Lexical embedding adaptation for open-domain spoken language understanding

Jeremie Tafforeau
Aix Marseille University - LIF-CNRS
Marseille, France
jeremie.tafforeau@univ-amu.fr

Frederic Bechet
Aix Marseille University - LIF-CNRS
Marseille, France
frederic.bechet@univ-amu.fr

Benoit Favre
Aix Marseille University - LIF-CNRS
Marseille, France
benoit.favre@univ-amu.fr

Thierry Artieres
Aix Marseille University - LIF-CNRS
Marseille, France
thierry.artieres@univ-amu.fr

Abstract

Recently approaches based on word embeddings, i.e. a continuous vector space representation for words, have been used in SLU to overcome the need of annotated corpus by taking advantage of very large collection of unlabeled data to model both semantic and syntactic information. One drawback of embeddings for SLU is the fact that they are usually obtained on very large written text corpora covering generic domains, such as news articles or Wikipedia pages, although SLU systems are dealing with spontaneous speech queries on specific domains. In the standard setting of embedding space usage, two kinds of training corpora are used: a very large unlabeled corpus on which word embeddings are learned, and a smaller in-domain training corpus with gold labels for supervised learning of task specific model. When the in-domain corpus is very specific and represents a different register of language than the unlabeled corpus, a large number of application-specific terms might have no representation in the embedding space. This paper deals with this particular problem of adapting a lexical embedding space to a specific SLU domain. We show that our method outperforms both a baseline using random initialization for unseen vectors and a CRF approach using Part-Of-Speech (POS) and Named Entity (NE) labels for a semantic frame recognition task.

1 Introduction

Semantic parsing is the process of producing semantic interpretations from words and other linguistic events that are automatically detected in a text conversation or a speech signal. Many semantic models have been proposed, ranging from formal models encoding *deep* semantic structures to shallow ones considering only the main topic of a document and its main concepts or entities. For Spoken Language Understanding, hierarchical shallow semantic models are widely used, consisting of determining first the domain, then the intent, and finally the slot-filling entities needed to fulfill a query. Domain, intent and slot labels are directly linked to the application targeted: personal assistant, web queries, ... The drawback of using application-specific labels is the need of an annotated corpus of sufficient size in order to perform supervised learning. For open domain SLU, generic purpose semantic models can be used, such as FrameNet or Abstract Meaning Representation (AMR). Once this generic meaning representation is obtained, a deterministic translation process can be applied for projecting generic predicates and concepts to application specific ones. This kind of approach can help reducing the need of annotated corpus for training SLU models.

Recently approaches based on a continuous vector space representation for words have been used to overcome this need of annotated corpus by taking advantage of very large collection of unlabeled data to model both semantic and syntactic information. In particular researchers in Natural Language Processing have focused on learning dense low dimensional (hundreds) representation space of words [1, 2], called *embeddings*.

In the SLU domain, embeddings have been used in Deep Neural Network (DNN) architectures for domain and intent classification [3] and slot-filling [4]. One drawback of embeddings for SLU, as noted by [5], is the fact that they are usually obtained on very large written text corpora covering generic domains, such as news articles or Wikipedia pages, although SLU systems are dealing with spontaneous speech queries on specific domains. When knowledge bases covering a specific SLU domain are available, such as Web linked data resources for the web query domain, embeddings can be enriched with such prior knowledge as in [5]. However, when the application domain is too specific, and when little annotated data is available, generic purpose embeddings might be inefficient as too many terms might be missing from the embedding space.

In the standard setting of embedding space usage, two kinds of training corpora are used: a very large unlabeled corpus (C_{emb}) on which word representations are learned, and a smaller in-domain training corpus with gold labels for training classifiers on the target SLU task (C_{task}). It is assumed that the syntactic/semantic contexts learned in C_{emb} are coherent with those of the in-domain corpus, and since C_{emb} has a much wider coverage than C_{task}, therefore all the words of C_{task} should have a representation in C_{emb}. When the task specific corpus represents a different register of language than the standard canonical written language (e.g. Wikipedia) covered by C_{emb}, these assumptions are not necessarily true. This is the case when embeddings are used to process spontaneous speech transcriptions of a specific domain for which few manual transcriptions are available. This situation is rather usual in SLU considering the difficulties of collecting spoken conversations for a new use-case. In such a situation unsupervisely learned embeddings should be adapted to the task.

We propose to perform such an adaptation embeddings in two ways. First, embeddings may be included as parameters of the model and may then be fine-tuned during training of the SLU model, a neural network in our case. Second we propose to create relevant embeddings for unknown words to which the above adaptation strategy cannot be applied : **OOE** words without embedding which do occur in the SLU training data and **OOV** words that not even belong to the target domain.

We show that our adaptation strategy improves a simple DNN model over a state-of-the-art baseline using a CRF tagger when there is a mismatch between C_{emb} and the target corpus, especially when only a small amount of data is available to train the models.

2 Related work

The main focus on this paper is on the study of Out-Of-Vocabulary (OOV) words and how to adapt models in order to handle them.

OOV word handling in NLP tasks is dependent on the feature space used to encode data. Features can be computed from the sequence of characters composing the word (e.g. morphological, suffix and prefix features [6, 7]) in order to steer the classifier's decision when the form is unknown. Contextual features try to take advantage of the words in the vicinity of the OOV, such as n-grams in sequence models; contexts can be gathered in external corpora or using web queries [8]. OOVs can also be replaced by surrogates which have the same distributional properties, such as word clusters which have proved to be effective in many tasks [9].

Besides, relying on an embedding space for encoding words opens new possibilities for OOV handling: the availability of large unlabeled corpora for learning embeddings can help reducing the number of OOVs. For words unknown from the task training corpus (C_{task}) but occurring in the embedding corpus (C_{emb}), a similarity distance in the embedding space can be used to retrieve the closest known words and use its characteristics. For words not in C_{emb}, a generic OOV model is used. These methods are reviewed and evaluated in [10] on a dependency parsing task showing that a small performance gain can be obtained when little training data is available. Yet as we will see in section 5 there still exists OOV words for which a particular strategy has to be defined in order to reach optimal results.

3 Corpus and semantic model

We use in this study the *RATP-DECODA*[1] corpus. It consists of 1514 conversations over the phone recorded at the Paris public transport call center over a period of two days [11]. The calls last 3 minutes on average, representing a corpus of about 74 hours of signal. The call center dispenses information and customer services, and the two-day recording period covers a large range of situations such as asking for schedules, directions, fares, lost objects or administrative inquiries.

The *RATP-DECODA* has been annotated with semantic frames. We used a FrameNet-based approach to semantics that, without needing a full semantic parse of an utterance, goes further than a simple flat translation of a message into basic concepts: FrameNet-based semantic parsers detect in a sentence the expression of frames and their roles. Because frames and roles abstract away from syntactic and lexical variation, FrameNet semantic analysis gives enhanced access to the meaning of texts: (of the kind *who does what, and how where and when ?*). We use in this study a FrameNet model adapted to French through the ASFALDA project[2]. The current model, under construction, is made of 106 frames from 9 domains. In the RATP-DECODA corpus, 188,231 frame hypotheses from 94 different frame definitions were found. We decided in this study to restrict our model to the frames generated by a verbal lexical unit. With this filtering we obtained 146,356 frame hypotheses from 78 different frames.

Domain	Frame	# hyp.
SPACE	Arriving	8328
COM-LANG	Request	7174
COG-POS	FR-Awareness-Certainty-Opinion	4908
CAUSE	FR-Evidence-Explaining-the-facts	4168
COM-LANG	FR-Statement-manner-noise	3892
COM-LANG	Text-creation	3809
SPACE	Path-shape	3418
COG-POS	Becoming-aware	2338
SPACE	FR-Motion	2287
SPACE	FR-Traversing	2008

Table 1: Top-10 frame hypotheses in the RATP-DECODA corpus

Table 1 presents the top-10 frames found in our corpus. As expected the top frames are related either to the transport domain (SPACE) or the communication domain (COM and COG). Each frame hypothesis does not necessarily correspond to a frame, most LUs are ambiguous and can trigger more than one frame or none, according to their context of occurrence.

In our experiments, the semantic frame annotations are projected at the word level: each word is either labeled as `null` if it is not part of a frame realization, or as the name of the frame (or frame elements) it represents. In our corpus, 26% of the words have a non-null semantic label and there are 210 different frame labels. A lot of ambiguities come from the disfluencies which are occurring in this very spontaneous speech corpus.

4 A neural network framework for Spoken Language Understanding

Our goal in this study is to evaluate our embedding adaptation strategy for all words missing in C_{emb} corpus. To do so we defined a simple Neural Network architecture that takes these adapted embeddings as input, and predict semantic frame labels for each word as output. In this network the input layer is a lookup layer (also called embedding), that we note Φ, which transforms a sequence of words $(w_1, ..., w_T)$ to a sequence of low dimensional vectors $(\Phi(w_1), ..., \Phi(w_T))$. The transformation Φ is initialized with the embedding learned in an unsupervised fashion using the approach in [1]. It is further fine-tuned during the supervised training of the neural net on the SLU task.

[1] The RATP-DECODA corpus is available for research at the Ortolang SLDR data repository: `http://sldr.org/sldr000847/fr`

[2] https://sites.google.com/site/anrasfalda

More concretely, the neural architecture we use is similar to [12] and is illustrated in Figure 1. It uses a two hidden layers network whose input is a window of 5 successive words in a sentence centered on the word to label. Its expected output is one of the 211 FrameNet tags.

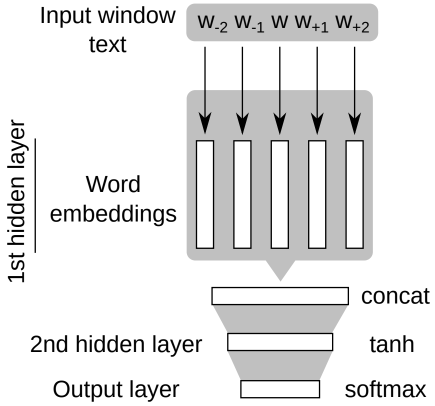


Figure 1: *Our system is a neural network which takes as input a window of words centered on the word to label. It is learned to predict the semantic frame label of the word of interest.*

The first layer is a *lookup* layer that replaces each word by its embedding representation. This layer is implemented as a concatenation of 5 parallel hidden layers of size 300, the dimension of the embedding space, these parameters stand for the word embeddings and can be fine-tuned during training on SLU targets. This first hidden layer is fully connected to a second nonlinear hidden layer (256 neurons in our experiments) which is itself fully connected to an output layer of 211 neurons (one neuron per semantic frame label). This model is learned with stochastic gradient descent using a log-likelihood criterion. We use dropout regularization with a firing rate $p = 0.5$.

5 Adapting lexical embeddings

In this section, we consider words w which belong to C_{test} but which do not belong to the embedding learning corpus (OOE). $w \notin C_{\text{emb}}$ means we cannot provide any proper representation encoded into a regular embedding. This lead to errors even when considering words actually occurring in C_{task}. Setting an initial embedding for words $w \notin C_{\text{emb}}$ could be done by using a unique representation for each of them, either fixed *a priori*, or learned from low frequency words in the C_{emb} corpus[12]. An other option is to assign an individual embedding randomly initialized to each $w \notin C_{\text{emb}}$. We will use this strategy as a baseline in our experimental results.

We propose to estimate a relevant embedding representation for each word $w \notin C_{\text{emb}}$ with the following method:

- retrieving all occurrences of w in C_{task} or C_{test}
- finding the *closest* word t of w in C_{emb} thanks to all the context of occurrence of w
- replacing the unknown embedding of w by the one of t

The closeness between two words is defined according to the similarity between two distributions, one for each word, which represent the empirical distribution of occurrence of the word in all possible contexts (set of K previous words c_p and of K following words c_f).

More formally, we consider a word w and all of its occurrences in all possible contexts as the distribution of n-grams centered on w, where $n = 2K + 1$. This distribution is defined as $\left\{P_w(c_p, c_f), \forall (c_p, c_f) \in C_{\text{task}}^{2K}\right\}$ with:

$$\forall (c_p, c_f) \in C_{\text{task}}^{2K}\ P_w(c_p, c_f) = P(\langle c_p, w, c_f\rangle | w) = \frac{count\langle c_p, w, c_f\rangle}{count\langle w\rangle} \qquad (1)$$

The similarity between two words u and v is computed as the KL-divergence between the two corresponding distributions. At the end, the embedding of a word $w \notin C_{\text{emb}}$ is set to the embedding of its closest word $t = \underset{u \in C_{\text{emb}} \cap C_{\text{task}}}{argmin} \; D_{KL}(P_w||P_u)$.

$$D_{KL}(P_u||P_v) = \sum_{c_p,c_f} P_u(c_p,c_f) \; \log \frac{P_u(c_p,c_f)}{P_v(c_p,c_f)} \tag{2}$$

6 Experiments

The two datasets used in our experiments are the French RATP-DECODA corpus (600K words) for the in-domain labeled corpus and the French part of Wikipedia for the unlabeled C_{emb} corpus (357M words). The train section C_{task} contains 575K words and the test section C_{test} 25K words. This corpus is manually transcribed and annotated with Part-Of-Speech (POS) and Named Entity (NE) labels (used only by the CRF++ experiment). In order to test our adaptation strategy with different sizes of adaptation corpus, we split C_{task} into 10 nested sections from D_0 to D_9.

C_{task}	$\lvert C_{\text{task}} \rvert$	**OOV**	**OOE**
D_0	1,667	1250 — 5.24%	1261 — 5.28%
D_1	11,273	697 — 2.92%	1814 — 7.60%
D_2	23,752	498 — 2.09%	2013 — 8.43%
D_3	65,057	203 — 0.85%	2308 — 9.67%
D_4	151,910	203 — 0.85%	2308 — 9.67%
D_5	230,950	157 — 0.66%	2354 — 9.86%
D_6	311,400	140 — 0.59%	2371 — 9.93%
D_7	387,689	132 — 0.55%	2379 — 9.96%
D_8	477,729	120 — 0.50%	2391 — 10.01%
D_9	576,056	108 — 0.45%	2403 — 10.06%

Table 2: *Distribution of words in the test corpus C_{test} according to the different training partitions. Of course, the sum of **OOV** and **OOE** words is a constant. As the number of words in the task training corpus $|C_{\text{task}}|$ increases, an increasing number of **OOV** words become **OOE** words.*

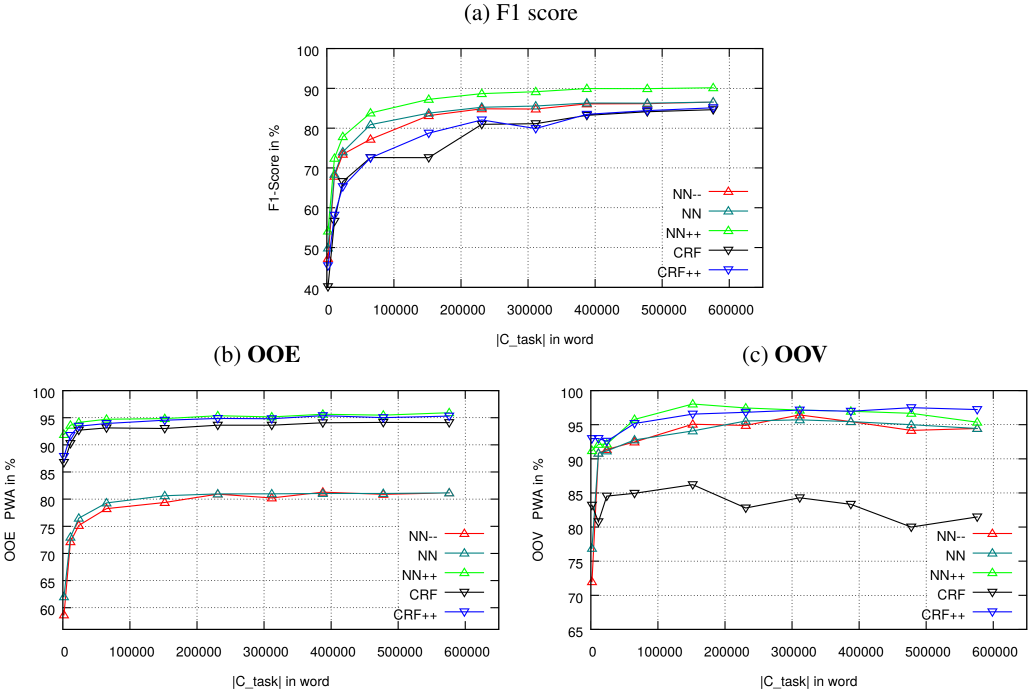
Our experimental results are presented in Table 3 and Figure 2. Five systems are compared:

- **CRF** is a state-of-the-art Conditional Random Field tagger using lexical context of 5 words for predicting the best sequence of FrameNet labels.
- **CRF++** is the same CRF using additional features (Part-Of-Speech, Named-Entities).
- **NN– –** corresponds to our Neural Network model described in section 4. This baseline uses random vectors instead of embeddings learned on C_{emb}.
- **NN** is the same NN using C_{emb} embeddings as word representation. We still consider a random vectors initialization for unseen words instead of adaptation.
- **NN++** integrates the word embeddings adaptation method proposed in section 5.

As we can see our strategy, which relies on a distributed representation of words to deal with OOV words, outperforms the CRF tagger, which had no access to external data. The gain is particularly significant when small amount of training data is available, but even when the full training corpus is used, we still observe improvements.

Adding POS and NE features improves performance (+1,25 F1-score on average for CRF++), especially for small corpora as it allows the CRF to generalize better on unseen data. Similarly we observe a very significant improvement from NN to NN++ by using our adaptation method. The embedding generation for words $w \notin C_{\text{emb}}$ leads to an average improvement of +3.34 F1-score. However, the initialization with embeddings learned on huge corpora only leads to improvements when a small amount of training data is available i.e when initialization process is highly relevant.

Focusing on the **OOE** and **OOV** accuracy, additional features increase the generalization of the subsequent models. POS and NE features help **OOV** recognition (2c) in the same way as **OOE** words adaptation fills the gap caused by mismatching resources (2b).


(a) F1 score

(b) **OOE**

(c) **OOV**

Figure 2: *F1-score and PWA (in %) restricted to* ***OOE*** *and* ***OOV*** *words as a function of the training corpus size. State-of-art baseline CRF tagger with and without additional features (POS, Named Entity) v.s. our proposed neural network model with and without adaptation strategy.*

Train	Model	F1-score	PWA	**OOV**	**OOE**
D_0	CRF	40.16	78.64	83.20	86.76
	CRF++	45.44	80.67	92.96	87.87
	NN– –	47.28	78.58	72.00	58.68
	NN	49.90	80.15	76.88	62.01
	NN++	54.15	82.17	91.20	91.91
D_1	CRF	56.60	82.67	80.77	90.24
	CRF++	58.13	83.84	92.97	91.73
	NN– –	67.91	86.18	90.82	72.16
	NN	68.41	86.52	90.82	72.99
	NN++	72.43	88.22	92.11	93.61
D_9	CRF	84.63	92.36	81.48	94.09
	CRF++	85.12	92.85	97.22	95.30
	NN– –	86.50	93.11	93.52	81.4
	NN	86.56	93.13	94.44	81.15
	NN++	90.14	94.87	95.37	95.92

Table 3: *Contrastive results when only small amount of training data is available (D_0, D_1) and when the full training corpus is used (D_9)*

7 Conclusion

This paper dealt with the particular problem of adapting a lexical embedding space to a specific SLU domain where a large number of application-specific terms do not have any representation in the initial vector space. We proposed to adapt lexical embeddings by creating accurate representations for unknown words: **OOV** words which do not occur in the SLU training data, nor in the embedding training data, and **OOE** words from the target domain which do not appear in the embedding training data. We showed on a semantic frame tagging task that our adaptation strategy improves over a state-of-the-art baseline using a CRF tagger when there is a mismatch between C_{emb} and the target corpus, especially when only a small amount of data is available to train the models.

References

[1] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," vol. abs/1301.3781, 2013.

[2] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *Advances in Neural Information Processing Systems 26*, C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 2787–2795.

[3] G. Tur, L. Deng, D. Hakkani-Tur, and X. He, "Towards deeper understanding deep convex networks for semantic utterance classification." IEEE International Confrence on Acoustics, Speech, and Signal Processing (ICASSP), March 2012. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=164624

[4] G. Mesnil, X. He, L. Deng, and Y. Bengio, "Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding." in *INTERSPEECH*, 2013, pp. 3771–3775.

[5] A. Celikyilmaz, D. Hakkani-Tur, P. Pasupat, and R. Sarikaya, "Enriching word embeddings using knowledge graph for semantic tagging in conversational dialog systems." AAAI - Association for the Advancement of Artificial Intelligence, January 2015. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=238362

[6] T. Schnabel and H. Schütze, "FLORS: Fast and Simple Domain Adaptation for Part-of-Speech Tagging," vol. 2, February 2014, pp. 15–26.

[7] C. D. Santos and B. Zadrozny, "Learning character-level representations for part-of-speech tagging," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, T. Jebara and E. P. Xing, Eds. JMLR Workshop and Conference Proceedings, 2014, pp. 1818–1826.

[8] S. Umansky-Pesin, R. Reichart, and A. Rappoport, "A multi-domain web-based algorithm for pos tagging of unknown words," in *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, ser. COLING '10. Association for Computational Linguistics, 2010, pp. 1274–1282.

[9] O. Owoputi, B. O'Connor, C. Dyer, K. Gimpel, N. Schneider, and N. A. Smith, "Improved part-of-speech tagging for online conversational text with word clusters." in *HLT-NAACL*, 2013, pp. 380–390.

[10] J. Andreas and D. Klein, "How much do word embeddings encode about syntax," in *Proceedings of ACL*, 2014.

[11] F. Bechet, B. Maza, N. Bigouroux, T. Bazillon, M. El-Beze, R. D. Mori, and E. Arbillot, "Decoda: a call-centre human-human spoken conversation corpus," in *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*. European Language Resources Association (ELRA), may 2012.

[12] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," in *the Journal of Machine Learning Research 12*, 2011, pp. 2461–2505.