TetraChrome--Boeing Aviation Service Word Representation Learning

Background

Word embedding is the collective name for a set of language modeling and feature learning techniques in natural language processing (NLP) where words or phrases from the vocabulary are mapped to vectors of real numbers in a low-dimensional space relative to the vocabulary size ("continuous space"). To be simple, in NLP architecture, we cannot make “String” as input, we need to find an encoding method instead, and the very straightforward encoding method is One-hot encoding, in which the vector dimension equals to the size of vocabulary.

Methods to generate this mapping include neural networks, dimensionality reduction on the word co-occurrence matrix, probabilistic models, and explicit representation in terms of the context in which words appear.

Word and sentence embedding, when used as the underlying input representation, have been shown to boost the performance in NLP tasks such as syntactic parsing and sentiment analysis.

Value Proposition

What we want?

All text mining project may need to try word embedding as features. Getting the best word embedding will lay a solid foundation for future text mining project. As a result, we treat word embedding as an important separate task in Boeing Information Extraction project, and perform it as a pre-trained phase for down streaming tasks, such as Maintenance Data NER (Name Entity Recognition).

1. Due to lack of domain-specific knowledge on doing handcrafted feature engineering, word embedding can help us do feature engineering automatically.
2. Boeing Corpus has small size of supervised (labeled) data, which is not sufficient for training a robust model. But Boeing Corpus has large size of unsupervised data, which can be sufficient enough to be utilized for word embedding.
3. Boeing Corpus is mostly maintenance data, instead of Natural Language, so it’s meaningless to use existed word embedding.
4. Entities in Boeing Corpus mostly occur in chunks, which have strong semantic relations by phrases
5. Boeing Corpus is quite noised, so the down streaming NER task can benefit a lot from a robust word embedding. Some examples of noised data can be seen below (Not Limited to)
    a. Spelling Error: FLIGHT -- > FLIGT
    b. Abbreviation: FLIGHT -- > FLT
    c. Synonym: SEAT 1A -- > SEAT 3B
    d. Space Error (may be difficult to correct, should come up with some clever token pre-processing or character-wise processing): CARRY OUT -- > CARRYOUT or FLIGHT -- > FLI GHT

Based on these reasons, we expect to train word embedding on unlabeled raw data, to make words with similar meanings have larger cosine similarity in a vector representation. Our group has done some preliminary implementations on word embedding. As stated before, we mainly focus on Maintenance Data NER task (using CRF and LSTM), which is down streaming task of word embedding. Currently with word embedding, our performance can be boosted by 2-3 percent by F1 Score. We treat it as huge improvement when performance is over 0.85. With help of CMU students, we may get high chance to improve our down streaming NER metrics with a robust and Boeing-data-specific word embedding implementation.

What you can learn?

Word Embedding can be very meaningful for master students as group project.

1. Word Embedding is an important branch in NLP and Deep Learning. Getting familiar with this technique can help you gain solid foundation on those area.
2. Word Embedding is a typical Machine Learning Projects, which can be easily expanded and applied by CMU students with curriculum learned. In turn, it may help your other machine learning or NLP related course project.
3. Many of state-of-art word embedding techniques are implemented by Neural Networks or even Deep Learning (LSTM encoding or CNN), which are useful for students' future career with staying on hot technology.
4. Word Embedding is a mature and hot academic problem, which you can solve by building an algorithm with code language you preferred, whatever using existed open source tools or start from scratch.
5. Word Embedding can be easily expanded to other techniques with very few modifications, like Collaborative Filtering, Recommendation System and Information Retrieval.
6. Word Embedding task can be easily split within group. Group members can do paper research together and implement different modules of algorithm separately, or can try different approaches and compare the results.

Main Task and Evaluation metric

The implementation and experiments basically focus on two areas

1. Model architecture. There are lots of variation based on standard word2vec besides what we have tried. Considering special property of our text, specialized model structure may improve quality of word vectors.
2. Parameter tuning. The best parameter settings on paper may be not transferrable to our dataset. Good parameters may further improve our metrics.
3. CMU teams will only need to provide a trained word to vector mapping for our packaged LSTM-CRF-Ensemble model, they can use our built pipeline to examine word embedding performance, and we can merge their results to existing model easily.

Word Embedding is an unsupervised task, but we have come up with some evaluation metrics to decide whether word vectors perform well, including unsupervised and supervised numeric metric.

1. Unsupervised: Perplexity on unseen Data. Our preliminary word embedding perplexity on Boeing dataset is 27.96, and we want to achieve a lower perplexity on that. (Students can perform experiments on standard Penn Treebank dataset first, however, since Boeing dataset has less flexibility than Penn Treebank dataset, they would probably get a relatively high perplexity on Penn Treebank data.)
2. Supervised: The ultimate gold for word embedding is to improve F1 Score on NER task with our CRF and LSTM models. We will package our models as black box (prefer docker) and let you run with their pertained word embedding, to see whether there are significant improvements (by t-Test).
3. You can also visualize word vectors and do word analogy, related work can be found by t-SNE technique.

Dataset

For training word embedding, we will provide "ALL CARRIER ATA36 & ATA78 MAINTENANCE RECORD SINCE 2012" dataset (location TBD), which has more than 819,498 records in total. CMU students need to split these dataset into training and hold-out validation data for experiments.

For Supervised examination, we will provide "ALL CARRIER ATA36 & ATA78 COMPLAINT & ACTION" dataset (location TBD), which has more than 7,000 records in total, with 5 labels.

We will provide F1 Score without pre-trained word2vec using LSTM-CRF-Ensemble model, as baseline. We will also provide F1 Score with our preliminary pre-trained word2vec using LSTM-CRF-Ensemble model, for reference and t-test.

Schedule

We already have mature content to start. The concrete schedule will be changed depends on students' progress. For CMU first year students, we will give enough time to research first to get familiar with the problem.

Boeing information extraction team will hold a bi-weekly meeting to update CMU workflow. Kang and Yan will hold a quick weekly meeting to update status and give feedback. CMU students need to update results on GitHub, either documentation or code, and write a progress report every month.

Resources

Our team has researched and used several word embedding techniques, which can be informed as below.

1. Word Embedding survey
   https://www.researchgate.net/publication/301779119_A_Survey_of_Word_Embedding_Literature_Context_Representations_and_the_Challenge_of_Ambiguity
2. Original word2vec paper
   https://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf

3. Extension of original Google word2vec, https://github.com/wlin12/wang2vec
4. CNN(Convolutional Neural Network) based Character level word embedding, using Torch, https://github.com/yoonkim/lstm-char-cnn

We Need Your Feedback!

Answer the following questions:

1. Any questions or suggestions about this project?
2. How many hours are you willing to devote to this project?
3. List 3 fixed time spot every week for weekly and biweekly meeting.
4. We have built a GitHub Organization, where we will update some documentations. Please send us your GitHub ID so that we can invite you to the organization.

Please send answers to us by email or Wechat before 1:00 PM Pittsburgh local time Friday (Oct 21st). Thanks!

We really look forward working with you !!