CS5339 Project Proposal

Paper to Study: *Distributed Representations of Sentences and Documents*[1]
Group Members: Tang Yijie (A0212491M), Zhang Hao(A0210996X)

1 Background

For many common machine learning tasks related to text processing such as text classification or clustering, usually a fixed-length input is required, such as *bag-of-words*, *bag-of-n-gram*. However, *bag-of-words* representation loses the order property in sentences, yet *bag-of-n-gram* preserves some order information but it suffers from data sparsity and high dimensionality. Another representation such as word-vector was also proposed (see Appendix).

2 Problem Statement

Those methods mentioned in backgrounds, especially *bag-of-words* and *bag-of-n-gram*, make little sense to capture semantic information of raw text, which limits the performance of tasks such as sentiment analysis. To mitigate the disadvantages of the representation of the text mentioned above, the authors of the paper proposed a new representation that can retain the semantic information of the original text. The proposed method in this paper, Paragraph Vector, is an unsupervised learning method that can learn the consistency and semantics of words in variable-length texts.

3 Contribution

One key advantage of the *Paragraph Vector* compared to previous representations is its capability of constructing representations of variable-length input sequences. Therefore, it can be used on sentences, paragraphs or documents of any length. The paper stated that they achieved new state-of-the-art results on sentiment analysis tasks with the new representation and it outperformed other more complex methods. Two frameworks were proposed:

1. **Paragraph Vector: A distributed memory model (PV-DM)**
   This method computes the target word from context by concatenating the word vectors of context and paragraph vectors. If the paragraph matrix is D, the only change in this method comparing to *word-vector* framework is the equation (1) (see Appendix), where h is constructed from both W and D

$$y = b + Uh(w_{t-k}, ..., w_{t+k}; W, D) \quad (2)$$

2. **Paragraph Vector without word ordering: Distributed Bag of Words version of Paragraph Vector (PV-DBOW)**
   The second method uses a paragraph matrix solely to compute the vector of words in context regardless of the context words in the input. This model is conceptually simple and needs less memory. At each epoch during training, a text window is sampled and a random word will be sampled from the text window forming a classification task given paragraph vector.

These two methods have their advantages, and they both turn out to have good performance according to the authors. Besides, the research also combines the paragraph vectors derived from two above ways and uses it in experiments, which is usually more consistent across various tasks.

4 Review Plan

- We will try to study and interpret the technique and significance of Word Vector proposed by authors in the paper. Particularly, the implementation of the second version (PV-DBOW) is very interesting and worth investigating.
- We aim to implement part of the test conducted by the authors and achieve similar performance using Stanford Sentiment Treebank Dataset[2] or IMDB Dataset[3] originally used in the paper. We may test other datasets if necessary.
- We will compare our test outcome and the result in paper. We plan to discuss why Paragraph Vector works well and its possible limitations.
- Since the paper now has over 5000 citations, we will follow up the latest development of authors' proposal and think about possible future directions.

---

[1] "Distributed Representations of Sentences and Documents." 16 May. 2014, https://arxiv.org/abs/1405.4053. Accessed 28 Feb. 2020.
[2] "Stanford Sentiment Treebank - Stanford NLP Group." https://nlp.stanford.edu/sentiment/. Accessed 29 Feb. 2020.
[3] "IMDb Datasets." https://www.imdb.com/interfaces/. Accessed 29 Feb. 2020.

Appendix

Previous study shows that learning good vector representations for millions of phrases is possible[4]. Following is the representation of word vectors:

- Given a sequence of training words $w_1, w_2, w_3 \ldots, w_T$, the objective of the word vector model is to maximize the average log probability:

$$min \frac{1}{T} \sum_{t=k}^{T-k} log\, p(w_1 | w_{t-k}\ ,\ \ldots,\ w_{t+k}\ )$$

- The prediction task can be done via softmax

$$p(w_t | w_{t-k},\ \ldots,\ w_{t+k}) = \frac{e^{y_{wt}}}{\sum e^{y_i}}$$

- Un-normalized log probability of each output word i which is y_i can be computed, where W is the word vector matrix, U and b are softmax parameters

$$y = b + Uh(w_{t-k}, \ldots, w_{t+k}; W) \quad (1)$$

[4] "Distributed Representations of Words and Phrases and their ...." 16 Oct. 2013, https://arxiv.org/abs/1310.4546. Accessed 28 Feb. 2020.