in English.

In [17], the authors propose a classification of those corpora according to their level of difficulty for system development. The ranking, from the most difficult to the easiest, is as follows: MEDIA, SNIPS70, SNIPS, SNIPS70 ATIS, M2M. Table 1 summarizes the description of all the corpora. More information can be found in [17]. Note that the MEDIA corpus used in this study is slightly different from the one used by [17], since our version contains empty turns and turn-mixes (*i.e.* turns of dialogues where there is a mix of speakers).

Corpus	ATIS	MEDIA	SNIPS	SNIPS70	M2M
vocab.	1117	2463	14354	4751	900
#tags	84	70	39	39	12
train size	4978	12908	13784	2100	8148
test size	893	3518	700	700	4800

Table 1. Benchmark corpora description

3.2. Experimental setup

3.2.1. Word embeddings training

Besides studying the performance of the different embeddings approaches, we are interested in the impact of the corpora used for their training: small and task-dependent corpus or huge and out-of-domain corpus.

For small and task-dependent corpus we used the training part of the benchmarks. In addition, we kept all the words due to the small data size.

For huge and out-of-domain corpus we used Wikipedia dumps (WIKI) in English and French, which are composed respectively of 2 billion and 573 million of words. Note that words occurring less than 5 times have been discarded, resulting in a vocabularies sizes of 923k words for French and for 2 million words for English. The common parameters used to train Skip-gram, CBOW, GloVe and FastText are: window size = 5, negative sampling = 5, dimension = 300. They have been selected based on previous studies [14, 15].

For the ELMo embeddings we used the default parameters [1]. The dimension of the resulting ELMo embeddings is equal to 1024, which corresponds to the weighted average of all biLM layers. As the training of ELMo on Wikipedia data from scratch takes a lot of time (more than 1 month on one GPU), we decided to used the publicly available pre-trained models [2], which are trained on 20-million-words data randomly sampled from the raw text released by the CoNLL 2018 shared task [23].

3.2.2. SLU model

The SLU model used in this study is based on the pytorch NeuroNLP2 implementation[3] [24], which is a BiLSTM (Bidirectional long short-term memory) network, that has been proven to be relevant to model output dependencies on MEDIA and ATIS data [3, 4, 5]. The network is composed of two hidden layers of n hidden units, followed by a Softmax output layer.

For our experiments we made some hyper-parameters tuning by varying the size of the BiLSTM hidden layers $n \in \{128, 256, or 512\}$ and the batch size $b \in \{16, 32, 64\}$. The feature set fed to the network is composed only of the word embeddings of size $d \in \{1024, 300\}$ according to the embedding approach as mentioned in section 3.2.1. Note that for test corpus, the out of vocabulary (OOV) words are represented by null vectors, except for FastText and ELMo, which are able to predict vectors for OOV words. The word embeddings have been frozen during training, in order to evaluate the performance of the different embeddings on SLU task, as it has been shown in [25] that fine-tuned different word embeddings show very similar performance and provide comparable results.

3.3. Quantitative evaluation

In this section, we provide the quantitative results for all the benchmark corpora, by comparing the different embeddings approaches trained on small and task-dependent (in domain) corpus and on huge and out-of-domain corpus (WIKI English or French). The results are evaluated using the standard evaluation metrics: F-measure F1 computed by conlleval [4] evaluation script that consider a segment correct if both boundaries and class are correct. In addition, we used Wilcoxon signed-rank test [5] to evaluate the significance of the results. The result is significant if the P-value [6] is lower than 0.05.

Results, summarized in Table 2, show that when the embeddings are trained on task-dependent data, GloVe significantly achieves the best results on all the benchmark corpora except ATIS. This is mainly due to the fact that GloVe, being a count-based approach, is not impacted by the small size of the training corpus, and can takes advantage of global context even if the training corpus is small. FastText achieves the lowest results on the five corpora. It seems that FastText is the most impacted approach by the small corpus size.

In addition, the embeddings trained on huge and out-of-domain corpus yield to better results than the ones trained on small and task-dependent corpus for all the benchmark corpora. Each of these embeddings is better in one of those benchmark corpora except CBOW that achieves the best results on ATIS and SNIPS70. We notice that increasing the data size and changing the domain have a big impact on FastText and yields to significant improvements varying from 9.8 to 41.56 points of F1. FastText achieves the best results on SNIPS. Another result is that context independent approaches outperform significantly the contextual embeddings (ELMo) on all the benchmark except for MEDIA where ELMo outperforms slightly (not significantly) CBOW.

Finally, as we mentioned before, the five benchmark corpora are ranked according to their level of difficulty for system development, however the performance of the embeddings is independent of the corpora difficulty. This can be explained by the fact that this difficulty is located at a level that does not seems to be directly modeled by the word embeddings approaches.

[1] https://github.com/allenai/bilm-tf

[2] https://github.com/HIT-SCIR/ELMoForManyLangs

[3] https://github.com/XuezheMax/NeuroNLP2

[4] https://github.com/tpeng/npchunker/blob/master/conlleval.pl

[5] https://en.wikipedia.org/wiki/Wilcoxon_signed-rank_test

[6] the probability that there is no difference between the samples

	task-dependent					Out-of-domain				
Bench.	**ELMo**	**FastText**	**GloVe**	**Skip-gram**	**CBOW**	**ELMo**	**FastText**	**GloVe**	**Skip-gram**	**CBOW**
M2M	88.89	72.13	**92.54**	88.87	89.39	91.14	93.01	91.77	**93.19**	92.13
ATIS	**94.38**	85.72	92.95	90.84	91.87	94.93	95.52	95.35	95.62	**95.77**
SNIPS	78.68	76.35	**87.40**	82.10	83.94	90.29	**94.85**	93.90	94.43	94.05
SNIPS70	53.06	38.19	**63.65**	47.11	49.76	75.19	79.75	78.68	78.90	**80.13**
MEDIA	80.26	71.73	**82.66**	80.01	79.57	**86.42**	85.30	85.11	85.95	86.06

Table 2. Tagging performance of different word embeddings trained on task-dependent corpus (ATIS, MEDIA, M2M, SNIPS or SNIPS70) and on huge and out-of-domain corpus (WIKI English or French) on all benchmark corpora in terms of F1 using conlleval scoring script (in %)

3.4. Qualitative evaluation

To perform a visual evaluation of the word representations we computed the t-SNE representations of the data sets transformed using the various embedding methods. For a given method and task, we compared the t-SNE obtained using embeddings learned on a small in-domain corpus versus a large general corpus (WIKI). This visual evaluation concerns the words that carry out frequent semantic tags that have an F1 score lower than the median. An example on MEDIA (top) using ELMo and on SNIPS70 using CBOW (bottom) is given in Figure 1. When comparing the representations, we observe

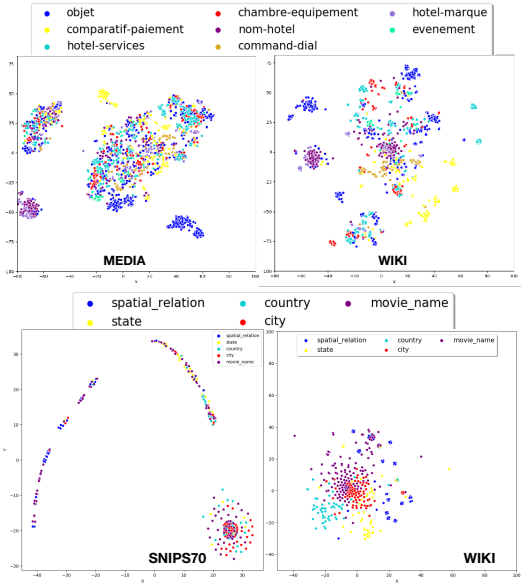

Fig. 1. t-SNE representations on MEDIA using ELMo and on SNIPS70 using CBOW, showing the most frequent tags that have F1 score below the median, when they are trained on Media or SNIPS70 and Wikipedia data (WIKI)

that tags of the same types are more scattered on the representation learned on the small in-domain corpus, whereas they are more compact and clustered when using the large and general corpus. This better separation between terms may allow the downstream model to generalize more efficiently. This phenomena is observed for all the embeddings on all the benchmark corpora.

We were also interested to the evaluation of computation time needed to train and test the embeddings. For training and test time, we observe that ELMo is the slowest one, however we can avoid training time by using pre-trained models. Regarding to the obtained results, for example for MEDIA, ELMo achieves the best results followed by CBOW which is the fastest in terms of train and test time. As for dialog system the SLU model has to be simple, robust, efficient and fast, in this case CBOW is the adequate approach we can use.

4. CONCLUSIONS

This paper presented the evaluation of the word embeddings on SLU task. In this study, we were interested in providing semantic evaluation of common word embeddings approaches (ELMo, FastText, GloVe, Skip-gram and CBOW). We also investigated the use of two different data sets to train the embeddings: small and task-dependent corpus or huge and out-of-domain corpus. Experiments were carried out on 5 benchmark corpora (ATIS, SNIPS and SNIPS70, M2M, MEDIA), on which a relevance ranking was proposed in the literature. Experimental results show that embeddings trained on huge and out-of-domain corpus yields to better results than the ones trained on small and task-dependent corpus, since huge and out-of-domain corpus can capture general semantic and syntactic characteristics that remain relevant to SLU tasks. A conclusion from these experimental results is that the count-based approaches like GloVe are not impacted by the lack of data. However CBOW, Skip-gram and especially FastText need more data for training to be efficient. Each of these embeddings is better in one of the benchmark corpora except CBOW that achieves the best results on ATIS and SNIPS70. The obtained results are interesting, since the embeddings are not tuned during training and we are not using additional features, so those results can be easily improved. Moreover, ELMo is the slowest one in terms of train and and test time, and for downstream tasks (*e.g.* dialog system), it is preferable to use the fastest embedding model that achieves good performance.

5. REFERENCES

[1] Gokhan Tur and Renato De Mori, *Spoken Language Understanding: Systems for Extracting Semantic Information from Speech*, John Wiley & Sons, May 2011.

[2] Hélène Bonneau-Maynard, Christelle Ayache, Frédéric Bechet, Alexandre Denis, Anne Kuhn, Fabrice Lefevre, Djamel Mostefa, Matthieu Quignard, Sophie Rosset, Christophe Servan, and Jeanne Villaneau, "Results of the French Evalda-Media evaluation campaign for literal understanding," in *lrec*, Genoa, May 2006.

[3] Kaisheng Yao, Baolin Peng, Yu Zhang, Dong Yu, Geoffrey Zweig, and Yangyang Shi, "Spoken language understanding using long short-term memory neural networks," in *Spoken Language Technology Workshop (SLT), 2014 IEEE*. IEEE, 2014, pp. 189–194.

[4] Grégoire Mesnil, Yann Dauphin, Kaisheng Yao, Yoshua Bengio, Li Deng, Dilek Hakkani-Tur, Xiaodong He, Larry Heck, Gokhan Tur, Dong Yu, and Geoffrey Zweig, "Using recurrent neural networks for slot filling in spoken language understanding," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 23, no. 3, pp. 530–539, Mar. 2015.

[5] Edwin Simonnet, Sahar Ghannay, Nathalie Camelin, Yannick Estève, and Renato De Mori, "ASR error management for improving spoken language understanding," in *Interspeech 2017*, Stockholm, Sweden, Aug. 2017.

[6] Grégoire Mesnil, Xiaodong He, Li Deng, and Yoshua Bengio, "Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding.," in *Interspeech*, 2013.

[7] James Glass Mandy Korpusik, Zoe Liu, "A comparison of deep learning methods for language understanding," in *Interspeech, September 15–19, 2019, Graz, Austria*.

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *NAACL-HLT*, Minneapolis, Minnesota, June 2019, Association for Computational Linguistics.

[9] Joseph Turian, Lev Ratinov, and Yoshua Bengio, "Word representations: a simple and general method for semi-supervised learning," in *ACL*. Association for Computational Linguistics, 2010.

[10] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa, "Natural language processing (almost) from scratch," *Journal of machine learning research*, vol. 12, no. Aug, 2011.

[11] Sahar Ghannay, Yannick Estève, Nathalie Camelin, Camille Dutrey, Fabian Santiago, and Martine Adda-Decker, "Combining continuous word representation and prosodic features for asr error prediction," in *International Conference on Statistical Language and Speech Processing*. Springer, 2015.

[12] Omer Levy and Yoav Goldberg, "Dependency-based word embeddings," in *ACL*, 2014, vol. 2.

[13] Sahar Ghannay, Benoit Favre, Yannick Estève, and Nathalie Camelin, "Word embedding evaluation and combination," in *10th edition of the Language Resources and Evaluation Conference (LREC 2016)*, Portorož, Slovenia, 2016.

[14] Jeffrey Pennington, Richard Socher, and Christopher D Manning, "Glove: Global vectors for word representation.," in *EMNLP*, 2014, vol. 14.

[15] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, 2017.

[16] Tobias Schnabel, Igor Labutov, David Mimno, and Thorsten Joachims, "Evaluation methods for unsupervised word embeddings," in *EMNLP*, 2015.

[17] Frédéric Béchet and Christian Raymond, "Benchmarking benchmarks: introducing new automatic indicators for benchmarking spoken language understanding corpora," in *Interspeech*, Graz, Austria, September 15–19 2019.

[18] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean, "Efficient estimation of word representations in vector space," in *arXiv preprint arXiv:1301.3781*, 2013.

[19] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer, "Deep contextualized word representations," *arXiv preprint arXiv:1802.05365*, 2018.

[20] P. J. Price, "Evaluation of spoken language systems: The atis domain," in *Proceedings of the Workshop on Speech and Natural Language*, Stroudsburg, PA, USA, 1990, HLT '90, Association for Computational Linguistics.

[21] Pararth Shah, Dilek Hakkani-Tür, Gokhan Tür, Abhinav Rastogi, Ankur Bapna, Neha Nayak, and Larry Heck, "Building a conversational agent overnight with dialogue self-play," *arXiv preprint arXiv:1801.04871*, 2018.

[22] Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, et al., "Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces," *arXiv preprint arXiv:1805.10190*, 2018.

[23] Murhaf Fares, Andrey Kutuzov, Stephan Oepen, and Erik Velldal, "Word vectors, reuse, and replicability: Towards a community repository of large-text resources," in *Proceedings of the 21st Nordic Conference on Computational Linguistics*, Gothenburg, Sweden, May 2017, Association for Computational Linguistics.

[24] Eduard Ma, Xuezheand Hovy, "End-to-end sequence labeling via bi-directional lstm-cnns-crf," in *ACL*. 2016, Association for Computational Linguistics.

[25] Rémi Lebret, Joël Legrand, and Ronan Collobert, "Is deep learning really necessary for word embeddings?," Tech. Rep., Idiap, 2013.