Topic Modeling and Sentiment Analysis on News Articles and Comments

Simon Fraser University, Big Data Programming II

Chithra Bhat, Ruoting Liang, Tianpei Shen

SFU WORK INTEGRATED LEARNING

SOCC: The SFU Opinion and Comments Corpus

Opinion articles and the comments published in the Canadian newspaper *The Globe and Mail* in the five year period between 2012 and 2016.

- 10,339 articles + 1,280,454 comments
- 6,895,696 words(articles) + 77,238,179 words (comments)

Motivation

Create a succinct summary of opinions on articles, issues, or policies among Canadian citizens to help organizations make better decisions.

❖Observe the change in hot topic for these 5 years (2012 - 2016).

❖Discover the surrounding topics on comments under a given article.

❖Discover the constructiveness/toxicity/sentiment for the comments under a given topic.

Case Study:

Data-Science Pipeline

Data Collection

Collected labeled training data for constructiveness /toxicity/sentiment analysis.

- Labeled constructiveness data from The Yahoo News Annotated Comments Corpus (YNACC).
- Labeled toxicity/sentiment data from Kaggle.

YAHOO! RESEARCH kaggle

Data Cleaning & Integration

Used Python NLTK and Spark to clean text on articles and comments:

- Removed HTML tags
- Decoded the data to UTF-8
- Tokenized string
- Removed punctuations, numbers, and stop words
- Stemming

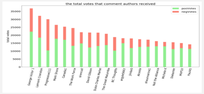
Prepared data files for topic modeling and sentiment analysis by integrating all data frames.

OpenRefine python Natural Language Analyses with NLTK APACHE Spark

Exploratory Data Analysis

- Number of opinion articles published from 2012 Jan-2016 Dec.

- Top-20 comment authors who got the most number of votes.



Topic Modeling

Article:
Used **LDA** and **NMF** algorithms to discover top 20 topics of all articles. Used **WikiApi** combined with human labeling to identify the topic name based on topic keywords generated.

Comment:
Created learning document by grouping all comments under one article. Used LDA and NMF algorithms to discover top 10 topics of all article comments.

WIKIPEDIA The Free Encyclopedia gensim scikit learn

Sentiment Analysis

Approach 1:

- Produced word embedding using **Word2vec** in *gensim*.
- Used **k-means** to do vector quantization (Assign centroid to each word).
- Number of features equal to the number of clusters.

Approach 2:

- Extracted Features from a Bag of Words used scikit-learn **CountVectorize**.

Trained model use **Random Forest Classifier.**

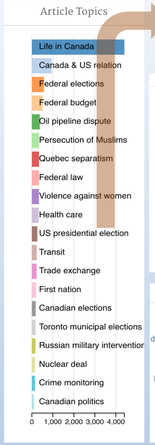
Visualization

Built a web application to create an interactive visualization of the learning results.
Tools used are listed below:

D3 ZingChart dc B Flask web development, one drop at a time plotly

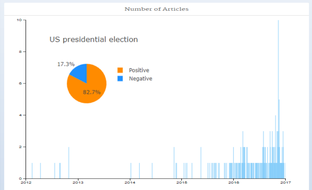
Result-web application

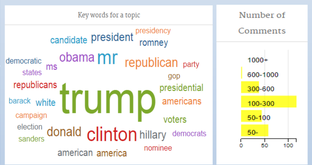
Change in Hot Topic



Topic: US presidential election

❖ Hot topic trends ❖ Sentiment
❖ Number of comments ❖ Keywords

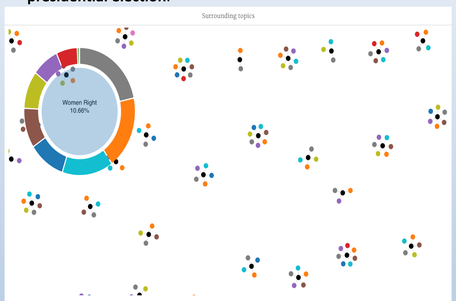




Surrounding Topics

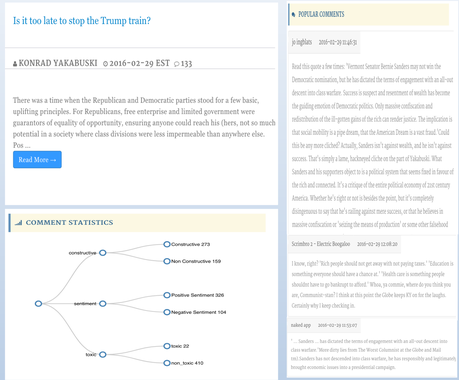
❖ Discover surrounding topics

- Colored nodes represents topic of comments.
- Black node are articles under topic **US presidential election**.
- Donut Chart displays the surrounding topics under topic **US presidential election**.



Constructiveness/toxicity/sentiment

❖ Discover constructiveness/toxicity/sentiment under an article (Filter toxic comments and display 5 most constructive comments)



Discussion

Stemming is not always good: The algorithm results in a list of word-vector, which is not human-readable. When performing the human labeling we noticed that stemming sometimes will give us strange word such as 'parti', 'conser'. Therefore we ignore the stemming step during cleaning data.

How to deal with large dataset: The comment dataset is 350MB. So we used Spark to process the data and created User Defined Function for code implementation.

How to do topic modeling on short-text: LDA is one of the most popular topic models and it models a document as a mixture of topics. However, it is not the case for short text, such as comments. Therefore we grouped all comments of one article and treated them as one document in order to perform the topic modeling.

The dataset doesn't contain label for constructiveness and toxicity: In order to analyze the constructiveness and toxicity of comments, we need to train the model with a labeled dataset. But the original data doesn't contain these labels, therefore, we use other sources to train the model and test on the original dataset.

Future Work

Crowdsourcing on article comments
As discussed above, the SOCC datasets do not contain labeled data. In order to build training dataset that better represent SOCC comment data, one approach is to use crowdsourcing and get labeled data with constructiveness, toxicity, and sentiment.

Make real time prediction
In the future, we can add more feature to our web application. For example, predict a topic in real time by giving an article URL or article text.

Reference:

- C., J., E., B., & A. (2017). *Finding Good Conversations Online: The Yahoo News Annotated Comments Corpus*. Valencia, Spain: Association for Computational Linguistics. Page 13-23
- Blei, D.M., Ng, A.Y., Jordan, M.I., 2003. Latent dirichlet allocation. J. Machine Learn.
- Cheng, X., Yan, X., Lan, Y., & Guo, J. (2014). BTM: Topic Modeling over Short Texts. IEEE Transactions On Knowledge And Data Engineering, 26(12), 2928-2941.