Topic Modeling in Social Media

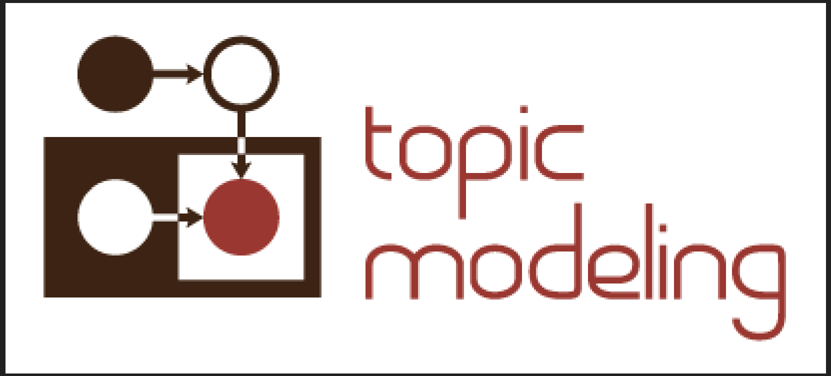
Faculty Mentor: Dr. Vasudeva Varma

Student Mentor: Sandeep Panem

Kashyap Murthy Krishnakant Vishwakarma Sajal Sharma Vinil Narang

Topic Modeling

- A Topic Model is a statistical model for discovering the abstract "topics" that occur in a collection of documents.

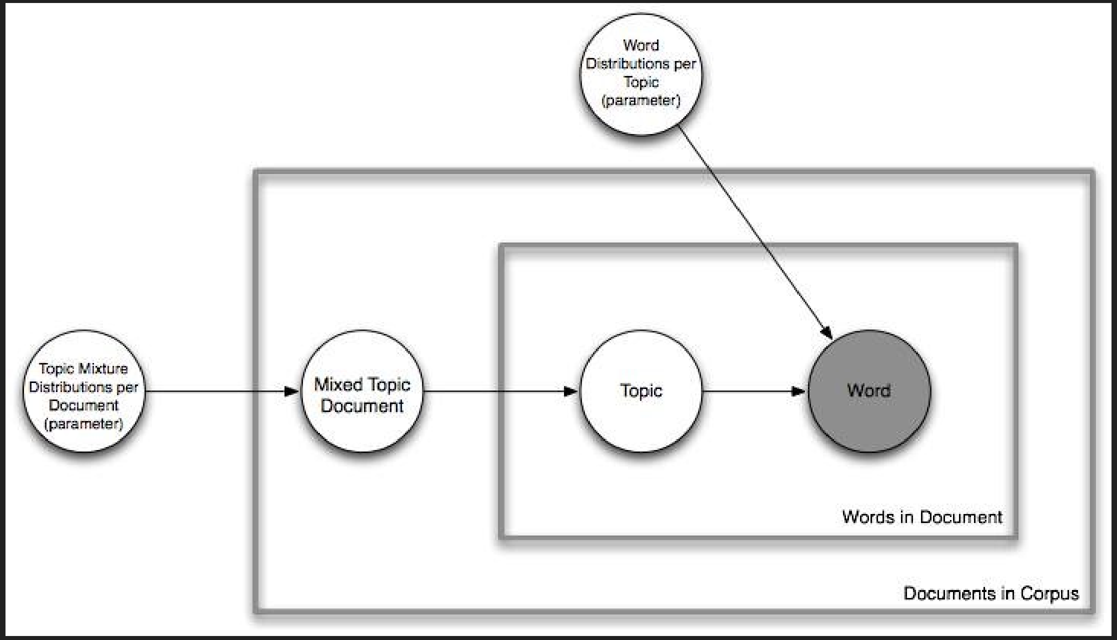


LDA(Latent Dirichlet Allocation)

- Topic Modeling represents documents as **mixtures of topics** that spit out words with certain probabilities.
- LDA is the most common topic model currently in use.
- LDA is used in social platforms to come up with a model which can predict on which topics related document a user prefers to read and write.

LDA …

- LDA is a method which considers documents beyond mere just bag of words.
- It describes a generative process whereby, given a Dirichlet conditioned bag filled with topic-distributions for each document, we draw a topic mixture from this bag. Then, we repeatedly draw both a topic and then a word from that topic to generate the words in that document.
- we have a generative model that represents the process by which a corpora of documents were created in terms of Topics.


Word Distributions per Topic (parameter)
Topic Mixture Distributions per Document (parameter)
Mixed Topic Document
Topic
Word
Words in Document
Documents in Corpus

Approaches for topic modeling

- Approach1: Finding LDA model for each user in the network.
- Approach2: Finding top K influential users and applying the LDA model on these users only.
- Approach3: Finding communities present in the network and approximating users topics by applying LDA model of its corresponding community.

Comparing different approaches.

- Approach1:
  - Calculating LDA for each user is very costly since the data set is very large.
  - So this approach is not feasible for practical purposes.

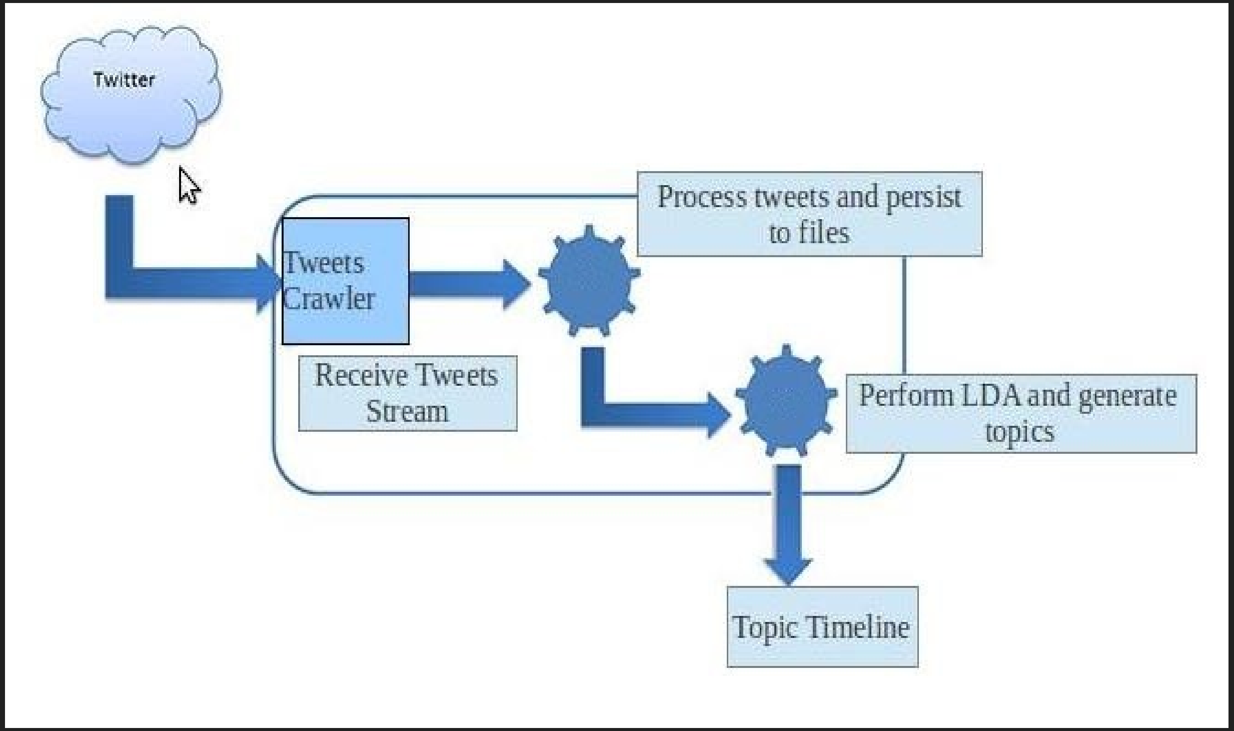
Comparing different approaches.

- Approach 2:
  - Since LDA can’t be applied to each user, therefore we have selected top k users on which we have applied LDA.
  - The top k users were found out using Page Rank algorithm.
  - Then we extracted the tweets for these k users and applied LDA on them.

- For this project we followed approach 2.

Approach 2…

- 1. Step 1:
  - Input : A file containing the follower-followee relationship in the form of edge list. It contains ~6.25 crore nodes(users) and ~146 crore edges(follower-followee relationship)
  - Output : A file containing the userid and pageranks of all the nodes(users) in the graph.
  - Approach : Used GraphChi’s pagerank algorithm to perform the above mentioned task.
- 2. Step 2:
  - Input : A file containing the pageranks and userids of all the nodes(users) in the graph.
  - Output : A file containing the pageranks and userids of all the users sorted on the basis of pagerank.
  - Approach : External merge sort is used to perform this kind of sorting because the size of the file to be sorted was 30GB.


Twitter
Tweets Crawler
Receive Tweets Stream
Process tweets and persist to files
Perform LDA and generate topics
Topic Timeline

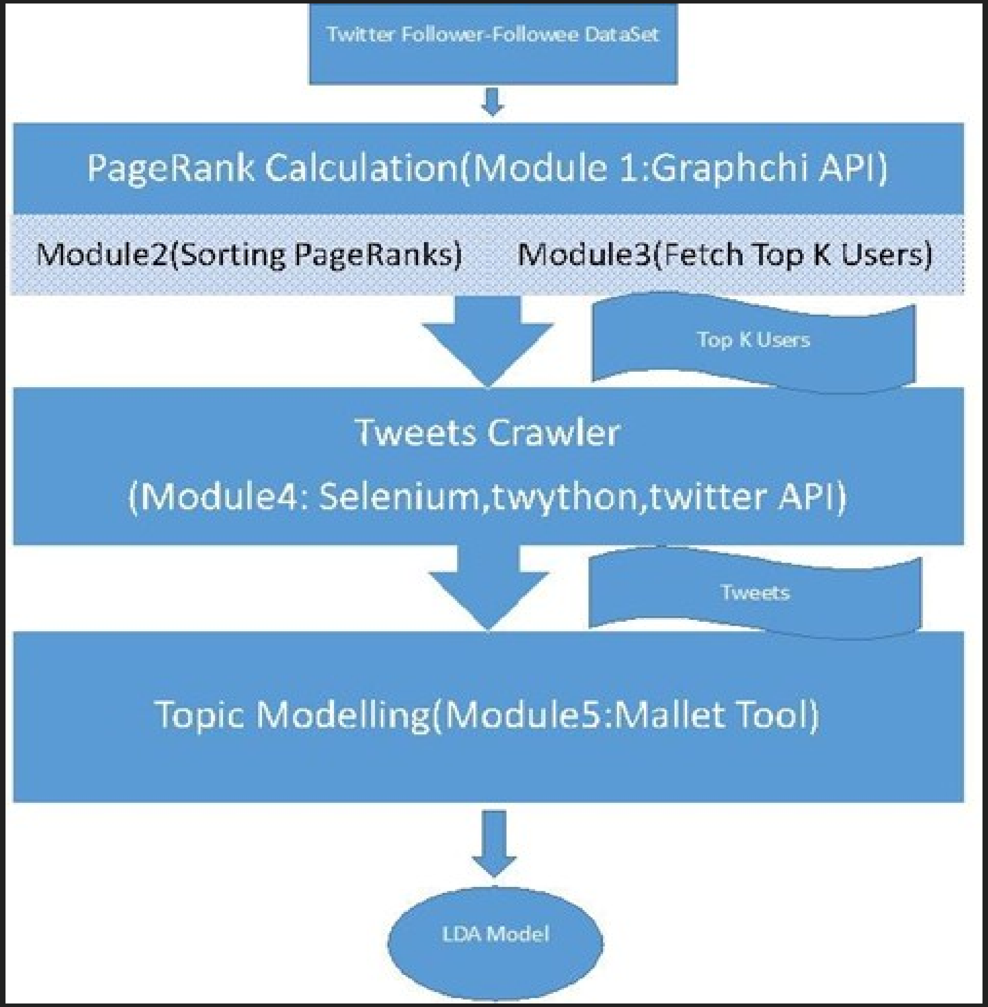
Approach 2…

- 3. Step 3:
  - Input : A file containing the pageranks and userids of all the nodes sorted by pagerank.
  - Output : A file containing the top 50 users(userids and pagerank)

- 4. Step 4:
  - Input : A file containing the top 50 users.
  - Output : The tweets of these top 50 users.
  - Approach : We used the tweet crawler to crawl the tweets of these users.

- 5. Step 5:
  - Input : 50 files containing tweets of top 50 users.
  - Output : LDA models of these 50 users.
  - Approach : Used Mallet to perform this task.

Approach 2...



Twitter Follower-Followee DataSet

PageRank Calculation(Module 1:Graphchi API)

Module2(Sorting PageRanks)

Module3(Fetch Top K Users)

Top K Users

Tweets Crawler

(Module4: Selenium,twython,twitter API)

Tweets

Topic Modelling(Module5:Mallet Tool)

LDA Model

Problems:

- We encountered a case where all the influential elements could belong to a single community.
- Then LDA on the top k influential elements would not result in a generic model, hence this technique is not robust and effective for this scenarios.
- Hence we had to move on to the next approach of community detection before applying the LDA.

Approach 3… and future works :

- Since Approach 2 was not giving expected results, so we tried Community Detection approach.
- The GraphChi Community Detection code is still in early phase of development and is not giving expected results.
- There is another API named Infomap which tends to work well for small graphs but the code burst for large graphs.

- We are still working on approach 3 and aiming for better accuracy and results.

Thank you ☺