The State of the Art on Text Classification Technology

Name: Jialong Yin E-main: jialong2@illinois.edu

1. Introduction

Text classification is a classical problem in natural language processing (NLP), aiming to assign labels to textual units such as sentences, paragraphs and documents. It has a wide range of applications such as sentiment analysis, spam detection, news categorization, question answering and so on. Text data come from different sources including web data, social media, user reviews and many more. Extracting information from text data is challenging due to its unstructured nature. In this review, we explore some models of the state of the art for text classification in the architecture of feed-forward neural networks, RNN-based models and CNN-based models.

2. Models

2.1 Feed-Forward Neural Networks

Feed-forward networks are simple deep learning models for text representation but still achieve a high accuracy on many text classification benchmarks. Text is viewed as a bag of words in these models. They use an embedding model to learn a vector representation for each word such as word2vec [8] or Glove [9]. The vector sum or average of the embeddings is taken as the representation of the text, and passed through one or more feed-forward layers, known as Multi-Layer Perceptrons (MLPs). And classification is performed on the final layer's representation using a classifier such as logistic regression or SVM. Deep Average Network (DAN) is one of the examples [10], as shown in Fig. 1. DAN can outperform many other sophisticated models despite its simplicity.

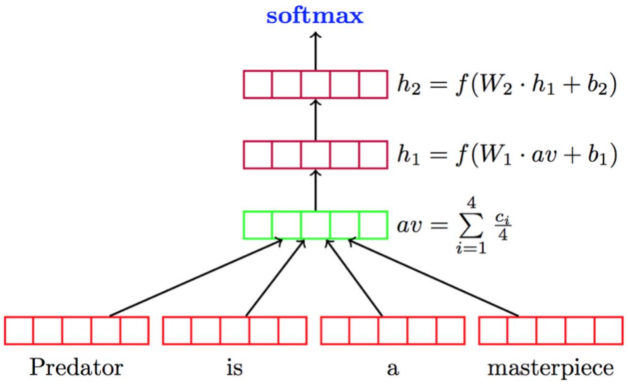

Figure 1. The architecture of the Deep Average Network (DAN).

2.2 RNN-Based Models

RNN-based models view text as a sequence of words, and capture word dependencies and text structures for text classification. However, vanilla RNN models do not work well, and often have worse performance than feedforward neural networks. Long Short-Term Memory (LSTM) is one of the most popular architectures among many variants of RNNs, which is designed to better capture long term dependencies. LSTM addresses the gradient vanishing or exploding problems suffered by the vanilla RNNs by introducing a memory cell to remember values over arbitrary time intervals, and three gates (input gate, output gate, forget gate) to regulate the flow of

information into and out of the cell. Many works have been done on improving RNNs and LSTM models for text classification to capture richer information, such as document topics, long-span word relations in text and so on.

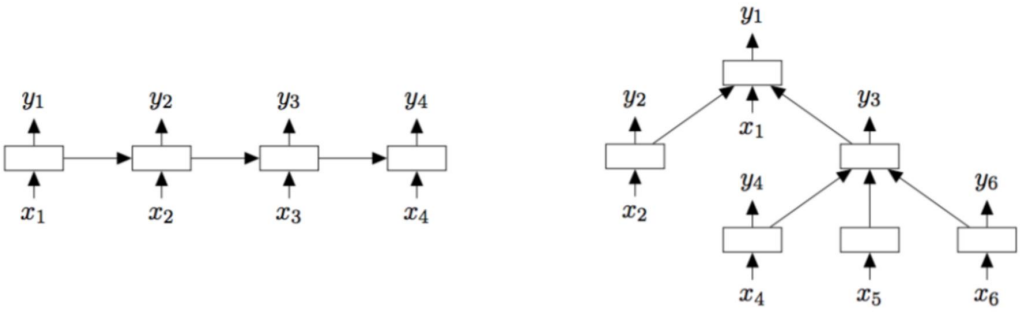

Figure 2. (Left) A chain-structured LSTM network and (right) a tree-structured LSTM network with arbitrary branching factor.

Tai et al. [15] propose a Tree-LSTM model, which is a generalization of LSTM to tree-structured network, to learn rich semantic representations. They state that Tree-LSTM has better performance than chain-structured LSTM for NLP tasks because some syntactic properties exits in natural language, for example, naturally combine words to phrases. The effectiveness of Tree-LSTM is validated on two tasks: sentiment classification and semantic relationship prediction. The architectures of these models are shown in Fig. 2.

2.3 CNN-Based Models

RNNs are trained to recognize patterns across time and work well for NLP tasks which require long-range semantics, while CNNs learn to recognize patterns across space [25] and work well when detecting local and position-invariant patterns is important. These patterns can be critical phrases that express a particular sentiment to be extracted by CNNs.

Kim [27] develop a CNN-based model for text classification. As shown in Fig. 3, Kim's model perform a convolutional layer on top of word vectors which is obtained from an unsupervised language model i.e., Glove. Kim also compared four different approaches to learning word embeddings: (1) CNN-random, where all word embeddings are randomly initialized and then modified during training; (2) CNN-static, where the pre-trained word2vec embeddings are used and stay fixed during model training; (3) CNN-non-static, where the word2vec embeddings are fine-tuned during training for each task; and (4) CNN multi-channel, where two sets of word embedding vectors are used, both are initialized using word2vec, with One updated during model training while the other fixed. These CNN-based models are reported to improve the state-of-the-art on sentiment analysis and question classification.

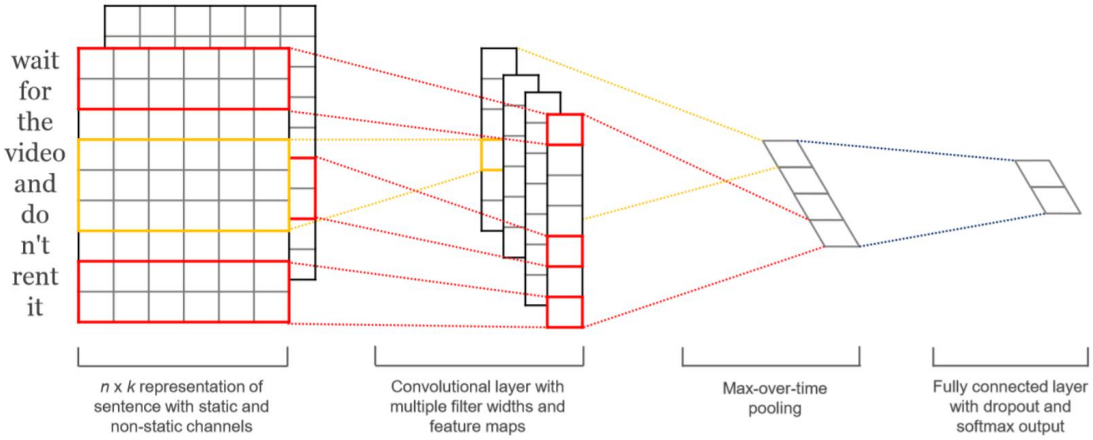

Figure 3. The architecture of a sample CNN model for text classification.

3. Conclusion

In this review, we explore some models in the architectures of feed-forward neural networks, RNN-based models and CNN-based models. They all have achieved high accuracy in text classification tasks.