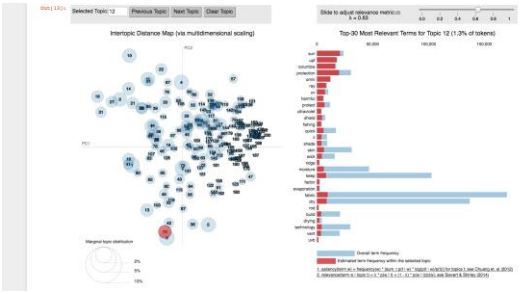
LDA/Doc2Vec example with PCA/LDAvis visualization

Indiana University
Data Science Summer Camp Poster Competition | gofind.ai
Bo Peng | bopeng.data@gmail.com

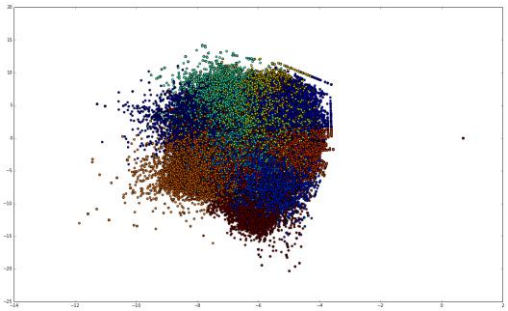
This poster project is about LDA/Doc2Vec with PCA/LDAvis as visualization. Latent Dirichlet allocation (LDA) is a topic model that generates topics from a set of documents. A 200 topics LDA model was generated using 1 million lines of clothes catalog data.

I used the Gensim LDA implementation and ran my data using a 32 core AWS EC2 instance. I then used the pyLDAvis package to generate a visualization of the 200 topic models. It is shown on the figure to the right.



You cannot see clearly from the figure, but even on 1 million lines of data, it was able to generate and cluster distinct topics. Topic 12 is a sun/outdoors topic, and it is very close to topic 3, an outdoors/sports/performance topic, and topic 42, a yoga/new age/exercise topic. See the github/youtube links for a clear picture of the model.

I then generated a 20 topic LDA model again using 1 million lines of data. I used this model to embed each clothes catalog line or document with the topics that it was related to. So each line was a 20x1 array, with the weights of the topics as inputs for the array. The documents were color coded as the maximum weighted topic in the array. Data was normalized with z score standardization on the columns. To the right is a figure of one of the images. As you can see, it shows distinct clusters.



Lastly, I used Doc2Vec to do document and word embeddings on the same corpus of data, but using 30 million lines of data as the input. This was also run on an EC2 instance. It was able to make some reasonable outputs for vector subtraction/addition as it knew that Calvin was to Klein, as Tommy was to Hilfiger. Both are clothing brands. It performed very well on computing cosine similarities of words and was able to distinguish which words least belonged in a group of words. You can see from the example below.

```
print(model.doesnt_match("black blue yellow shirt navy black green orange".split()))
print(model.doesnt_match("halloween costume devil party  scarf".split()))
```

```
shirt
scarf
```

In conclusion, LDA and Word2Vec/Doc2Vec are great natural language processing tools that can be used to understand and make correlations to large amounts of unstructured text as we saw in this paper.

Project Youtube: https://youtu.be/i3Opb3-QNX4
Project Github: https://github.com/BoPengGit/LDA-Doc2Vec-example-with-PCA-LDAvis-visualization