Convolutional Neural Networks for Sentence Classification

Authors: Yoon Kim
Publisher: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)
Year: 2014

Keywords: CNN, semantic analysis, NLP

1. **Abstract:**

"We report on a series of experiments with convolutional neural networks (CNN) trained on top of pre-trained word vectors for sentence-level classification tasks. We show that a simple CNN with little hyperparameter tuning and static vectors achieves excellent results on multiple benchmarks. Learning task-specific vectors through fine-tuning offers further gains in performance. We additionally propose a simple modification to the architecture to allow for the use of both task-specific and static vectors. The CNN models discussed herein improve upon the state of the art on 4 out of 7 tasks, which include sentiment analysis and question classification."

2. **Introduction:**
   - The motivation of the paper is sentiment classification with simple architecture of CNN.
   - This neural net training with just a bit hyper parameter tuning had been giving significant results for all the benchmark datasets.
   - The training is done on the pre trained word vectors for the purpose of sentiment classification.
   - This is a very simple phenomenon which results in the extraordinary results through the approach.

3. **Related Works:**
   - Previously, the application of CNNs in computer vision localization had been conducted and the same approach has been undertaken into NLP for the purpose of semantic parsing.
   - A model which had been trained by Milkov et al. on billions of words from google news had been used by the author to train on above the word vectors.
   - Apart from those, the work has been very relevant to the Razavian et al.(2014) approach which stated that the feature extractors from the pretrained model for image classification had shown the significant results in various applications.
4. **Content and Approach:**

- Let us assume a word x_i from the sentence of n words representing a k dimensional vector and the concatenation of words from x_1 to x_n result in $x_{1:n}$.
- Similarly a symbol $x_{i:i+j}$ represent the concatenation of words,resulting in the feature as follows:
    - $C_i = f(w. X_{i:i+h-1} + b)$, where w refers to the weight matrix with (h x k) order.
- The above function is a non-linear hyperbolic tangent function. By extracting a set of features, we represent a feature map *c*.
- Similarly, multiple layers are formed with multiple features leading to the penultimate layer, which works on the softmax function resulting in the maximum probability distribution.
- A max-over-pooling operation is done on the feature map to obtain the maximal feature that corresponds the sentence.The pooling operations can be applied to the variable lengths of sentences.
- Two channels are created for the word vectors in which one is fixed static throughout the total training process and the other is fine-tuned with the help of backpropogation.
- Later dropout is carried out for the words based on the weight vectors' L_2 norm. We then calculate the labels as follows:
    - $y = w.(z \circ r) + b$
      Where w refers to the weight vectors,
      z refers to the penultimate layer referring the features from pretrained model
      r is the masking vector with the probability of 1 obtained from the Bernoulli's random variables
      b is the bias and $\circ$ refers to element wise multiplication.
- The gradients in backpropogation were applied only on the unmasked units of the feature vector.
- The experiments had been done on various datasets and applications. The datasets and their applications are referred below:
    - **Movie Review(MR):** The classification of review is either positive or negative, in which only a single sentence is taken into consideration per review.
    - **Stanford Sentiment Treebank(SST-1):** It is similar to the movie reviews but it has multiple degrees of positiveness as very positive, positive, neutral, negative and very negative.
    - **SST-2**: It is similar to the SST-1, but the neutral reviews were eliminated and the labels were classified into binary elements.
    - **Subjectivity dataset(Subj):** The classification is done based on the subjective basis to define whether the sentence is subjective or objective.

    - **TREC:** This task refers to the classification of questions into 6 types based on the question to define whether the question refers to the person, number or a thing etc.,
    - **Customer Reviews(CR):** To classify the customer review into positive or negative.
    - **MPQA:** To detect the opinion polarity of the dataset.
- The datasets are trained with ReLUs and with a dropout rate of 0.5, batch size of 50 and l_2 constraint of 3.
- Experiments on different datasets were also carried out by different model variations such as CNN-rand, CNN-static, CNN-non-static and CNN-multi channel.

5. Results:

- The model variant of CNN-rand has not achieved better results normally. But when the pretrained vectors were used the performance had been significantly increased.
- The application of pre trained vectors on static and non-static also increased the performance measure of the neural networks.
- From these results, the authors stated that the feature extractors from the pretrained vectors had been very favourable to gain better results.
- In the comparison between the single channel and the multichannel models, the results had mixed up unlike the perception of the author that the multichannel would avoid overfitting.
- The comparison is also done in between static and non-static channels, which resulted in the fact that the static channel represented the syntactic equivalents most of the time and the other referred to the semantic orientation based on the norm between the vectors.

6. Conclusion:

- From the results, it is evident that the pretrianing CNN on top of the word vectors would result in the better performance just with a little tuning of the hyper parameters.
- The author had done a remarkable job to increase the performance just with a single layer of CNN, simple to implement.
- It can be concluded that the supervised pre training is an critical factor in the deep NLP.