Project 4: Report

CSE 535: Information Retrieval (Fall 2016)

Instructor: Rohini K. Srihari, Teaching Assistants: Nikhil Londhe, Lu Meng, Le Fang, Jialiang Jiang

Team Name: Ask Snow

Members:

Pruthvi Mulagala (50208595)

Vaibhav Sinha (50208769)

Vamsi Krishna Nagabhiru (50204813)

Vinod Veparala (50208038)

Table of Contents

Overview

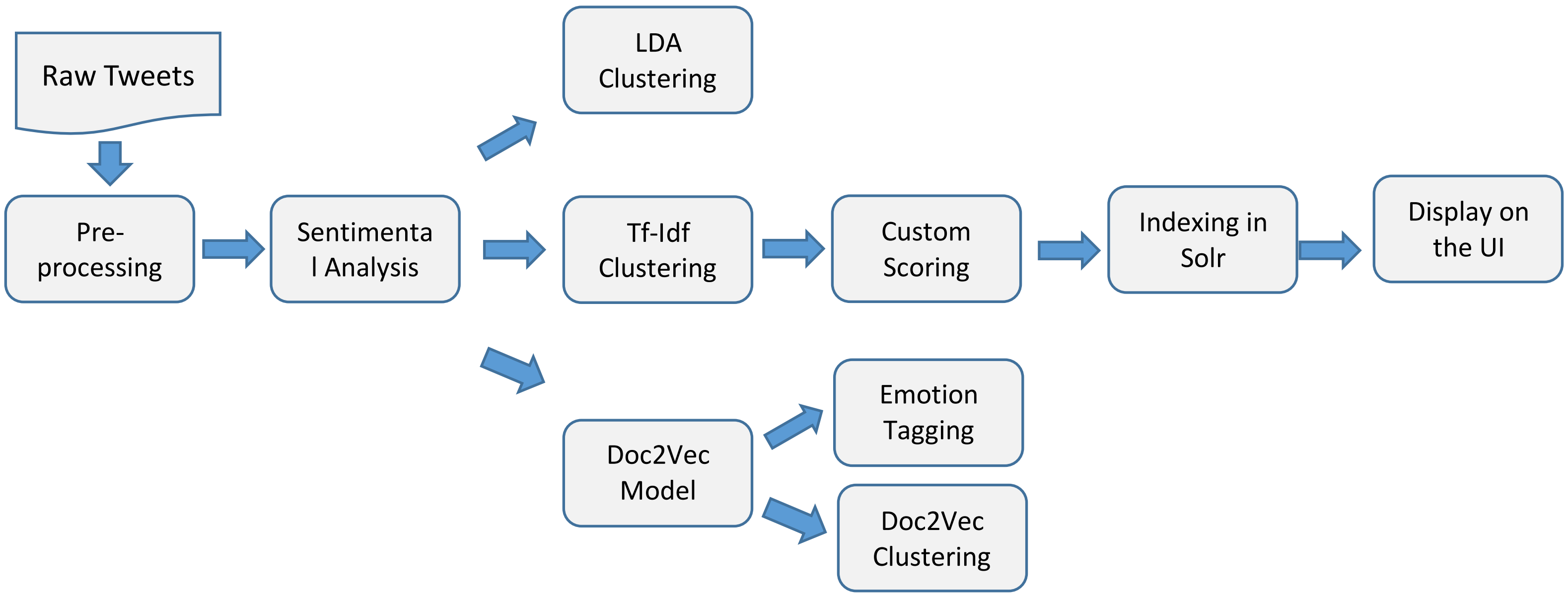
As part of this project we have developed a search system powered by Solr, which lets the user to check for summarized data from twitter indicating an overall picture of the several types of sub-topics related to the searched keyword. In order to achieve this we have tried implementing several Machine Learning models such as LDA, Doc2Vec, Tf-Idf for clustering the tweets based on their relevancy. We have also integrated sentimental analyzers and emotions tagging for the tweets and designated a custom score for each tweet for deciding the relevancy.

Data Corpus

The data corpus used for our search system has been collected from Twitter for the topic “Christmas”. We have collected a total of 50000 tweets spanning over 4-5 days using Twitter streaming API. Raw tweets collected from Twitter were pre-processes to remove emoticons, urls from the Tweet text and then we implemented all the clustering algorithms before indexing them in Solr.

Overall Topic Summarizing Workflow

Given below is the overall work flow of the data from Data Collection to displaying the data on the UI.



Implementation

After we have collected the raw tweets from Twitter for the topic Christmas, we have implemented few clustering algorithms to categorize the tweets into multiple clusters and then obtain the summary of each cluster. Besides, we have also implemented sentimental analysis on all the tweets to tag them with respect to their sentiment (positive/negative/neutral) using Natural Language Tool Kit (NLTK). We have also done Emotions tagging for all the tweets and categorized the tweets into five broad emotions – joy, fear, anger, sad, disgust. We have used Tensor Flow and Alchemy API in this regard.

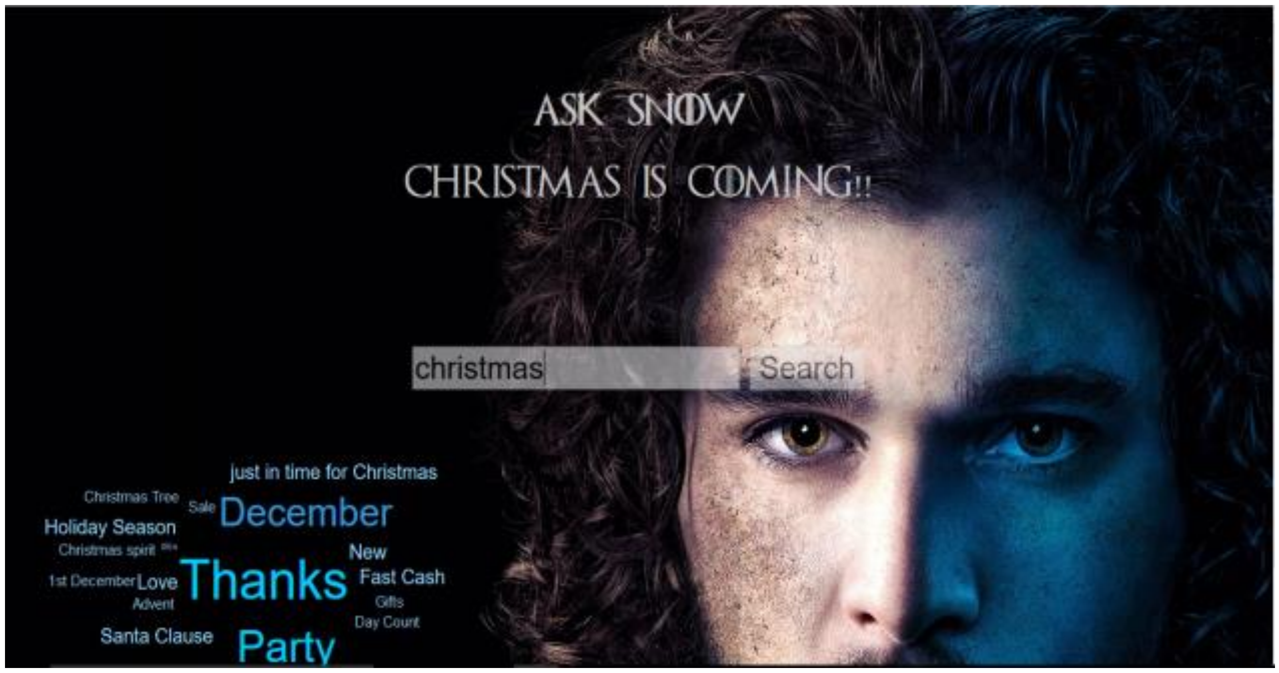
Coming to the User Interface part, the search system is built on Python and Django web framework. The web page is developed using HTML and CSS and on the backend we used Django. We have come up with a simple uncluttered user interface to enable smooth searching functionality to get the best possible search experience. Dynamic components such as the Tag Cloud and the Pie Charts were developed in JavaScript.

Code Repository

User Interface:

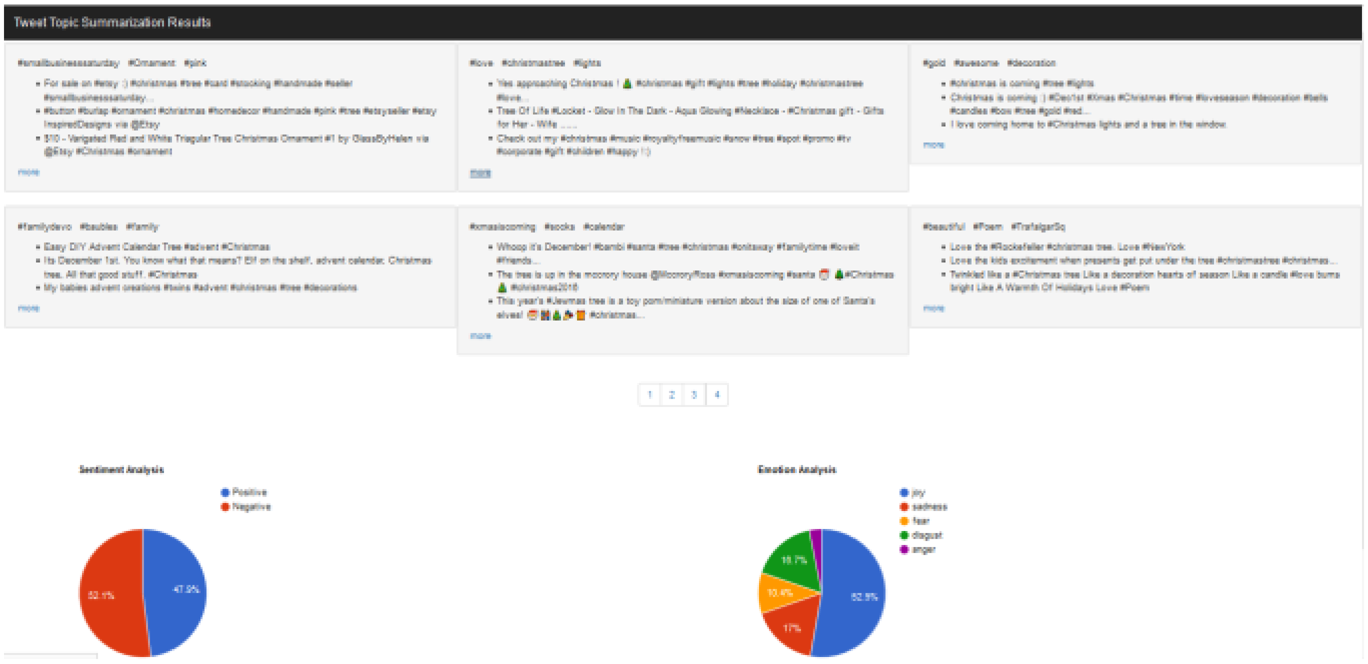
UI for Search Page

This is the home page containing the search box and some sample queries. Topic names of each cluster are shown in a word cloud beside the search box. Given below is the screen shot of our UI search page



Results Page

Upon clicking the search button, the user is taken to the Results page. This page contains different clusters into which the results were divided and each cluster's label, top tweets and emotions extracted using Alchemy API are shown. The top Hashtags along with the emotion analysis help the user understand what is going on in each cluster.



Tweet Topic Summarization Results

#smallbusinesssaturday #Ornament #pink

- For sale on #etsy :) #christmas #tree #card #stocking #handmade #seller #smallbusinesssaturday...
- #button #burlap #ornament #christmas #homedecor #handmade #pink #tree #etsyseller #etsy InspiredDesigns via @Etsy
- $10 - Varigated Red and White Triagular Tree Christmas Ornament #1 by GlassByHelen via @Etsy #Christmas #ornament

more

#love #christmastree #lights

- Yes approaching Christmas ! 🎄 #christmas #gift #lights #tree #holiday #christmastree #love...
- Tree Of Life #Locket - Glow In The Dark - Aqua Glowing #Necklace - #Christmas gift - Gifts for Her - Wife ....
- Check out my #christmas #music #royaltyfreemusic #snow #tree #spot #promo #tv #corporate #gift #children #happy !:)

more

#gold #awesome #decoration

- #christmas is coming #tree #lights
- Christmas is coming :) #Dec1st #Xmas #Christmas #time #loveseason #decoration #bells #candles #box #tree #gold #red...
- I love coming home to #Christmas lights and a tree in the window

more

#familydevo #baubles #family

- Easy DIY Advent Calendar Tree #advent #Christmas
- Its December 1st. You know what that means? Elf on the shelf, advent calendar, Christmas tree. All that good stuff. #Christmas
- My babies advent creations #twins #advent #christmas #tree #decorations

more

#xmasiscoming #socks #calendar

- Whoop it's December! #bambi #santa #tree #christmas #onitsway #familytime #loveit #friends...
- The tree is up in the mccrory house @MccroryRosa #xmasiscoming #santa 🎅 🎄#Christmas 🎄 #christmas2016
- This year's #Xmmas tree is a toy pom/miniature version about the size of one of Santa's elves! 🎅🎁🎄🦌🎁 #christmas...

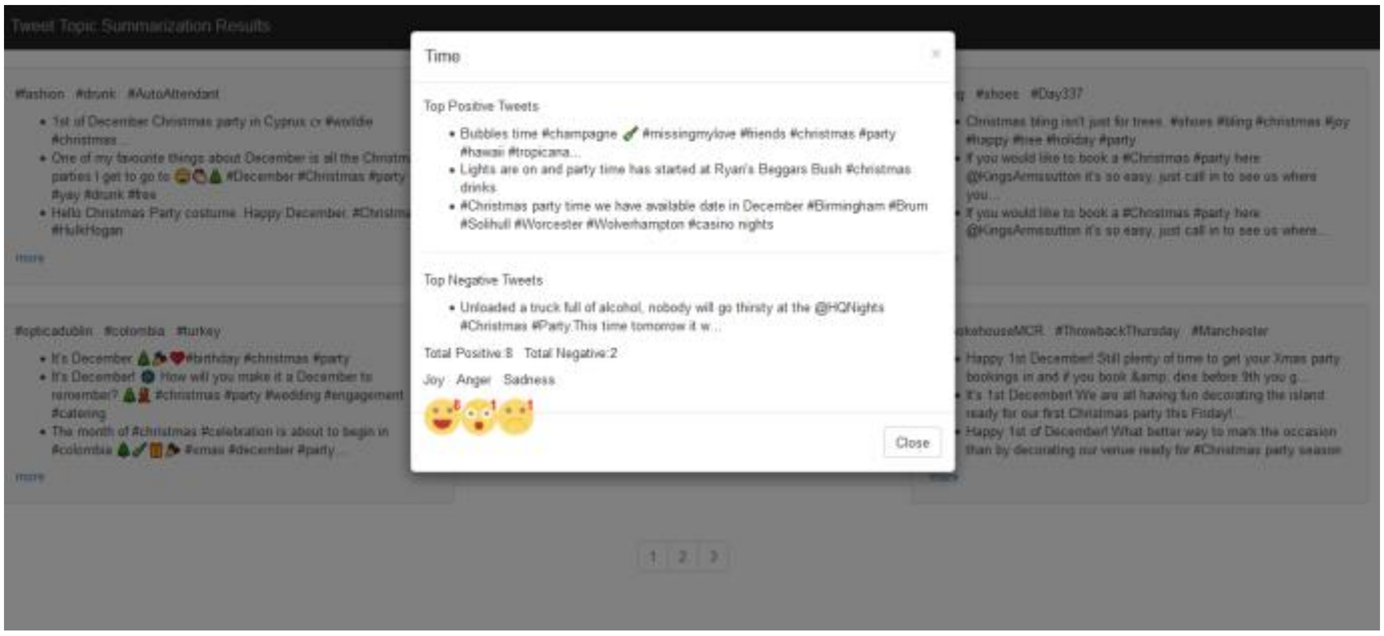
more

#beautiful #Poem #TrafalgarSq

- Love the #Rockefeller #christmas tree. Love #NewYork
- Love the kids excitement when presents get put under the tree #christmastree #christmas...
- Twinkled like a #Christmas tree Like a decoration hearts of season Like a candle #love burns bright Like A Warmth Of Holidays Love #Poem

more

1 2 3 4



Tweet Topic Summarization Results

#fashion #drunk #AutoAttendant

- 1st of December Christmas party in Cyprus or #worldie #christmas
- One of my favourite things about December is all the Christm parties I get to go to 😂😂🎄 #December #Christmas #party #yay #drunk #tree
- Hello Christmas Party costume. Happy December. #Christm #HulkHogan

more

#opticadublin #colombia #turkey

- It's December 🎄🎉❤#birthday #christmas #party
- It's December! 🌍 How will you make it a December to remember? 🎄🎅 #christmas #party #wedding #engagement #catering
- The month of #christmas #celebration is about to begin in #colombia 🎄🍾🎁🎉 #xmas #december #party...

more

Time

Top Positive Tweets

- Bubbles time #champagne 🍾 #missingmylove #friends #christmas #party #hawaii #tropicana...
- Lights are on and party time has started at Ryan's Beggars Bush #christmas drinks
- #Christmas party time we have available date in December #Birmingham #Brum #Solihull #Worcester #Wolverhampton #casino nights

Top Negative Tweets

- Unloaded a truck full of alcohol, nobody will go thirsty at the @HQNights #Christmas #Party This time tomorrow it w...

Total Positive:8 Total Negative:2

Joy Anger Sadness

😜😲😔

Close

#shoes #Day337

- Christmas bling isn't just for trees. #shoes #bling #christmas #joy #happy #tree #holiday #party
- If you would like to book a #Christmas #party here @KingsArmsSutton it's so easy, just call in to see us when you...
- If you would like to book a #Christmas #party here @KingsArmsSutton it's so easy, just call in to see us when...

#...kehouseMCR #ThrowbackThursday #Manchester

- Happy 1st December! Still plenty of time to get your Xmas party bookings in and if you book &amp; dine before 9th you g...
- It's 1st December! We are all having fun decorating the island ready for our first Christmas party this Friday!
- Happy 1st of December! What better way to mark the occasion than by decorating our venue ready for #Christmas party season

1 2 3

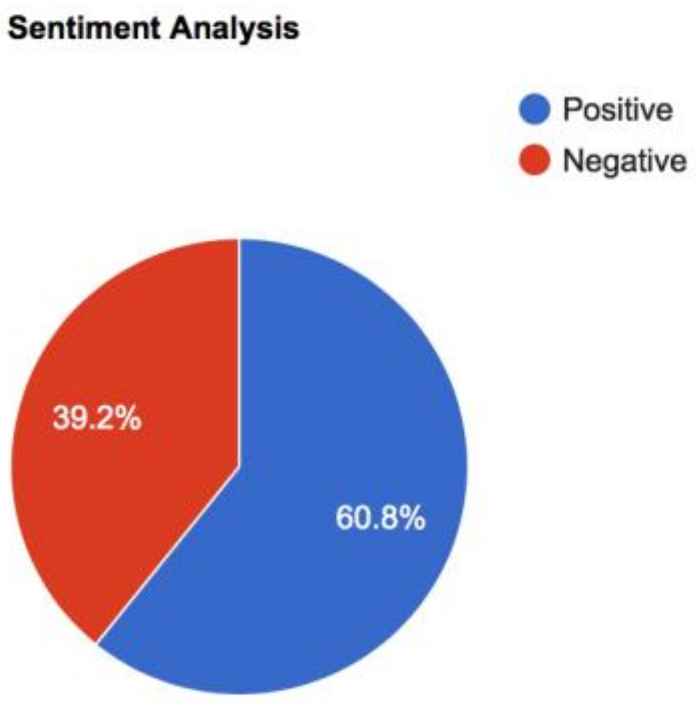
Sentiment Analysis using NLTK

All the tweets were tagged with sentiments using Sentiment Vader module in NLTK. Four types of score were generated. Below is a gist of the field look up in the json object.

- Tweet_text : “Cross stitching alllll night #Christmas kill me”
  "neg": 0.439, "neu": 0.561, "pos": 0.0, "compound": -0.6908
- Twwet_text : “I should make a new icon just for #Christmas. Gotta get into that holiday spirit 🎄🎄🎁”
  {"neg": 0.0, "neu": 0.732, "pos": 0.268, "compound": 0.5267}

For most of the tweets, the result was mostly favoring “compound” sentiment. But since we wanted to extract positive and negative tweets, we took only the values of these two fields.

The results were also shown graphically using a pie chart. A sample is shown below.



Emotion Tagging using Tensor flow and Alchemy

Emotions like sadness, anger and joy can also be treated as part of the summary. In general, we can also model a part of summary as a mixture of emotions conveyed. Our system is trained to identify emotions like joy, sadness, anger, disgust and fear in tweets.

In order to find out emotions, we used Alchemy API to tag tweets with emotions. But since we had API request limit on Alchemy API, we diligently tagged around 5000 tweets using Alchemy API.

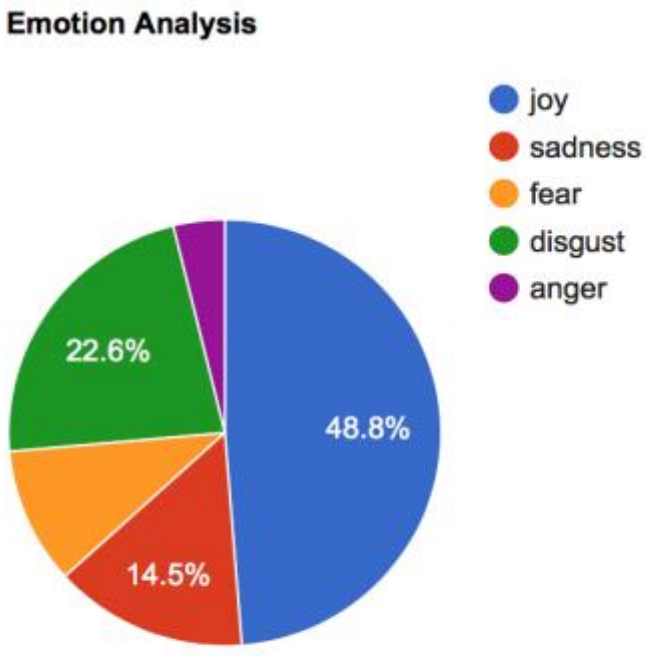
Since, we had around 50,000 tweets in our corpus, we used a sample of 5000 tagged tweets as training data and trained a multilayer artificial neural network to predict the emotion of rest of the tweets. Tensor flow is used to train a neural network having two hidden layers each having 256 nodes and the output layer having 5 outputs as described above using one hot encoding.

Tweets were converted to feature vectors using Doc2Vec Model. We partitioned 5000 tweets into two sets - 4000 tweets were used in training and 1000 tweets were used for testing. After round 100 iterations, we were able to achieve a training accuracy of more than 90% and around 87% on the testing set. Since we did not have a large training data, there were not much promising results, but this model can be extended if we had more data.

Besides, most of the data was biased towards joy emotion since Christmas mostly deals with positive tweets.

In Short,

1. Tagged 5000 Tweets using Alchemy API
2. Created Feature Vectors using Doc2Vec
3. Trained a multilayer neural network to tag rest of the 45000 tweets using Tensorflow
4. 90+% accuracy on training set and 87% on testing set



Sample Image which is shown in results which gives general outlook of the tweets in results.

This can help the user get an overall outlook of the people about a particular topic.

Code for Training Neural Network using Tensorflow

Clustering via LDA

Quoting Wikipedia

"In natural language processing, **latent Dirichlet allocation** (**LDA**) is a generative statistical model that allows sets of observations to be explained by unobserved groups that explain why some parts of the data are similar. For example, if observations are words collected into documents, it posits that each document is a mixture of a small number of topics and that each word's creation is attributable to one of the document's topics."

We tried to use LDA Model to find sub topics and group similar models together. For this, we used Gensim library in python.

Basically to implement LDA Model, we,

1. Tokenized each tweet using NLTK Regexp Tokenizer
2. Stemmed each tweet using Porter Stemmer
3. Removed the stop words
4. Created a dictionary of all texts using Gensim
5. Created bag of words format for each document using the above dictionary
6. Trained the Gensim model using different number of cluster topics and 10 passes over dictionary

Code for Model Training

Trained Model

Topics Identified

Sample Topic Identified Using LDA

0.058*"christma" + 0.024*"need" + 0.020*"give" + 0.019*"live" + 0.014*"away" + 0.013*"punch" + 0.013*"pinch" + 0.011*"hamper" + 0.010*"hour" + 0.010*"rt"

To identify which topic a tweet belongs to, LDA model gives the probability with which a tweet can be formed from a topic. From those probabilities, the maximum topic probability was used to identify the corresponding topic of that tweet.

The results of clustering were not that promising. It can be because of scarcity of data, so we decided to explore other models.

Doc2Vec Model for Tweet Document Vectors

In order to extract feature vectors for each tweet, Doc2Vec model was used to identify word embedding and generate proper feature vectors for each tweet.

Doc2Vec is an extension of Word2Vec Model from Gensim library. It works on the principle of Deep learning via the distributed memory and distributed bag of words models from using either hierarchical softmax or negative sampling. Document Vectors were successfully generated.

Code for training model

Trained Model

Clustering Using Doc2vec

1. Using the trained model, we inferred feature vectors for tweets to generate a feature matrix which is to be sent for KMeans Clustering.
2. Using the above feature vectors, we used KMeans clustering to cluster similar tweets together.
3. Unfortunately, the results were not that promising. This reason can be attributed to the fact that there was not enough data for training Doc2Vec Model and the document vectors were not that accurate.
4. So, we decided to explore other models.

Code for Clustering Using Doc2Vec

Document Vectors using TF-IDF

One more method of generating document feature vector was using the Tf-IDF Matrix. We constructed the term frequency-inverted document frequency matrix using NLTK’s TfidfVectorizer. We tuned the following parameters for generating document vectors

```
max_df=0.99
max_features=200000
min_df=0.01
stop_words='english'
use_idf=True
 ngram_range=(1,3))
```

Clustering using KMeans

The Tf-IDF matrix generated above was sent to KMeans Clustering algorithm to cluster the document into different number of clusters. On trying different number of clusters, we narrowed it down to 20 clusters.

Fortunately, clustering using Tf-Idf matrix lead to promising results and hence, we decided to use this as our final clustering model.

Code for Tf-Idf Matrix Clustering

Custom Scoring Algorithm for Summarization

In order to find out relevant tweets for summary, we came up with a custom scoring algorithm which added Static Score to all tweets taking the following factors into consideration.

1. Tweet's Retweet Count – The more the tweet is retweeted, the more chances are that it is relevant for the summary
2. Tweet's Favorite Count - – The more the tweet is liked by people, the more chances are that it is relevant for the summary
3. Search Scoring using BM25 Model – This makes sure the tweet is relevant for the search query.

Thus, while displaying tweets, tweets were scored by adding the above individual scores.

Limitations

- The data size we have used isn’t big enough to train the data for Doc2Vec and LDA clustering. With a significant data size ideally Doc2Vec and LDA models would have yielded better results.
- We have implemented all our models using English as the sole language. Results could vary a little bit with the ingestion of data from other languages.
- Even after implementing the relevancy of data in each cluster is not entirely accurate and there is still a lot of scope for improvement.
- We have finalized the number of clusters to 25 based on manual heuristics and is not the ideal way to cluster the data.
- The amount of topics we have covered in each cluster is limited.

Future Work

- We have tried and tested only three different clustering techniques for this project. Several other techniques like LSI, Word2Vec etc. could also have been implemented for better analysis.
- There is lot more scope for this project wherein we can implement several features taking advantages of Solr and our extensible UI framework.
- The UI can be expanded to include various customized dashboards spanning across different variables of the indexed data. This way we could show the demographics of the data rather than the actual data itself.
- Currently the effort has been confined only to a limited set of twitter data for a certain topic. We could expand the approaches we have taken for a larger data set which can even be expanded beyond the scope of Twitter.

References

1. Natural Language Toolkit
2. Doc2vec Model
3. LDA Model
4. Tensorflow
5. Sklearn
6. Alchemy API