LAZADA

I Problem

II Literature Review

Words Embedding:

<http://blog.christianperone.com/2017/02/introduction-to-word-embeddings/>

Idea: instead of create a space with a dimension per word which takes a lot of processing time and memory; use a language modeling of low dimension and dense, but increased complexity.

Models: Word2vec, GloVe, Neual Language Model.

**Sequential Short-Text Classification with Recurrent and Convolutional Neural Networks**

**(Ji Young Lee, Franck Dernoncourt)**

This paper explores different machine learning algorithms for short-text classification. They first vectorize the short text through either CNN or RNN. Then these short text vectors are classified trough a two-layer feedforward ANN.

As the data they classify is short text extracted from dialogs, they include a sequential memory process (taking a number of previous and following short text vectors with the current vector to classify) for classification.

The results, for the no memory model, is around 65% for both CNN and RNN vectorizations, fairly better than classical SVM or Naïve bayes classification. Still, this study main focus is classification of sequential short-texts.

**Product Title Classification versus Text Classification**

**(Hsiang-Fu Yu∗, Chia-Hua Ho†, Prakash Arunachalam‡, Manas Somaiya‡, Chih-Jen Lin†)**

This paper tries to identify the key differences between product title classification and general text classification.

They first conclude that Stemming and stop word removal should not not be applied in product title classification.

Secondly, data normalization does not downgrade SVM performance. Also binary (vectors of 0, 1 according to presence or absence of corresponding word) and TF-IDF representations perform similarly

Thirdly, because of the short length of titles, bigram or degree-2 polynomial mappings are more effective than in general text classification case.

The method employed is quite interesting:

* Conversion of all words into lower case.
* Simple tokenization by splitting titles on spaces, punctuations and number/alphabet transitions. For example, “aa:bb 70d” becomes four tokens: “aa,” “bb,” “70,” and “d.”
* No stemming; no stop-word removal.
* Binary unigram as feature values.
* One-versus-rest multi-class SVM for classification

III Data Exploration

IV Choice of Features Space

V Implementation