Review: Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank

Anindya Sundar Das

This article is a brief review of the research paper (Socher et al., 2013) in which the authors proposed an efficient, novel approach that focuses on grammatical structure of a sentence for fine-grained sentiment analysis.

The paper discusses various compositional methods to combine words and phrases (n-gram) to predict the binary (positive or negative) as well as fine-grained (very positive, positive, neutral, negative, very negative) sentiments of words, phrases and whole sentence in a bottom-up fashion. **The main contribution of this paper is to introduce a parse tree based dataset with fine-grained sentiment labels: "Stanford Sentiment Treebank" and proposes a neural compositional model: Recursive Neural Tensor Network (RNTN) that outperforms all previous recursive models and achieves state-of-the-art performance.**

Dataset: Stanford Sentiment Treebank

The dataset was created by parsing $11,855$ sentences of a movie review excerpt corpus with Stanford Parser, resulting 215,154 phrases which were then randomly sampled and labelled into 25 values (Figure 1) using Amazon Mechanical Turk. It is observed that shorter phrases have neutral sentiments while more polarized sentiments are being noticed in longer phrases. Also it has been observed based on annotators grading on slider scale, a 5-class classification is enough to capture the major variabilities.

The treebank dataset facilitates creating efficient models that can predict the polarity of short sentences and classify difficult negation examples which was not attainable by previous bag-of-words approaches which ignore the word orders in a sentence. Also the binary (positive or negative) classification accuracy on sentiment analysis task crossed 80% mark for the first time after introduction of treebank.

RNTN: Recursive Neural Tensor Network

The authors first discusses the compositional methods used by recursive models such as **Recursive Neural Network(RNN)** and **Matrix-Vector Recursive Neural Network (MV-RNN)** to predict sentiments of n-gram phrases and their limitations and then proposes **RNTN** which overcomes the limitations and outperforms all previous models on this task.

All the recursive models parse the input n-gram into a binary tree, with constituent words as leaves. These words are represented by d-dimensional vectors. The Embedding matrix $L \in \mathbb{R}^{d \times |V|}$ of word vectors ($|V|$ is size of vocabulary) is trained jointly with the models. These word vectors are used to predict the sentiment on word level. Then the recursive models compute the parent vectors (d-dimensional) in a bottom up fashion (Figure 2) using various compositional methods, after all of its children vectors are computed. The parent vector at each node is used as input to $softmax$ classifier to compute class probabilities at that node.

Though **RNN** uses single composition function to compute n-gram vector for phrases at each node, the input vectors interact with each other through a non-linearity ($tanh$ activation). A more robust and direct interaction between the input vectors is desired, which is achieved in **MV-RNN** in which each n-gram phrase ($n >= 1$) is represented by a vector and a matrix. Word vectors and word matrices are the parameters of MV-RNN which are learned during model training, hence with increase in vocabulary size,

Figure 1: Slider for selecting sentiments in range 0-25

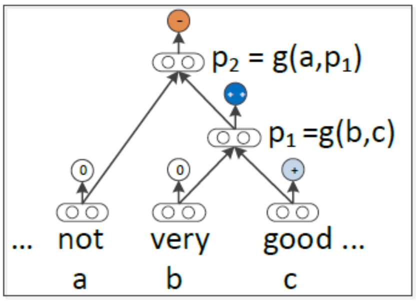

Figure 2: Recursive models computing parent vector in a bottom up approach

the number of parameters becomes very large.

RNTN overcomes these limitation of RNN and MV-RNN; It has lesser number of fixed parameters compared to MV-RNN and uses more powerful and single composition function for all nodes; the input vectors interact explicitly in RNTN unlike standard RNN.

Insights

The paper offered several important insights and observations:

- Models were compared with Naive Bayes, SVMs, BiNB (NB with bigram features), VecAvg(average of word vectors). On fine-grained classification for all phrases (at all node levels of the parse trees) RNTN achieves best performance, followed by MV-RNN, RNN and other models. For binary classification on sentence level, RNTN pushes state of the art accuracy from 80% to 85.4% .

- Optimal performances for all the models were achieved for word vector dimension between 25 and 35, performance deteriorates for smaller and larger value of word vectors which confirms RNTN performance enhancement is not dependent on its increased parameter size as MV-RNN has largest number of parameters.

- RNTN reasonably captures the effect of Contrastive Conjunction ('*but*') on overall sentiment of the sentence.

- RNTN also captures the effect of negation in both positive and negative sentences. It has highest accuracy for negating the positive sentences; it also increases non-negative activation (degree of non-negative sentiment in a sentence) for negation of negative sentence cases, which clearly indicates the model learns the negation concept well beyond simple negation rules.

Conclusion

RNTN model is powerful in capturing the structural composition of the words and phrases in a sentence and learning the effect of composition in detecting sentiments in a principled and efficient way. The treebank dataset captures intricacies of linguistic phenomena; all models show substantial improvement on their performances when trained on this new dataset. However, it is to be noted that as RNTN requires the parse tree of the input sentences to be constructed; the model might not perform well in cases of poor grammatical constructions such as dialogues in chatbots or tweets. Another interesting case would be, to observe the effect of pre-trained word embeddings such as word2vec, glove, fasttext on over all performance of the model instead of learning the word vector embeddings as parameters during training.

References

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.