CDSMs for Semantic Relatedness and Entailment

Sidharth Gupta
sidharg@iitk.ac.in
IIT Kanpur

Sai Krishna Prasad
ssai@iitk.ac.in
IIT Kanpur

Amitabha Mukerjee
amit@cse.iitk.ac.in
IIT Kanpur

April 24, 2014

Abstract

Distributional Semantics Models (DSMs) have become widely accepted as successful models for lexical semantics. However their extension to handling larger structural units such as entire sentences remains challenging. Compositional DSMs (CDSMs) aim to successfully model sentence semantics by taking into account grammatical structure and logical words, which are ignored by simpler models. We explore a recursive matrix-vector space model, where each word or phrase has associated with it a vector capturing its semantics, as well as a matrix capturing how it alters the meanings of other words or phrases in its vicinity. We proceed to test this proposed CDSM on the tasks of semantic relatedness score prediction and semantic entailment classification, over the SICK data set of approximately 10,000 sentence pairs.

1 Introduction

Semantics is essentially the study of meaning, and how it may be gleaned from language. Distributional semantic models (DSMs) have been widely applied to understand the semantics of words or lexemes. DSMs are based on the Distributional Hypothesis which essentially states that "words that are used and occur in the same contexts tend to purport similar meanings" [4]. Or as explained by Firth, "a word is characterized by the company it keeps".

DSMs approximate the meanings of words by studying the distribution of the word across different contexts in the given training corpus. This distribution, learned in an unsupervised manner, characterises lexical semantics and is specified for every word in the form of a high dimensional vector. Semantic similarity at the lexical level may then be modelled as distances between these semantics vectors.

Although DSMs have found a number of successful applications, their extension to model the semantics of larger grammatical structures such as phrases or sentences has proved challenging. Simple approaches, such as modelling the semantic vector for a sentence as a weighted sum of the semantic vectors for it's constituent words, give poor results failing to account for grammatical structure and 'operator words' (words like adverbs and adjectives which alter the behavior of other words in their neighborhood).

Compositional DSMs (CDSMs) seek to overcome these limitations and successfully model sentence semantics. In this project, we seek to apply CDSMs to the tasks of semantic relatedness and entailment, both of require an underlying understanding of semantics. This is the first task of SemEval-2014. [1]

2 Related Work

In their work Grefenstette's and Sadrzadeh[3] implement a compositional DSM that makes use of Lambek pregroup grammars, training it over the entire BNC. The evaluation is based on the word disambiguation task developed by Mitchell and Lapata[5], and the results obtained match or better those of other competitors. They take an unsupervised learning approach to learn the matrices corresponding to relational words (suitably identified) as well as the distribution vectors corresponding to other words. Relational words are modeled as matrices to allow them to act on the vectors corresponding to the semantics of other words, thus modelling how 'operator words' alter the meanings of other words. This is needed while composing the semantics of the sentence as a whole, which is a function (linear map) of the Kronecker product of the word vectors.

However our work is primarily based on the recursive matrix-vector spaces model proposed by Socher, Huval, Manning and Ng[6]. In this model each word has associated with it a vector and a matrix. The vector captures the semantics of the word itself and is obtained from the underlying Distributional Semantics Model. The matrix captures how the word can alter the semantics of other words in its neighborhood, essentially approximating the effects the effects of 'operator words' on semantics.

The authors then outline a two step procedure for evaluating sentence semantics:

1. Build the parse tree for the given sentence

2. Recursively combine the words according to the syntactic structure

of the parse tree, proceeding in a bottom up manner to obtain the semantic representations for the entire sentence

The evaluation of the matrix and the vector at each combination step in the recursive procedure is done in such a way that the dimensions of both are preserved, making the bottom up recursive approach to combination feasible.

3 Dataset and Task Description

We have chosen the first task of SemEval-2014, which has two subtasks - semantic relatedness and semantic entailment prediction. These need to be performed over the SICK (Sentences Involving Compositional Knowledge) dataset, specifically designed for this challenge.

3.1 Dataset

- The dataset consists of a little under 10,000 sentence pairs hand labeled with semantic similarity scores (on a scale of 1 to 5) and the nature of semantic entailment (entailment, contradiction, or other) between them.

- As mentioned in the task guidelines, the SICK dataset is specifically designed to include "sentence pairs that are rich in the lexical, syntactic and semantic phenomena that CDSMs are expected to account for, but do not require dealing with other aspects of existing sentential data sets that are outside the domain of CDSMs (such as multiword expressions, named entities and telegraphic language)."

- A sample sentence pair from the dataset would look like: Sentence A: A man in a black jacket is doing tricks on a motorbike. Sentence B: A person in a black jacket is doing tricks on a motorbike. Relatedness Score: 4.9 (indicates that the two sentences are highly similar) Entailment Relationship: Entailment (Sentence A entails Sentence B)

3.2 Semantic Relatedness Score Prediction

- To predict the semantic relatedness score for 500 sentence pairs from the dataset.

- We estimate the performance of our model by comparing our predicted scores, with the actual scores (which are known as the data is labeled).

3.3 Semantic Entailment Classification

- To predict the semantic entailment relationship for the same 500 sentence pairs as used above.
- Once again performance is measured by comparing our predicted relationship with the actual one.

4 Approach

We took a two step approach to solve each sub-task:

1. Obtain the vectors representing the semantics of all sentence pairs (training, validation and test sets) using our chosen CDSM model.
2. Applying appropriate regression and classification techniques to predict the semantic similarity score and semantic entailment relationship respectively.

4.1 Obtaining Sentence Semantics Vectors

Socher provides code for this model to solve the problem of classifying relations between words in a sentence[6]. We have suitably modified this code for obtaining the required sentence semantics vectors. Socher's implementation uses of the Stanford Parser[2] to obtain the required parse trees.

4.2 Predicting Semantic Similarity Scores

Regression techniques are used to estimate the relatedness score between the sentence pairs in the test set. The regression model was trained using the samples for the training set, whose sentence semantics vectors are calculated above, and whose similarity scores are known.We have explored two techniques - logistic regression and neural networks.

4.3 Predicting Semantic Entailment Relationship

Classification techniques are used to predict the nature of the semantic entailment relationship between the sentence pairs in the test set. Once again we train the classification model using labeled samples from the training set. We have explored the technique of neural networks for this purpose.

5 Results

5.1 Semantic Similarity

5.1.1 Logistic Regression

For logistic regression we divided the data into the following two parts

- Training set = 9427 samples
- Test Set = 500 samples

The hypothesis function was calculated without using regularisation. The mean of the absolute difference between the actual and predicted semantic similarity scores over the test set = 2.96

5.1.2 Neural Networks

For neural networks the data was divided into the following three parts:

- Training set = 7070 samples
- Validation Set = 1885 samples
- Test Set = 500 samples

Here we kept the test set fixed and from the remaining samples, the validation and the training set are chosen at random. We used a neural network consisting of **one hidden layer containing 200 neurons**. The weights of the neural net converged after 15 iterations, as no further reduction in error was observed over any of the three sets.
The mean of the absolute difference between the actual and predicted semantic similarity scores over the test set = 0.71

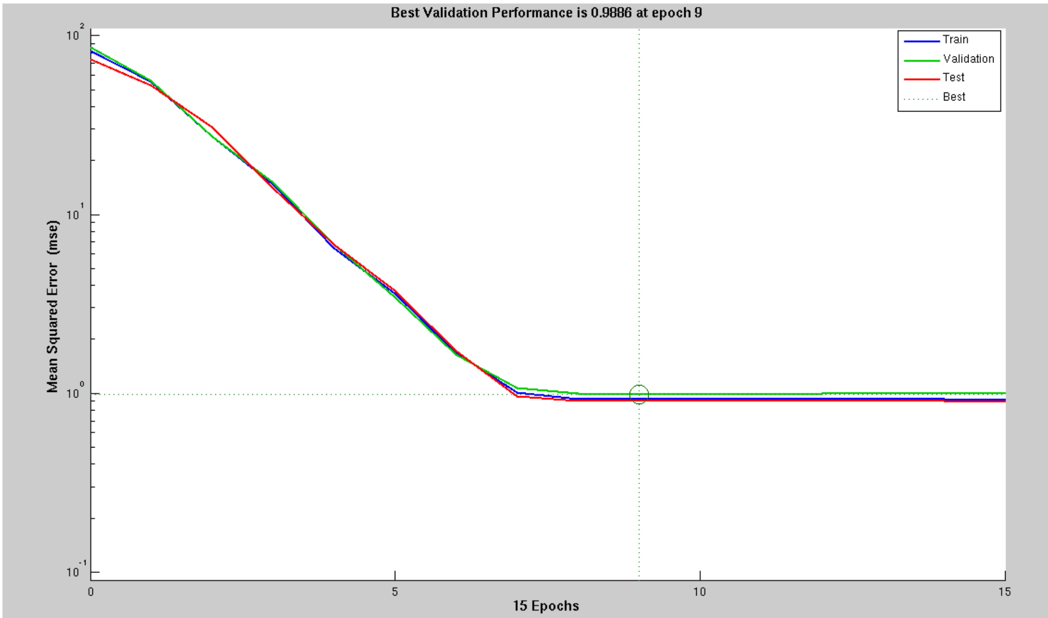

Figure 1: Plot of error vs. number of epochs

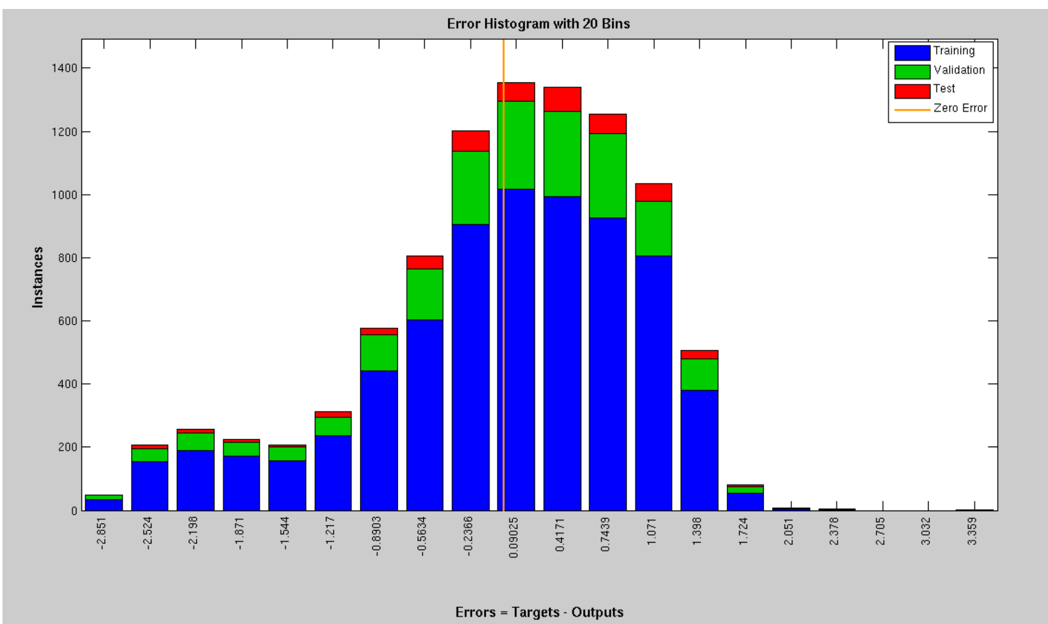

Figure 2: Error histogram

5.2 Semantic Entailment

Semantic Entailment relationship prediction over the test set is carried out using a trained neural net. The division of the data into training, validation and test sets is done as discussed in **Section 5.1.2** The neural network makes use of a **single hidden input layer of 700 neurons**, as seen in the image below:

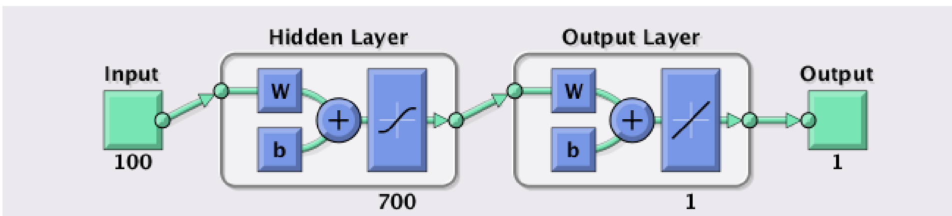

Figure 3: Neural Network

The classification accuracy obtained over the test data set of 500 samples = 67.3%

6 Conclusion

In this project we explored the limitations of DSMs in modelling the semantics of entire sentences, in spite of their success in modelling lexical semantics. We introduce the idea of CDSMs that seek to overcome these limitations, and successfully model sentence semantics by accounting for grammatical structure. We explore competing CDSMs based on the ideas of Lambek pregroup grammars and recursive matrix-vector spaces. We proceed to modify an existing implementation of the matrix-vector spaces based model, to test the performance of this model over the tasks of semantic relatedness and semantic entailment prediction. The results we obtain are promising for a first attempt to fit this model to these specific tasks, and by varying the many underlying parameters, we can hope for incremental improvements in performance.

7 Future Work

- The sentence semantics vectors produced by our modified version of Socher's code at times produces very similar vectors for loosely related sentences. We could explore whether this can be overcome by changing the underlying DSM or the non linear function used to combine semantics vectors. Greater differentiation between such vectors would allow for better regression and classification performance.

- Alternately we could explore whether the use of classification and regression that make use of deep belief networks would produce better results, than the single hidden layer neural networks used by us.
- Lastly we could explore whether the use of further linguistic resources, beyond the Stanford Parser, such as WordNet or POS could be used to boost performance over the tasks discussed.

References

[1] Semeval'14 task 1:http://alt.qcri.org/semeval2014/task1/. 2014.

[2] Stanford natural language processing parser: http://nlp.stanford.edu/software/lex-parser.shtml. 2014.

[3] Edward Grefenstette and Mehrnoosh Sadrzadeh. Experimental support for a categorical compositional distributional model of meaning. *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, 2011.

[4] Zellig Harris. Distributional structure. *Word*, 10(23):146–162, 1954.

[5] Jeff Mitchell and Mirella Lapata. Composition in distributional models of semantics. *Cognitive Science*, 2010. To appear.

[6] Richard Socher, Brody Huval, Christopher D. Manning, and Andrew Y. Ng. Semantic Compositionality Through Recursive Matrix-Vector Spaces. In *Proceedings of the 2012 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2012.