Embedding Imputation with Grounded Language Information

Ziyi Yang¹, Chenguang Zhu², Vin Sachidananda³, and Eric Darve¹, ⁴

¹Department of Mechanical Engineering, Stanford University
²Microsoft Speech and Dialogue Research Group
³Department of Electrical Engineering, Stanford University
⁴Institute for Computational and Mathematical Engineering, Stanford University
{ziyi.yang, vsachi, darve}@stanford.edu, chezhu@microsoft.com

Abstract

Due to the ubiquitous use of embeddings as input representations for a wide range of natural language tasks, imputation of embeddings for rare and unseen words is a critical problem in language processing. Embedding imputation involves learning representations for rare or unseen words during the training of an embedding model, often in a post-hoc manner. In this paper, we propose an approach for embedding imputation which uses grounded information in the form of a knowledge graph. This is in contrast to existing approaches which typically make use of vector space properties or subword information. We propose an online method to construct a graph from grounded information and design an algorithm to map from the resulting graphical structure to the space of the pre-trained embeddings. Finally, we evaluate our approach on a range of rare and unseen word tasks across various domains and show that our model can learn better representations. For example, it improves Pearson’s and Spearman’s correlation coefficients on Card-660 task by 7.7% and 6.7% respectively.

1 Introduction

Word embeddings (Mikolov et al., 2013; Pennington et al., 2014) are used pervasively in deep learning for natural language processing. However, due to fixed vocabulary constraints in existing approaches to training word embeddings, it is difficult to learn representations for words which are rare or unseen during training. This is commonly referred to as the out-of-vocabulary (OOV) word problem. In the original embedding implementations, a special OOV token is typically reserved for such words. However, this rudimentary approach often detrims the performance of downstream tasks which contain numerous rare or unseen words. Recent works have proposed subword approaches (Zhao et al., 2018; Sennrich et al., 2015), which construct embeddings through the composition of characters or sentence pieces for OOV words. Vector space properties are also utilized to learn embeddings with small amounts of data (Bahdanau et al., 2017; Herbelot and Baroni, 2017). In this paper, we propose a novel approach, knowledge-graph-to-vector (KG2Vec), for the OOV word problem. KG2Vec makes use of the grounded language information in the form of a knowledge graph. Grounded information has been extensively used in various NLP tasks to represent real-world knowledge (Niles and Pease, 2003; Gruber, 1993; Guarino, 1998; de Bruijn et al., 2006; Paulheim, 2017). In particular, early question answering systems used expert-crafted ontologies in order to endow these systems with common knowledge (Harabagiu et al., 2005; Xu et al., 2016). Additionally, lexical-semantic ontologies, such as WordNet, have been used to provide semantic relations between words in a wide variety of language processing and inference tasks (Morris and Hirst, 1991; Ovchinnikova et al., 2010).

In particular, grounded language information has been observed to augment model performance on a wide variety of natural language processing and understanding tasks (He et al., 2017; Choi et al., 2018). In these settings, a model is able to provide better generalization by using relational information from a knowledge graph or knowledge base in addition to the standard set of training examples. Additionally, outputs from models with grounded approaches have been observed to be more factually consistent and logically sound (Bordes et al., 2014) compared with outputs from models without grounding information.

By foregoing the usage of vector space or subword information, KG2Vec is able to capture semantic meanings of words directly from the graph-
atical structure in grounded knowledge using recent advances in network representation learning. Furthermore, KG2Vec leverages the most updated information from comprehensive knowledge bases (Wikipedia & Wiktionary). Therefore, KG2Vec can be applied to training embeddings of newly emerging OOV words.

In summary, our contributions are three-fold:
1. An approach to constructing graphical representations of entities in a knowledge base in an unsupervised manner.
2. Methods for mapping entities from a graphical representation to the space in which a pre-trained embedding lies.
3. Experimentation on rare and unseen word datasets and a new state-of-art performance on Card-660 dataset.

2 Related Work

2.1 Graph Neural Networks

Graph neural networks (GNN) are an emerging deep learning approach for representation learning of graphical data (Xu et al., 2018; Kipf and Welling, 2016). GNNs can learn a representation vector $h_v$ for each node in the network by leveraging the graphical structure and node features $f_v$. Node embeddings are generated by recursively aggregating each node’s neighborhood information and features. At the $t$-th iteration, the information aggregation is defined as:

$$h_v^t = M^t(h_v^{t-1}, \{h_u^{t-1}\}_{u \in N(v)})$$

(1)

where $h_v^t$ is the representation for $v$ at the $t$-th iteration, $M^t$ is an iteration-specific message aggregation function parametrized by a neural network and $N(v)$ is the set of neighbors of node $v$. One simple form of $M^t$ is mean neighborhood aggregation:

$$h_v^t = \text{ReLU}(\sum_{u \in N(v)} \frac{W^t h_u^{t-1}}{|N(v)|} + B^t h_v^{t-1})$$

(2)

where $W^t$ and $B^t$ are trainable matrices. Typically, $h_v^0$ is initialized as $f_v$. The final node representation is usually a function of $h_v^T$ from the last iteration $T$, such as an identity function or a transformation function (Ying et al., 2018).

2.2 The OOV word problem

The out-of-vocabulary (OOV) word problem has been present in word embedding models since their inception (Mikolov et al., 2013; Pennington et al., 2014). Due to space and training data constraints, words which are either infrequent or do not appear in the training corpus can lack representations at the time of inference.

Numerous methods have been proposed to tackle the OOV word problem with a small amount of training data. Deep learning based approaches (Bahdanau et al., 2017) and vector-space based methods (Herbelot and Baroni, 2017) can improve the rare word representations on various semantic similarity tasks. One downside to these approaches is that they require small amounts of training data for words whose embeddings are being imputed and, as a result, can have difficulties representing words for which training samples do not exist.

Sub-word level representations have been studied in the context of the OOV word problem. Pi- ter et al. (2017) uses the RNN’s hidden state of the last sub-word in a word to produce representations. Zhao et al. (2018) proposes using character-level decomposition to produce embeddings for OOV words.

3 Model

We propose the knowledge-graph-to-vector (KG2Vec) model for building OOV word representations from knowledge base information. KG2Vec starts with building a knowledge graph $\mathcal{K}$ with nodes consisting of pre-trained words and OOV words. It then utilizes a graph convolutional network (GNN) to map graph nodes to low-dimensional embeddings. The GNN is trained to minimize the Euclidean distance between the node embeddings to pre-trained word embeddings in the dictionary such as GloVe (Pennington et al., 2014) and ConceptNet Numberbatch (Speer et al., 2017). Finally, the GNN is used to generate embeddings for OOV words.

3.1 Build the Knowledge Graph

In a knowledge graph $\mathcal{K}$, each node $v$ represents a word $w_v$. The nodes (words) in the graph are chosen as follows. We count the frequency of occurrences for English words from the Wikipedia English dataset (with 3B tokens). The 2000 words with the highest frequencies of occurrence are skipped to diminish the effect of stop words. Among the words left, we choose the $|V'|$ words with the highest frequencies of occurrence. All
OoV words for which we would like to impute embeddings are also added to the graph as nodes.

For each node, we obtain its grounded information from two sources: i. the words’ summary, defined as the first paragraph of the Wikipedia page when this word is searched; ii. the word’s definition in Wiktionary. We choose Wikipedia and Wiktionary over other knowledge bases because they are comprehensive, well-maintained and up-to-date. Here is an example of the grounded information for the word Brexit.

- Wikipedia page summary: Brexit, a portmanteau of “British” and “exit”, is the impending withdrawal of the United Kingdom (UK) from the European Union (EU). It follows the referendum of 23 June 2016 when 51.9 per cent of voters chose to leave the EU...
- Wiktionary definition: Brexit (Britain, politics) The withdrawal of the United Kingdom from the European Union.

All the words in the Wikipedia summary and the Wiktionary definition form the grounded language information of this word \( w_v \), defined as \( D_v \). Specifically, \( D_v \) is the concatenation of \( w_v \)'s Wikipedia summary and the Wiktionary definition. An undirected edge \( e_{vu} \) exists between node \( v \) and \( u \) if the Jaccard coefficient \( \frac{|D_v \cap D_u|}{|D_v \cup D_u|} > \eta \), where \( \eta \) is a pre-defined threshold and chosen to be 0.5 empirically in the experiments. The edge \( e_{vu} \) is then assigned with a weight \( s_{vu} = \frac{|D_v \cap D_u|}{|D_v \cup D_u|} \).

We also compute a feature vector \( f_v \) as the mean of pre-trained embeddings of words in \( D_v \). Finally, the obtained knowledge graph \( \mathcal{K} = (V, E) \) has a feature vector \( f_v \) for each node \( v \in V \).

3.2 Graph Neural Network

The nodes in the graph are mapped to low-dimensional embeddings via graph convolutional neural network (GCN) (Kipf and Welling, 2016). It follows that, at the \( t \)-th neighborhood aggregation, the node embedding \( h^t_v \) for node \( v \) is modelled as:

\[
h^t_v = \text{ReLU}(W^t \sum_{u \in S(v)} s_{vu}h^{t-1}_u + b^t)
\]

where \( S(v) = N(v) \cup \{v\} \), and the normalization constant \( C = 1 + \sum_{u \in N(v)} s_{vu} \). \( W^t \) and \( b^t \) are trainable parameters. The node embeddings are initialized as the feature vector \( f_v \), i.e. \( h^0_v = f_v \).

At the final iteration \( T \), the generated node embeddings \( \{h^T_v\} \) are computed without the ReLU function. The loss function of the GNN model is the mean square error between the pre-trained word vectors and generated embedding \( h^T_v \) for all words in the graph in the dictionary (e.g. GloVe). During inference, OoV words are assigned embeddings computed by the GNN.

4 Experiments

To evaluate our method’s ability to impute embeddings, we conduct experiments on the following rare and unseen word similarity tasks.

4.1 Card-660: Cambridge Rare Word Dataset

Card-660 (Pilehvar et al., 2018) is a word-word similarity task with 660 example pairs involving uncommon words and provides a benchmark for rare word representation models. Card-660 has a inter-annotator agreement (IAA) measure of 0.90, which is significantly higher than previous datasets for rare word representation. Additionally, Card-660 contains examples from a disparate set of domains such as technology, popular culture and medicine.

4.2 Stanford Rare Word (RW) Similarity

The Stanford Rare Word (RW) Similarity Benchmark (Luong et al., 2013) is a word-word semantic similarity task including 2034 word pairs and tests the ability of representation learning methods to capture the semantics of infrequent words. Due to the probabilistic underpinnings of word embeddings, where distances between two words’ representations are approximately proportional to their co-occurrence probability in a corpus, the authors found that rare words often have more noisy representations due to having fewer training samples. Although RW has a relatively low IAA measure of 0.41, the benchmark has been well-studied in previous literature.

4.3 Results

Experiment results, measured by Pearson’s and Spearman’s correlation, on the Card-660 and Stanford rare words datasets are shown in table 1. The Wikipedia pages and Wiktionary definitions used in the following experiments are snapshots on Feb 16th, 2019. We compare KG2Vec to other embedding imputation models, including Mimick (Pinter et al., 2017), Definition centroid (Herbelot and Baroni, 2017), Definition LSTM (Bah-
Table 1: Performance of OOV models on Stanford Rare Word Similarity and Card-660 datasets. Two word dictionaries are used: ConceptNet and GloVe. The overall best are underlined for each column, and the best results for each type of word dictionary are in bold. We rerun the BoS experiments with the default hyper-parameters from Zhao et al. (2018). Performances of other baseline models are collected from Pilehvar et al. (2018). 

During evaluation, zero vectors are assigned to missing words and word-word similarity is computed as the inner product of the corresponding embeddings. In KG2Vec, the number of iterations $T = 3$ for GCN, and the number of nodes with pre-trained word vectors $|V'| = 9000$. We test on two types of pre-trained word vectors GloVe (Common crawl, cased 300d) and ConceptNet Numberbatch (300d). KG2Vec shows competitive performance in all test cases. On Card-660 dataset KG2Vec achieves state-of-the-art results by a significant margin. When using ConceptNet embeddings, KG2Vec results in improvements of 7.7% and 6.7% on Pearson’s and Spearman’s correlation coefficients, respectively, when compared to prior state-of-the-art performance (BoS). When using GloVe embeddings, KG2Vec improves upon SemLand by 11% and 17.8% on Pearson’s and Spearman’s correlation coefficients. Considering the fact that Card-660 contains a significant amount of recent OOV words (e.g. “Brexit”), this improvement indicates that KG2Vec’s can leverage up-to-date information from knowledge bases. Additionally, this shows that GNN can effectively cover OOV words and precisely model their semantic meanings. On Stanford Rare Word dataset, KG2Vec is comparable with other state-of-the-art models, suggesting its robustness across various test schemes. Note that the graph used in KG2Vec has a much smaller size compared with knowledge graphs used in SemLand, the WordNet, which has 155,327 words.

To fairly evaluate KG2Vec, we include a baseline model that assigns the node feature $f_v$ as the final word representations for word $w_v$, if $w_v$ is not in the pre-trained dictionary. The results are denoted as “Node features” in table 1. In all test cases, KG2Vec improves by a large margin upon this baseline. For example, using GloVe on the Card-660 dataset, KG2Vec’s achieves a performance increase of 14.5% and 14.2% respectively for Pearson’s and Spearman’s coefficients over Node features. This observation suggests that the information aggregation by GNN is critical for embedding imputation and semantic inference. It also indicates that learning from the knowledge graph and its language information is an effective way to parse the semantic meaning of a rare word.

5 Discussion

Application on Entity Relations Knowledge Base. Many public knowledge bases consist of
relational data in a tuple format: (entity1, entity2, relation), where entities can be considered as the “nodes” in the graph and relations define the edges. Note that there are different kinds of relations and therefore edges in the graph have different types or labels. To impute the embeddings for entities in such scenario, one can conveniently adapt KG2Vec following Schlichtkrull et al. (2018) by learning different transformations for different types of edges.

Adaption to New Vocabularies and Information. Considering the fast growth of vocabularies and information nowadays, the ability of online learning and quick adaption for embedding imputations is desired. One can combine KG2Vec with meta-learning, e.g., MAML in Finn et al. (2017), such that the resulting model can quickly learn the embeddings of newly added nodes (words), or updated node features.

6 Conclusion and Future Work

In this paper, we introduce KG2Vec, a graph neural network based approach for embedding imputation of OOV words which makes use of grounded language information. Using publicly available information sources like Wikipedia and Wiktionary, KG2Vec can effectively impute embeddings for rare or unseen words. Experimental results show that KG2Vec achieves state-of-the-art results on the Card-660 dataset. Future research directions include a theoretical explanation of KG2Vec and applications to downstream NLP tasks.

Acknowledgments

We would like to thank anonymous reviewers for their valuable feedback.

References


Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In In EMNLP.


Kun Xu, Yansong Feng, Songfang Huang, and Dongyan Zhao. 2016. Hybrid question answering over knowledge base and free text. In COLING.