Fixed-Size Ordinally Forgetting Encoding Based Word Sense Disambiguation

Abstract

In this paper, we present our method of using fixed-size ordinally forgetting encoding (FOFE) to solve the word sense disambiguation (WSD) problem. FOFE enables us to encode variable-length sequence of words into a theoretically unique fixed-size representation that can be fed into a feed forward neural network (FFNN), while keeping the positional information between words. In our method, a FOFE-based FFNN is used to train a pseudo language model over unlabelled corpus, then the pre-trained language model is capable of abstracting the surrounding context of polyseme instances in labelled corpus into context embeddings. Next, we take advantage of these context embeddings towards WSD classification. We conducted experiments on several WSD data sets, which demonstrates that our proposed method can achieve comparable performance to that of the state-of-the-art approach at the expense of much lower computational cost.

Introduction

Words with multiple senses commonly exist in many languages. For example, the word bank can either mean a “financial establishment” or “the land alongside or sloping down to a river or lake”, based on different contexts. Such a word is called a “polyseme”. The task to identify the meaning of a polyseme in its surrounding context is called word sense disambiguation (WSD). Word sense disambiguation is a long-standing problem in natural language processing (NLP), and has broad applications in other NLP problems such as machine translation BIBREF0 . Lexical sample task and all-word task are the two main branches of WSD problem. The former focuses on only a pre-selected set of polysemes whereas the later intends to disambiguate every polyseme in the entire text. Numerous works have been devoted in WSD task, including supervised, unsupervised, semi-supervised and knowledge based learning BIBREF1 . Our

work focuses on using supervised learning to solve all-word WSD problem.

Most supervised approaches focus on extracting features from words in the context. Early approaches mostly depend on hand-crafted features. For example, IMS by BIBREF2 uses POS tags, surrounding words and collections of local words as features. These approaches are later improved by combining with word embedding features BIBREF0 , which better represents the words' semantic information in a real-value space. However, these methods neglect the valuable positional information between the words in the sequence BIBREF3 . The bi-directional Long-Short-Term-Memory (LSTM) approach by BIBREF3 provides one way to leverage the order of words. Recently, BIBREF4 improved the performance by pre-training a LSTM language model with a large unlabelled corpus, and using this model to generate sense vectors for further WSD predictions. However, LSTM significantly increases the computational complexity during the training process.

The development of the so called “fixed-size ordinally forgetting encoding” (FOFE) has enabled us to consider more efficient method. As firstly proposed in BIBREF5 , FOFE provides a way to encode the entire sequence of words of variable length into an almost unique fixed-size representation, while also retain the positional information for words in the sequence. FOFE has been applied to several NLP problems in the past, such as language model BIBREF5 , named entity recognition BIBREF6 , and word embedding BIBREF7 . The promising results demonstrated by the FOFE approach in these areas inspired us to apply FOFE in solving the WSD problem. In this paper, we will first describe how FOFE is used to encode sequence of any length into a fixed-size representation. Next, we elaborate on how a pseudo language model is trained with the FOFE encoding from unlabelled data for the purpose of context abstraction, and how a classifier for each polyseme is built from context abstractions of its labelled training data. Lastly, we provide the experiment results of our method on several WSD data sets to justify the equivalent performance as the state-of-the-art approach.

Fixed-size Ordinally Forgetting Encoding

The fact that human languages consist of variable-length sequence of words requires NLP models to be able to consume variable-length data. RNN/LSTM addresses this issue by recurrent connections, but such recurrence consequently increases the computational complexity. On the contrary, feed forward neural network (FFNN) has been widely adopted in many artificial intelligence problems due to its powerful modelling ability and fast computation, but is also limited by its requirement of fixed-size input. FOFE aims at encoding variable-length sequence of words into a fixed-size representation, which subsequently can be fed into an FFNN.

Given vocabulary INLINEFORM0 of size INLINEFORM1 , each word can be represented by a one-hot vector. FOFE can encode a sequence of words of any length using linear combination, with a forget factor to reflect the positional information. For a sequence of words INLINEFORM2 from V, let INLINEFORM3 denote the one-hot representation for the INLINEFORM4 word, then the FOFE code of S can be recursively obtained using following equation (set INLINEFORM5): INLINEFORM6

where INLINEFORM0 is a constant between 0 and 1, called forgetting factor. For example, assuming A, B, C are three words with one-hot vectors INLINEFORM1 , INLINEFORM2 , INLINEFORM3 respectively. The FOFE encoding from left to right for ABC is [INLINEFORM4 , INLINEFORM5 ,1] and for ABCBC is [INLINEFORM6 , INLINEFORM7 , INLINEFORM8]. It becomes evident that the FOFE code is in fixed size, which is equal to the size of the one-hot vector, regardless of the length of the sequence INLINEFORM9 .

The FOFE encoding has the property that the original sequence can be unequivocally recovered from the FOFE encoding. According to BIBREF5 , the uniqueness for the FOFE encoding of a sequence is confirmed by the following two theorems:

Theorem 1 If the forgetting factor INLINEFORM0 satisfies INLINEFORM1 , FOFE is unique for any sequence of finite length INLINEFORM2 and any countable vocabulary INLINEFORM3 .

Theorem 2 If the forgetting factor INLINEFORM0 satisfies INLINEFORM1 , FOFE is almost unique for any finite value of INLINEFORM2 and vocabulary INLINEFORM3 , except only a finite set of countable choices of INLINEFORM4 .

Even for situations described by Theorem SECREF2 where uniqueness is not strictly guaranteed, the probability for collision is extremely low in practice. Therefore, FOFE can be safely considered as an encoding mechanism that converts variable-length sequence into a fixed-size representation theoretically without any loss of information.

Methodology

The linguistic distribution hypothesis states that words that occur in close contexts should have similar meaning BIBREF8 . It implies that the particular sense of a polyseme is highly related to its surrounding context. Moreover, human decides the sense of a polyseme by firstly understanding its occurring context. Likewise, our proposed model has two stages, as shown in Figure FIGREF3 : training a FOFE-based pseudo language model that abstracts context as embeddings, and performing WSD classification over context embeddings.

FOFE-based Pseudo Language Model

A language model is trained with large unlabelled corpus by BIBREF4 in order to overcome the shortage of WSD training data. A language model represents the probability distribution of a given sequence of words, and it is commonly used in predicting the subsequent word given preceding sequence. BIBREF5

proposed a FOFE-based neural network language model by feeding FOFE code of preceding sequence into FFNN. WSD is different from language model in terms of that the sense prediction of a target word depends on its surrounding sequence rather than only preceding sequence. Hence, we build a pseudo language model that uses both preceding and succeeding sequence to accommodate the purpose of WSD tasks.

The preceding and succeeding sequences are separately converted into FOFE codes. As shown in Figure FIGREF3 , the words preceding the target word are encoded from left to right as the left FOFE code, and the words succeeding the target word are encoded from right to left as the right FOFE code. The forgetting factor that underlies the encoding direction reflects the reducing relevance of a word due to the increasing distance relative to the target word. Furthermore, the FOFE is scalable to higher orders by merging tailing partial FOFE codes. For example, a second order FOFE of sequence INLINEFORM0 can be obtained as INLINEFORM1 . Lastly, the left and right FOFE codes are concatenated into one single fixed-size vector, which can be fed into an FFNN as an input.

FFNN is constructed in fully-connected layers. Each layer receives values from previous layer as input, and produces values through a function over weighted input values as its output. FFNN increasingly abstracts the features of the data through the layers. As the pseudo language model is trained to predict the target word, the output layer is irrelevant to WSD task and hence can be discarded. However, the remaining layers still have learned the ability to generalize features from word to context during the training process. The values of the held-out layer (the second last layer) are extracted as context embedding, which provides a nice numerical abstraction of the surrounding context of a target word.

WSD Classification

Words with the same sense mostly appear in similar contexts, hence the context embeddings of their

contexts are supposed to be close in the embedding space. As the FOFE-based pseudo language model is capable of abstracting surrounding context for any target word as context embeddings, applying the language model on instances in annotated corpus produces context embeddings for senses.

A classifier can be built for each polyseme over the context embeddings of all its occurring contexts in the training corpus. When predict the sense of a polyseme, we similarly extract the context embedding from the context surrounding the predicting polyseme, and send it to the polyseme's classifier to decide the sense. If a classifier cannot be built for the predicting polyseme due to the lack of training instance, the first sense from the dictionary is used instead.

For example, word INLINEFORM0 has two senses INLINEFORM1 for INLINEFORM2 occurring in the training corpus, and each sense has INLINEFORM3 instances. The pseudo language model converts all the instances into context embeddings INLINEFORM4 for INLINEFORM5 , and these embeddings are used as training data to build a classifier for INLINEFORM6 . The classifier can then be used to predict the sense of an instance of INLINEFORM7 by taking the predicting context embedding INLINEFORM8 .

The context embeddings should fit most traditional classifiers, and the choice of classifier is empirical. BIBREF4 takes the average over context embeddings to construct sense embeddings INLINEFORM0 , and selects the sense whose sense embedding is closest to the predicting context embedding measured by cosine similarity. In practice, we found k-nearest neighbor (kNN) algorithm, which predicts the sense to be the majority of k nearest neighbors, produces better performance on the context embeddings produced by our FOFE-based pseudo language model.

Experiment

To evaluate the performance of our proposed model, we implemented our model using Tensorflow

BIBREF11 and conducted experiments on standard SemEval data that are labelled by senses from WordNet 3.0 BIBREF12 . We built the classifier using SemCor BIBREF13 as training corpus, and evaluated on Senseval2 BIBREF14 , and SemEval-2013 Task 12 BIBREF15 .

Experiment settings

When training our FOFE-based pseudo language model, we use Google1B BIBREF10 corpus as the training data, which consists of approximately 0.8 billion words. The 100,000 most frequent words in the corpus are chosen as the vocabulary. The dimension of word embedding is chosen to be 512. During the experiment, the best results are produced by the 3rd order pseudo language model. The concatenation of the left and right 3rd order FOFE codes leads to a dimension of 512 * 3 * 2 = 3072 for the FFNN's input layer. Then we append three hidden layers of dimension 4096. Additionally, we choose a constant forgetting factor INLINEFORM0 for the FOFE encoding and INLINEFORM1 for our k-nearest neighbor classifier.

Results

Table TABREF6 presents the micro F1 scores from different models. Note that we use a corpus with 0.8 billion words and vocabulary of 100,000 words when training the language model, comparing with BIBREF4 using 100 billion words and vocabulary of 1,000,000 words. The context abstraction using the language model is the most crucial step. The sizes of the training corpus and vocabulary significantly affect the performance of this process, and consequently the final WSD results. However, BIBREF4 did not publish the 100 billion words corpus used for training their LSTM language model.

Recently, BIBREF9 reimplemented the LSTM-based WSD classifier. The authors trained the language model with a smaller corpus Gigaword BIBREF16 of 2 billion words and vocabulary of 1 million words,

and reported the performance. Their published code also enabled us to train an LSTM model with the same data used in training our FOFE model, and compare the performances at the equivalent conditions.

Additionally, the bottleneck of the LSTM approach is the training speed. The training process of the LSTM model by BIBREF9 took approximately 4.5 months even after applying optimization of trimming sentences, while the training process of our FOFE-based model took around 3 days to produce the claimed results.

Conclusion

In this paper, we propose a new method for word sense disambiguation problem, which adopts the fixed-size ordinally forgetting encoding (FOFE) to convert variable-length context into almost unique fixed-size representation. A feed forward neural network pseudo language model is trained with FOFE codes of large unlabelled corpus, and used for abstracting the context embeddings of annotated instance to build a k-nearest neighbor classifier for every polyseme. Compared to the high computational cost induced by LSTM model, the fixed-size encoding by FOFE enables the usage of a simple feed forward neural network, which is not only much more efficient but also equivalently promising in numerical performance.