Learning Distributed Word Representations For

Bidirectional LSTM Recurrent Neural Network

Learning Distributed Word Representations For

Bidirectional LSTM Recurrent Neural Network

**Learning Distributed Word Representations for BLSTM-RNN**

BLSTM-RNN relies on the distributed

representation of words, which implies that

the former can be futhermore improved

through learning the latter better.

**BLSTM-RNN** (Bidirectional long short memory recurrent neural network) relies on the distributed representation of words, which implies that the former can be furthermore improved through learning the latter better.

the proposed

approach learns useful distributed word

representations, as the trained representations

signiﬁcantly elevate the performance of

BLSTM-RNN on three tagging tasks: part-of-

speech tagging, chunking and named entity

recognition, surpassing word representations

trained by other published methods.

Proposed approach learns useful distributed word representations, as the trained representations signiﬁcantly elevate the performance of BLSTM-RNN on three tagging tasks: part-of-speech tagging, chunking and named entity recognition, surpassing word representations trained by other published methods.

**How Distributed Word Representation work and do**

Distributed word representations represent word with a real valued vector, which is also referred to as word embedding. Well learned distributed word representations have been shown capable of capturing semantic and syntactic regularities and enhancing neural network model by being used as features.

Better trained word representations would

further improve the state-of-the-art performance

Better trained word representations would

further improve the state-of-the-art performance

Better trained word representations would further improve the performance.

**Sequence tagging** is a basic structure learning task for natural language processing. Many primary processing tasks over sentence such as word segmentation, named entity recognition and part-of-speech tagging can be formalized as a tagging task.

**Training Methods for Word Representation**

1. Matrix factorization (行列定义不同 从而分析)

Utilize low-rank approximation to decompose a large matrix that contains corpus statistics. But only uses the statistics of co-occurrence counts, disregarding of the position of word in sentence and word order.

Eg. LSA (latent semantic analysis), HAL (hyperspace analogue to language)

1. Windows-based
2. learn representations by training a neural
3. network model to make prediction within local con-
4. text windows

Learn representations by training a neural network model to make prediction within local context windows. But only consider local context, which is incapable of involving information outside the context window.

can also be obtained by train-

ing recurrent neural network (RNN) language model

proposed by (Mikolov et al., 2010) or GloVe model

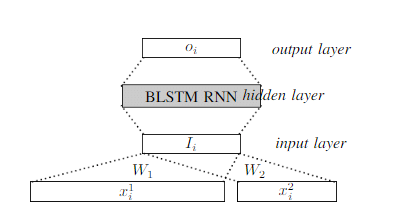
proposed by (Pennington et al., 2014a) which trains

a log-bilinear model on word-word co-occurrence

counts.

Can also be obtained by training RNN language model proposed by or GloVe model (might limit the size of the text window) proposed by which trains a log-bilinear model on word-word co-occurrence counts. But theoretically considers all information of the previous sequence but fails to involve the information of the posterior sequence.

**Model Structure of BLSTM-RNN**

 ***Ii=W1\*xi + W2\*xi2***

**Learning Representation of BLSTM-RNN**

BLSTM-RNN is applied to per-

form a tagging task with only two types of tags to

predict: incorrect/correct.

* BLSTM-RNN is applied to per-form a tagging task with only two types of tags to predict: incorrect/correct.
* The input is a sequence of words which is a normal sentence with several words replaced by words randomly chosen from vocabulary.

***Noted that***: if words are replaced, tag them as 0 (incorrect), if words are not replaced, tag them as 1 (correct)

learns represen-

tations for a feedforward network and (Gutmann and

Hyv¨

arinen, 2012; Mnih and Teh, 2012; Vaswani et

al., 2013) learns normalization parameters instead of

representations

* learns representations for a feedforward network and learns normalization parameters instead of representations

**Experimental setup**

Consecutive digits occurring within a word

are replaced with the symbol “#”

1. Consecutive digits occurring within a word are replaced with the symbol “#” (eg. aab11, aab2 both change to aab#)
2. The threshold to determine whether a word is replaced is 0.2, which means about 20% tokens in corpus are re-placed with tokens randomly selected from vocabulary.
3. BLSTM-RNN is implemented based on CURRENNT (open source GPU-based toolkit of BLSTM-RNN). The dimension of word representation as well as input layer size of BLSTM-RNN is 100 and hidden layer size is 128.
4. Train the Skip-gram and CBOW model using the word2vec tool with a con-text window size of 10 and 10 negative samples. For training GloVe, we use the GloVe tool with a context window size 10.

**Evaluation**

Focus on the effect of word representation, for all tasks, use the network with the same hidden structure and input features.

BLSTMWE (BLSTM Word Embedding) trained by approach get the best performance. CBOW, Skip and GloVe enhance the performance of BLSTM-RNN as well.