News Article Recommendation

Project for Stat689

Submitted by : Arch Desai, Sohil Parsana, Jay Shah

Motivation:

Online news articles reading has exploded as the web provides access to millions of news sources from around the world. The sheer volume of articles can be overwhelming to readers.

A key challenge of news service website is help users to find news articles that are interesting to read. This is advantageous to both users and news service, as it enables the user to rapidly find what he or she needs and the news service to help retain and increase customer base

Objective of the Project:

Objective of the project is to build a hybrid-filtering personalized news articles recommendation system which can suggest articles from popular news service providers based on reading history of twitter users who share similar interests (Collaborative filtering) and content similarity of the article and user's tweets (Content-based filtering).

This system can be very helpful to Online News Providers to target right news articles to right users.

But Why Twitter?

Statistics

- 74% of Twitter users say they use the network to get their news.
- 500 million tweets are sent each day.
- 24% of US adults use Twitter.

How Twitter can be used?

Based on user's tweets we can know user's interests and can recommend personalized news articles which user would share on Twitter. This can increase news articles and news service's popularity.

Project Flow:

1. **Collect active Twitter users' data**
2. **Analyze users' tweets**
3. **Cluster users according to their interests**
4. **Perform sentiment analysis and topic modeling**
5. **Collect and analyze news articles**
6. **Get user's Twitter handle & Recommend news articles**

1. Collect active Twitter users' data

As a first step, the engine identifies readers with similar news interests based on their behavior of retweeting articles posted on Twitter. **(Library : Tweepy)**

The flow of collecting active users' data:

- Get Twitter users who retweet tweets of New York Times, Bloomberg, Washington Post. We identify them as active news readers.
- Create a popularity Index
  - Popularity = Number of Followers / Number of Friends
- Filter users based on their twitter activity and popularity
- Collect information from Twitter profiles of these filtered users

Collecting Twietter Users Data

2. Analyze users' Tweets

The tweets contains URLs, Usernames, non-english words, punctuations and numbers. Sometimes whole tweets are in different languages. To get information from tweets, preprocessing is important. **(Library : NLTK)**

Preprocessing:

- Clean tweets
    - Removal of URLs, Usernames, numbers, non-english words, punctuations
- Tokenize tweets
    - Process of breaking stream of textual content into words
- Remove Stop words
    - Stop words: Most common words in a languages e.g the, is, am, are, there
- Stemming and Lemmatization
    - Both of these are text normalization techniques used to reduce inflectional forms and sometimes derivationally related forms of a word to a common base form.
    - Stemming : Chops off words without any context (PortStemmer)
    - walking : walk, smiled : smile, houses : house
    - Lemmatization : Finds the lemma of words with the use of a vocabulary and morphological analysis of words (WordNetLemmatizer)
    - better : good, are : be, ran : run
    - Difference:
    - Caring - Car : Stemming
    - Caring - Care : Lemmatization

Text Cleaning

3. Cluster users according to their interests

We can cluster users based on their similarity of interests retrieved from their tweets and that requires vectorized representation of tweets.

- Find TF-IDF matrix **(Library : TfidfVectorizer from sklearn)**

    - TF-IDF stands for Term Frequency- Inverse Document Frequency
    - TF : Gives frequency of words in each user’s tweets
    - IDF : Calculates the weight of rare words across all users’ tweets. The words that occur rarely in the corpus have a high IDF score.

    TDIF

    - TF-IDF is a weight that ranks the importance of a term in its contextual document corpus.
    - Perform K-means clustering to cluster users based on tf-idf matrix.
    - To reduce the dimension of Tf-Idf matrix we define error term, distance matrix
    - Distance Matrix = 1 - Cosine Similarity of users’ tweets
    - Cosine similarity = (dot product of two vectors) / (product of vectors’ magnitudes)
    - The cosine of the angle between the vectors is a good indicator of similarity
      cosine
    - Reduce dimension matrix using multi-dimension-scaling. **(Library: Skleran’s MDS)**

- Selection of optimal K with Elbow Method

    - The elbow method, in which the sum of squares at each number of clusters is calculated and graphed, and the user looks for a change of slope from steep to shallow (an elbow) to determine the optimal number of clusters.
      optimalK

    - We created 5 clusters and top words in 5 clusters are shown below.
      SelectedClusters

    - Clusterd User's Vizualization:

    - Here, we have used Manifold learning for vizualization.High-dimensional datasets can be very difficult to visualize. While data in two or three dimensions can be plotted to show the inherent structure of the data, equivalent high-dimensional plots are much less intuitive. To aid visualization of the structure of a dataset, the dimension must be reduced in some way.

    - The simplest way to accomplish this dimensionality reduction is by taking a random projection of the data. Though this allows some degree of visualization of the data structure, the randomness of the choice leaves much to be desired. In a random projection, it is likely that the more interesting structure within the data will be lost.

    - Multidimensional scaling (MDS) seeks a low-dimensional representation of the data in which the distances respect well the distances in the original high-dimensional space.

    - In general, is a technique used for analyzing similarity or dissimilarity data. MDS attempts to model similarity or dissimilarity data as distances in a geometric spaces. The data can be ratings of similarity between objects, interaction frequencies of molecules, or trade indices between countries

    Users Clusters

4. Perform Sentiment Analysis and Topic Modelling

Sentiment analysis -Computational study of opinions, sentiments, evaluations, attitudes, appraisal, affects, views, emotions, subjectivity, etc., expressed in text.

It is also called opinion mining

Sentiment Analysis

We have used pretrained model from Textblob library that gives two results:

Subjectivity and Polarity

- Polarity is score between [-1,1], where 0 indicates neutral, +1 indicates a very positive sentiment and -1 represents a very negative sentiment.
- Subjectivity is score between [0,1], where 0.0 is very objective and 1.0 is very subjective.
- Subjective sentence expresses some personal feelings, views, beliefs, opinions, allegations, desires, beliefs, suspicions, and speculations where as Objective sentences are factual.

Topic Modeling

Topic_model

Each topic is a distribution of words; each document is a mixture of corpus-wide topics; and each word is drawn from one of those topics.

In reality, we only observe documents. The other structures are hidden variables. Our goal to infer the hidden variables.

Topic_model2

- Per-document topics proportions θ_d is a multinomial distribution, which is generated from Dirichlet distribution parameterized by α.

- Smilarly, topics β_k is also a multinomial distribution, which is generated from Dirichlet distribution parameterized by η.

- For each word n, its topic $Z(d,n)$ *is drawn from document topic proportions* θd.
  Then, we draw the word $W(d,n)$ *from the topic* βk, where k=$Z\_(d,n)$.

Application in our problem

- Using LDA Mallet model for topic modelling of each individual cluster. It is more efficient than Gensim's LDA package requiring O(corpus).
- Tuning of number of topic for each cluster accomplished using the coherence measure: using C_v measure (combining normalized pointwise similarity and cosine similarity)
- There are 3 types of Topic Model gerneration
  i. EM
  ii. Variational EM
  iii. Full Gibbs estimating LDA generative model

The best performing coherence measure (the most left column) is a new combination found by systematic study of the conguration space of coherence measures.

This measure (CV) combines the indirect cosine measure with the NPMI and the boolean sliding window. This combination has been overlooked so far in the literature. Also, the best direct coherence measure (CP) found by our study is a new combination.

Topic Model Interactive Visualization

Interactive Vizualization of Topic model for Cluster 1

Interactive Vizualization of Topic model for Cluster 2

To use the visualization tool, click a circle in the left panel to select a topic, and the bar chart in the right panel will display the 30 most relevant terms for the selected topic, where we define the relevance of a term to a topic, given a weight parameter, 0 ≤ λ ≤ 1, as λ log(p(term | topic)) + (1 - λ) log(p(term | topic)/p(term)).

The red bars represent the frequency of a term in a given topic, (proportional to p(term | topic)), and the blue bars represent a term's frequency across the entire corpus, (proportional to p(term)).

Change the value of λ to adjust the term rankings -- small values of λ (near 0) highlight potentially rare, but exclusive terms for the selected topic, and large values of λ (near 1) highlight frequent, but not necessarily exclusive, terms for the selected topic.

A user study described in our paper suggested that setting λ near 0.6 aids users in topic interpretation, although we expect this to vary across topics and data sets (hence our tool, which allows you to flexiby adjust λ).

5. Collect and analyze news articles

We scraped recent news articles from different news channels using **python package : Newspaper3k**.

They have different categories so we can train our algorithm using all topics. And our algorithm can satisfy wide range of topics giving good and similar recommendations.

Newspaper is a Python module used for extracting and parsing newspaper articles. Newspaper use advance algorithms with web scrapping to extract all the useful text from a website. It works amazingly well on online newspapers websites.

6. Get user’s Twitter handle & Recommend news articles

There are two main types of collaborative filtering: user-based and item-based. Note that the two are entirely symmetric (or more precisely the transpose of each other)

1. **Content-based Filtering**

   - Based on the tweets of a user, we can identify his or her interests.
   - Based on the similarity of user’s interests and news article’s content/tags/headlines, we can recommend news articles.
   - The approach has intuitive appeal: If a user posts ten tweets having the word “Clinton,” user would probably like future “Clinton”-tagged news articles.
   - Example- Amazon(Recommendation based on recently viwed items)

- Shortcomings of this method:
- Since, rare words have large weightage in the algorithm it sometimes degrades the performance.
- For example, if a user’s one tweet contains word “election”, he would get recommendation of news articles tagged election as it is a rare word and higher weightage is given to it.

2. **Collaborative Filtering**

- Based on tweets of a user, we can identify a cluster in which user belongs to.
- Based on the topics of each cluster, we can recommend news article to a user
- For example, If a user tweets about election, he or she can be assigned to a cluster of users who have read and retweeted news articles that our user isn't aware of on the topic of election and we can recommend it to a user
- Example - Amazon (Customer who bought this item also bought)
- Shortcoming:
  - this approach fails at recommending newly-published, unexplored articles: articles that are relevant to groups of readers but hadn’t yet been read by any reader in that group.

Recommendation

3. **Hybrid Filtering**

- In these methods, a combination of both recommendation algorithms are used to maximize advantage and minimize the drawbacks of both algorithms.

- The different methods for hybridization are shown below

Method	Description
Weighted	The scores (or votes) of several recommendation techniques are combined together to produce a single recommendation
Switching	The system switches between recommendation techniques depending on the current situation.For example, in earlier phases, one might use a knowledge-based recommender system to avoid cold-start issues
Mixed	Recommendations from several different recommenders are presented at the same time
Feature combination	Features from different recommendation data sources are thrown together into a single recommendation algorithm. Similarity with Stacking

Our Approach (Feature Combination)

- We get the news articles based on the topics of a cluster in which user belongs to.
- We compare the content of news articles to the content of a user’s tweets and find similarity
- We rank the articles based on similarity with users personal interest and recommend to users.

Collaborative Filtering Results

		Example Recommended Article	
	Cluster	CNN	NY Times
Earth Day	0	https://www.cnn.com/2019/04/24/entertainment/top-credit-cards-for-those-with-excellent-credit?	https://www.nytimes.com/2019/04/24/opinion/california-wildfire-climate.html
Trump News	1	https://www.cnn.com/2019/04/24/politics/presidential-tax-returns-states-2020-trump/index.html	http://www.nytimes.com/2019/04/24/us/politics/russia-2020-election-trump.html#commentsContainer
Terrorist Attack	2	https://www.cnn.com/2019/04/24/investing/ford-rivian/index.html	http://www.nytimes.com/interactive/2019/04/23/world/asia/sri-lanka-isis-religious-ethnic-tensions-map.html
Entertainment News	3	https://www.cnn.com/2019/01/30/business/kohls-weight-watchers/index.html	http://www.nytimes.com/2019/04/24/sports/damian-lillard-portland-trail-blazers.html
World News	4	https://www.cnn.com/2019/04/15/australia/australia-racism-media-christchurch-attack-intl/index.html	http://www.nytimes.com/2019/04/24/opinion/rwanda-genocide.html

- Sample Articles identified wih the Clusters:
  Recommendation2

- Here we added weight to our recommendation personalized by individual user

- After user has been classified into clusters, we will calculate similarity score of user’s interest with identified articles within each clusters.

- Weight = Topic Modeling Normalized Prob(80%) + Sentiment Score (20%)

- Based on this weight criteria we will rank the articles personalized for each user

Final Pipeline

FinalPipeline

- Currently we have already trained this on latest few articles:
- If you want to try this model, you can clone this repo and run following code in your prompt.
- Before running you make sure that you have access to the twitter Authentication and change in model/config_model.ini file.

```
python test.py <user_id>
# example
# python test.py katyperry
```

- And if you want to run this analysis on current data, you can create an Anaconda env with provided requirement file.

```
# For Windows users# Note: <> denotes changes to be made

conda create --name <env_name> requirements.yml

# Make sure you have updateed the provided config file
python train.py config_model.ini

python test.py <user_id>
```

Current Challenges:

- How to recommend news articles to a user if he or she does not have any tweets? (Cold Start Problem)
- How to evaluate the performance of a recommendation system?
- Topic Modelling and retrieving users' data are bottlenecks as they need to be updated very frequently.
- Scale this model on large dataset