Distributed Representations of Sentences and Documents

Link

arxiv

Summary

- Bag-of-word features lack word ordering or semantics of the words. Bag-of-n-grams considers word order to some extent but they suffer from data sparsity and high dimensionality and also lacks semantics. Word vectors on the other hand can capture semantic information(similar words have similar vector representations). In this paper they propose a **Paragraph Vector** that can learn vector representations for pieces of texts. The texts can be of variable-length, ranging from sentences to documents.
- They use two approach to learn paragraph vectors

    1. Distributed memory model: This is similar to how word vectors are trained. We first extract fix length contexts using sliding window over the paragraph. The paragraph vector is shared across all contexts generated from the same paragraph but not across paragraphs. Word vectors are shared across paragraphs. The paragraph vector and word vectors are concatenated and used to predict the next word in context and trained via back propagation using SGD. During inference we freeze the vector representation for each word, and learn the representations for the sentences using gradient descent.
    2. Distributed bag of words: In this approach the paragraph vector is trained to predict random word selected from the context one at a time. So this just works like bag-of-words and ignores word orders. We don't have to train word vectors in this approach.

    During experiments we combine the two representations to learn the paragraph vector. First approach works much better than the second approach if used separately but combining them improved and stabilized performance.

- Using the paragraph vector with a logistic regression classifier achieved state of the arts result on sentiment analysis task on movie review datasets(IMDB and rotten tomatoes). The main advantage of this is the same approach can be used across sentences, paragraphs and documents without requiring any parsing.
- Paragraph vector also performed significantly better than bag-of-words and bigrams on information retrieval task which required the model to identify which two paragraphs were returned from same search query and which two from different search query by measuring their distance(paragraphs A and B obtained from same query, paragraph C obtained from a different query)