CSE3021 – Social and Information Network (SIN)

Project Report

Identification of Conflicting Communities on Twitter using SNA and NLP

By

19BCE1180 Avireddy NVSRK Rohan

B. Tech Computer Science and Engineering

Submitted to

Prof.Noel Jeygar Robert V

School of Computer Science and Engineering

Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

June 2022

<u>DECLARATION</u>

I hereby declare that the report titled "**Identification of Conflicting Communities on Twitter using SNA and NLP"** submitted by me to VIT Chennai is a record of bona-fide work undertaken by me under the supervision of **Prof.Noel Jeygar Robert V** School of Computer Science and Engineering, Vellore Institute of Technology, Chennai.

Signature of the Candidate

Avireddy NVSRK Rohan

Reg. No. 19BCE11180

CERTIFICATE

Certified that this project report entitled "Identification of Conflicting Communities on Twitter using SNA and NLP**"** is a bonafide work of Avireddy NSRK Rohan 19BCE1180 and he carried out the project work under my supervision and guidance for CSE3021 – Social and Information Network (SIN)

Prof.Noel Jeygar Robert V

SCOPE, VIT Chennai

ACKNOWLEDGEMENT

I would like to express my sincere thanks and gratitude to my SIN project guide Prof.Noel Jeygar Robert V for giving me and my teammates the opportunity to work on this project. I am very grateful for the support and guidance in completing this project. I would also like to thank all my dear friends and other acquaintances who have aided us in completing this project.

Avireddy NVSRK Rohan

Reg. No. 19BCE1180

ABSTRACT

Hate and offensive environment online is detrimental for users of the platform and can lead to various consequences which can endanger the sanity of the users and will eventually lead to the loss of users from the platform. So far, most solutions to solve this problem plainly look at it from the NLP standpoint. However early However early detection of possible conflict can provide us ample time for the platform to devise solutions to prevent hate. This pro project uses a cascaded approach of SNA and NLP to detect communities that currently may have a possible conflict in the future. We employ an integrative computational framework comprising Community detection, text summarization, conflict sentence detection to map out and visualize communities that can have a conflict with each other. This is a new perspective toward solving the greater problem.

CONTENTS

1. Introduction

1.1 **Objective and goal of the project**

People on social media connect with each other because of various factors; be it they follow the same person, have similar political, financial interests etc.
Online communities varying from the size of just 100 people to hundreds of thousands of people exist on Twitter, Facebook, etc.

Sometimes, people don't agree on the same thing and end up having a verbal dispute over the internet.

This can sometimes lead to large-scale hate among communities. Consequently, this can lead to hateful and offensive exchange of messages between people belonging to different communities

.

1.2 **Problem Statement**

The proposed technique uses social network analysis techniques to first identify communities on Twitter. We can then observe and summarize various tweets made by the people within a community using Extractive text summarization (NLP).

These summaries depict the ideology/ themes of that community.

Now we can use some emerging NLP techniques to identify which communities have conflicting themes.

1.3 **Motivation**

Hate and offensive environment online is detrimental for users of the platform and can lead to various consequences which can endanger the sanity of the users and will eventually lead to the loss of users from the platform. However early detection of possible conflict can provide us ample time for the platform to devise solutions to prevent hate. This pro project uses a cascaded approach of SNA and NLP to detect communities that currently may have a possible conflict in the future.

1.4 **Challenges**

There is a huge lack of publicly accessible research for this problem as this is a very new and novel problem statement. There were not many resources available which actually helped in implementing conflict detection algorithms as very less research has been done regarding it.

2. Literature Survey

[6] the word embedding (Word2Vec) model was trained and human annotators were selected to label texts based on given guidelines and conventions. Hereafter, feature extraction methods using Word2Vec word embedding controlled by TF-IDF, TF-IDF alone, and word n-grams were applied in Apache Spark environment. The classical GBT and RF algorithms and deep learning algorithms including RNN-LSTM and RNN-GRU were compared Word2Vec embedding and RNN-GRU achieved the best performance with an AUC of 97.85% and an accuracy of 92.56%. [5] This paper provides a modified sentiment analysis method for tweets made within a community to show the levels of tension that exists among the commmunities. Tweets are preprocessed then post processed and conventional sentiment analysis were used to classify tweets as low tension, high tension etc.[3] Long short-term memory (LSTM) and Global Vectors for Word Representation (GloVe). approach for detecting three different types of contradiction: negation, antonyms and numeric mismatch.The accuracy of classifier based on both LSTM and manual features for the SemEval dataset is 91.2%. The classifier was able to correctly classify 3204 out of 3513 instances. The accuracy of classifier based on both LSTM and manual features for the Stanford dataset is 71.9%. The classifier was able to correctly classify 855 out of 1189 instances. The accuracy for the PHEME dataset is the highest across all datasets. The accuracy for the contradiction class is 96.85%.[4] This paper presented a novel NLU task of identifying a counter speech, which best counters an input speech, within a set of candidate counter speeches. Used BERT to generate document embeddings experiments suggest that the best results are achieved using Jensen-Shannon similarity, for speeches that contain explicit responses (accuracy of 80%) and using conditional mutual information on speeches that respond to the input speech in an implicit way (accuracy of 43%).

3 Requirements Specification

3.1 Hardware Requirements

The back-end server should run on a processor with a minimum clock rate of 1GHz with 512MB of RAM and an external storage of 20MB database space to store webpage data along with 300MB disk space for database storage.
A device with a minimum of 8 GB ram is needed to run NLP models to extract and structure data.

3.2 Software Requirements

A simple latest windows software, Python 3 , Jupyter Notebook. The experiment can be easily replicated with Google Colab as well. Twitter API V2 elevated access was made use. Tweepy client version 4.

4 System Design

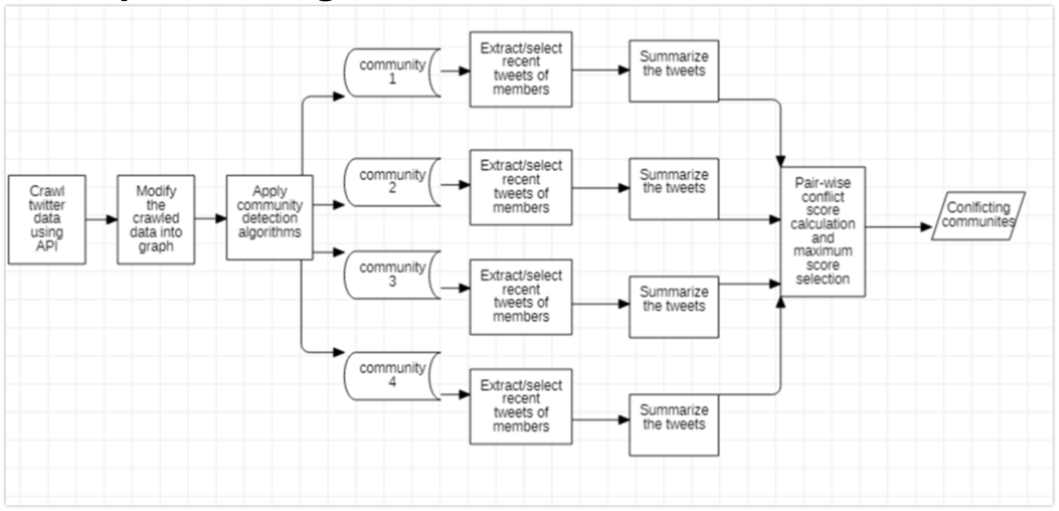
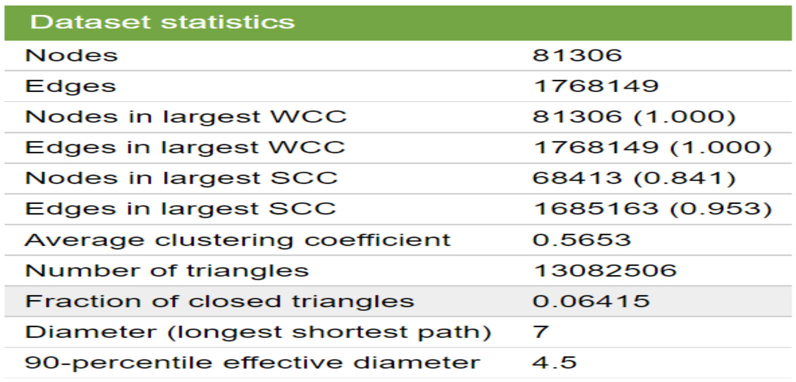

Crawl twitter data using API
Modify the crawled data into graph
Apply community detection algorithms
community 1
community 2
community 3
community 4
Extract/select recent tweets of members
Summarize the tweets
Pair-wise conflict score calculation and maximum score selection
Conflicting communites

Figure 1

This section gives a detailed design description of the proposed system. First we divide the the system into 4 modules

Modules

1. **Network Data collection**
   a. We need elevated twitter api v2 access to crawl data from the twitter website.
   b. But the network data that we will be using has been obtained from the Stanford Network Analysis Project (SNAP).



Dataset statistics	
Nodes	81306
Edges	1768149
Nodes in largest WCC	81306 (1.000)
Edges in largest WCC	1768149 (1.000)
Nodes in largest SCC	68413 (0.841)
Edges in largest SCC	1685163 (0.953)
Average clustering coefficient	0.5653
Number of triangles	13082506
Fraction of closed triangles	0.06415
Diameter (longest shortest path)	7
90-percentile effective diameter	4.5

Figure 2

The dataset contains ego networks of various users indicated with their user name.

The data is however first collected in 2012. So to only use the active account data we refined and picked 10 random ego networks whose most recent activity on twitter is no older than 3 years (>=2019).

2. **Community Detection**
   1. **Louvain community detection** was originally proposed in 2008 as a fast community unfolding method for large networks. This approach is based on modularity, which tries to maximize the difference between the actual number of edges in a community and the expected number of edges in the community. Sample code:
   2.
   ```
   from cdlib import algorithms
   import networkx as nx
   G = nx.karate_club_graph()
   coms = algorithms.louvain(G, weight='weight', resolution=1., randomize=False)
   ```

   3. **edge.betweenness.community** is a hierarchical decomposition process where edges are removed in the decreasing order of their edge betweenness scores (i.e. the number of shortest paths that pass through a given edge).
   4. This method yields good results but is very slow because of the computational complexity of edge betweenness calculations and because the betweenness scores have to be re-calculated after every edge removal
   5. Another disadvantage is that edge.betweenness.community builds a full dendrogram and does not give you any guidance about where to cut the dendrogram to obtain the final groups

   6. **fastgreedy.community** is another hierarchical approach, but it is bottom-up instead of top-down. It tries to optimize a quality function called modularity in a greedy manner. Initially, every vertex belongs to a separate community, and communities are merged iteratively such that each merge is locally optimal (i.e. yields the largest increase in the current value of modularity).

7. The algorithm stops when it is not possible to increase the modularity any more, so it gives you a grouping as well as a dendrogram.
8. However, it is known to suffer from a resolution limit, i.e. communities below a given size threshold (depending on the number of nodes and edges) will always be merged with neighboring communities.

9. **walktrap.community** is an approach based on random walks. The general idea is that if you perform random walks on the graph, then the walks are more likely to stay within the same community because there are only a few edges that lead outside a given community.
10. It is a bit slower than the fast greedy approach but also a bit more accurate (according to the original publication).

Louvain community detection is fit for our experiment as it is highly scalable.

3. Tweet Concatenation and Summarization

BART, a denoising autoencoder for pretraining sequence-to-sequence models. BART is trained by corrupting text with an arbitrary noising function, and learning a model to reconstruct the original text. It uses a standard Transformer-based neural machine translation architecture which, despite its simplicity, can be seen as generalizing BERT (due to the bidirectional encoder), GPT (with the left-to-right decoder), and many other more recent pretraining schemes.

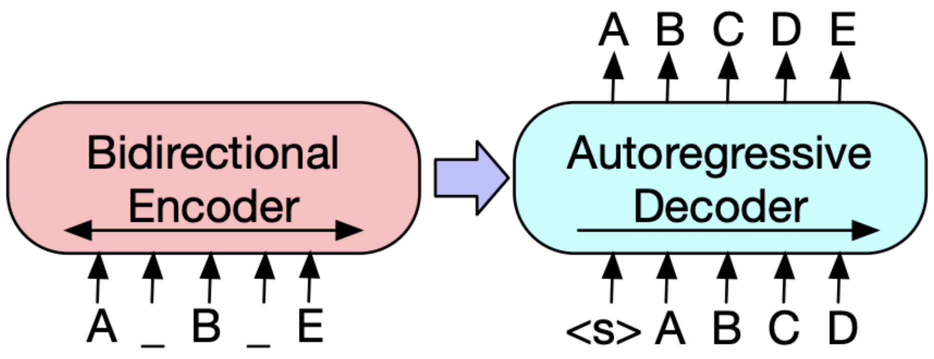


Figure 3

4. Conflicting community detection

Detecting conflicting statements may be a foundational text understanding task with applications in info .We tend to propose an applicable definition of contradiction for information science tasks and develop out there corpora, from that we construct a categorization of contradictions. we tend to demonstrate that a system for contradiction must create a lot of fine-grained distinctions than the common systems for entailment. In particular, we argue for the spatial relation of event grammatical relation and so incorporate such an element supported topicality. we tend to gift the primary careful breakdown of performance on this task. sleuthing some kinds of contradiction needs deeper inferential methods than our system is capable of, however, we tend to come through} good performance on sorts arising from negation and semantic relation

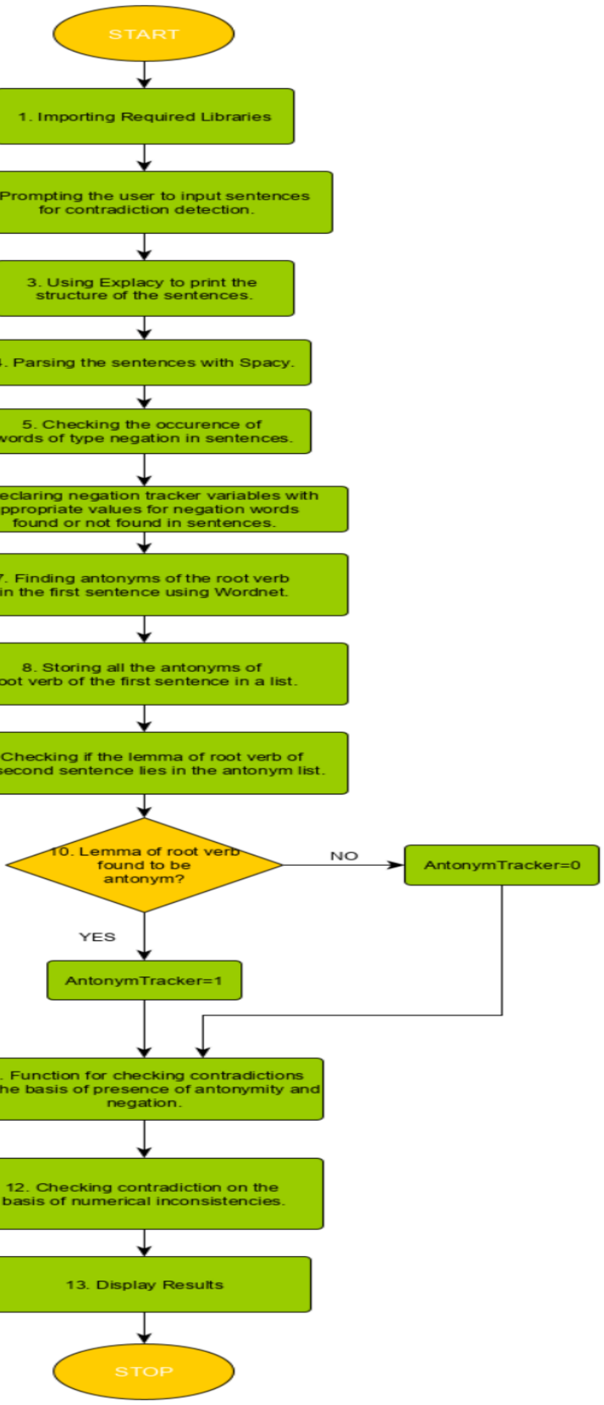

START
1. Importing Required Libraries
2. Prompting the user to input sentences for contradiction detection.
3. Using Explacy to print the structure of the sentences.
4. Parsing the sentences with Spacy.
5. Checking the occurence of words of type negation in sentences.
6. Declaring negation tracker variables with appropriate values for negation words found or not found in sentences.
7. Finding antonyms of the root verb in the first sentence using Wordnet.
8. Storing all the antonyms of root verb of the first sentence in a list.
9. Checking if the lemma of root verb of the second sentence lies in the antonym list.
10. Lemma of root verb found to be antonym?
NO
AntonymTracker=0
YES
AntonymTracker=1
11. Function for checking contradictions on the basis of presence of antonymity and negation.
12. Checking contradiction on the basis of numerical inconsistencies.
13. Display Results
STOP

Figure 4

5 Implementation of System

Libraries required:

1. **NLTK:**

We used NLTK using python environment to implement our code. NLTK stands for Natural Language Toolkit. NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum. This toolkit is one of the most powerful NLP libraries which contains packages to make machines understand human language and reply to it with an appropriate response. It is one of the most suitable and usable libraries in Natural Language Processing.

2. **spaCy**

spaCy is a free, open-source library for advanced Natural Language Processing in Python. If you're working with a lot of text, you'll eventually want to know more about it. For example, what's it about? What do the words mean in context? Who is doing what to whom? What companies and products are mentioned? Which texts are similar to each other? All these questions can be answered easily with spaCy which is designed specifically for production use and helps you build applications that process and understand large volumes of text.

Once we have the network data we run the louvain community detection algorithm on the network data collected.

3. Tensorflow
4. Keras
5. Scikit Learn
6. HuggingFace
7. Transformers

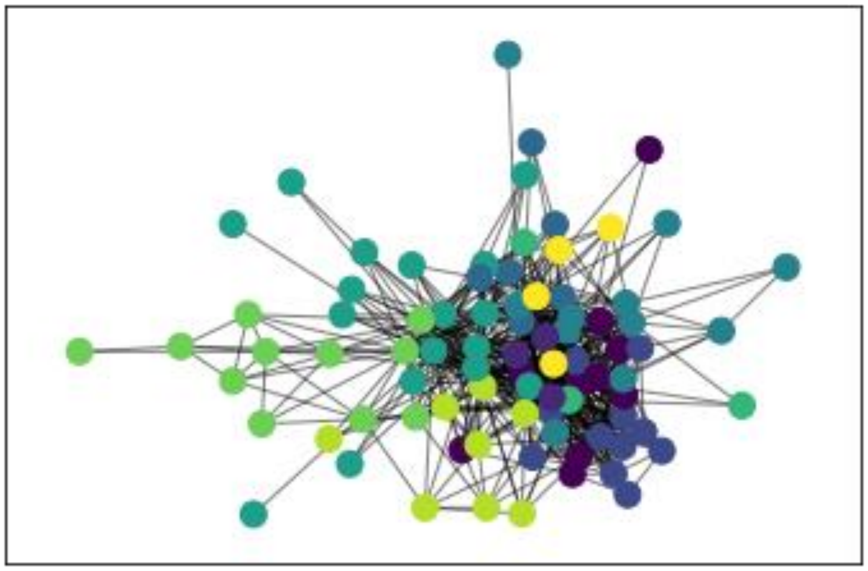


Figure 5: Ego network of 402970625 after com detection

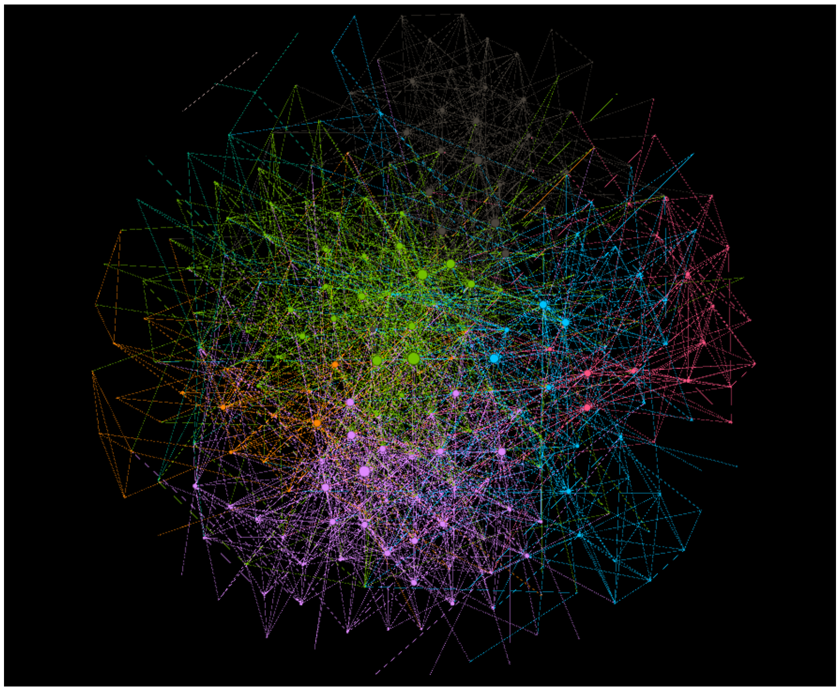


Figure 6 Ego network of 12831 visualized using Gephi after com detection

The above figures show the visualization after applying louvain community detection on two different ego networks; Now we get the community/cluster labels of each node

We proceed to collect tweets from each node within a community and we continue to do that for every community.

CLEANING AND PREPROCESSING

All the sentences provided in the data were tokenized at word level. However, the data was taken from twitter so the text data contains various types of noise , like usernames that start with @ and urls, emojis, multiple punctuations like !!!! etc. These have to be filtered from the text.We have filtered out non-informative features from the tweets like URLs, white spaces,usernames that start with@ and hashtags. Other features like emojis have been filtered out.The words have been decontracted. Words like won't,don't,can't,he'll, I'll etc have been converted to their complete forms. Stop words have been preserved as this is a language identification task..In addition, we have performed lemmatization of the preprocessed text data

Now we approach to the tweet collection stage:

TWEET COLLECTION HEURISTICS:

1. get 20 tweets per person in every community

2. put them all into a data frame

3. clean and preprocess those tweets

4. now concatenate all of them tweets

5. now perform BART text summarization on them to generate a summary for every community

MODIFICATIONS*

to be performed between step 3 and step 4

1. execute Latent Dirichlet allocation on the cleaned tweets to assign topic label

2. get the topic label that is in majority now take the most frequent word used in the tweets belonging to that topic and assign that word as the topic and also as the community's name

-----

CORRECTION OF POINT 2: RATHER THAN TRYING GET THE MAJORITY TOPIC FOR A COMMUNITY. USE THE LIST RETURNED BY THE LDA MODEL AND REMOVE STOP WORDS FROM THAT LIST AND USE THE REMAINING WORDS TO PERFORM THE SEARCH AGAIN

-----

3. now get the list of top 5 or top 10 frequent words in the topic based on frequency
4. now again perform a search for 20 tweets per person containing these the top word

Topically Modeled tweet querying using Latent Dirichlet Allocation
A tool and technique for Topic Modeling, Latent Dirichlet Allocation (LDA) classifies or categorizes the text into a document and the words per topic, these are modeled based on the Dirichlet distributions and processes.

The LDA makes two key assumptions:

1. Documents are a mixture of topics, and
2. Topics are a mixture of tokens (or words)

And, these topics using the probability distribution generate the words. In statistical language, the documents are known as the probability density (or distribution) of topics and the topics are the probability density (or distribution) of words.

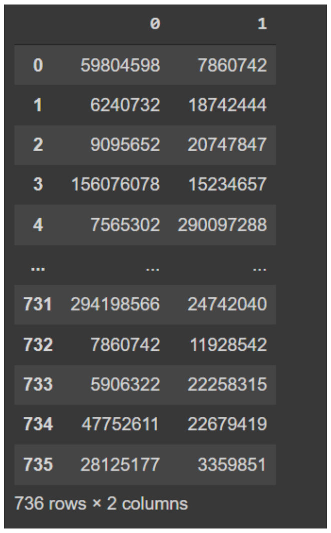
Once we get the topics for each community. We extract all the important keywords and use these to perform a second refined search for tweets which contain only these keywords.

These reduces topical drift and maintains coherency in the concatenated tweets.

Once a second search is performed and new tweets from each community are extracted. Now we simply clean, preprocess for noise and concatenate them into a single paragraph. Now the output from this phase is fed into a BART summarizer which uses extractive text summarization and picks out the most relevant tweets. Once we have that now we are ready to apply our previously discussed Contradiction detection algorithm

6. Results and Outputs

- Ego Network



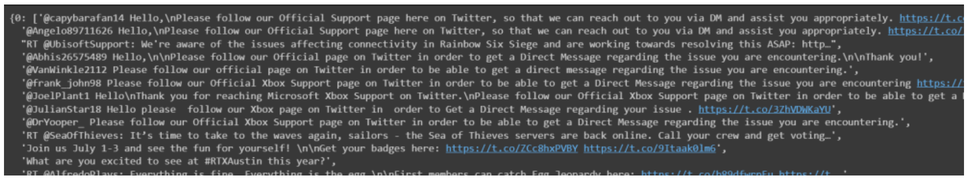
	0	1
0	59804598	7860742
1	6240732	18742444
2	9095652	20747847
3	156076078	15234657
4	7565302	290097288
...	...	...
731	294198566	24742040
732	7860742	11928542
733	5906322	22258315
734	47752611	22679419
735	28125177	3359851

736 rows × 2 columns

- Louvain Community Detection output

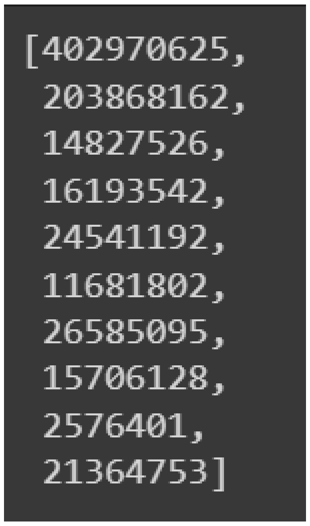
  {59804598: 0, 7860742: 0, 6240732: 2, 18742444: 2, 9095652: 2, 20747847: 2, 156076078: 3, 15234657: 2, 7565302: 2, 290097288: 4, 46146200: 2, 289738351: 5, 156588288: 3, 18927441: 0, 47161442: 0, 3359851: 6, ...

- Sample Tweets collected per community



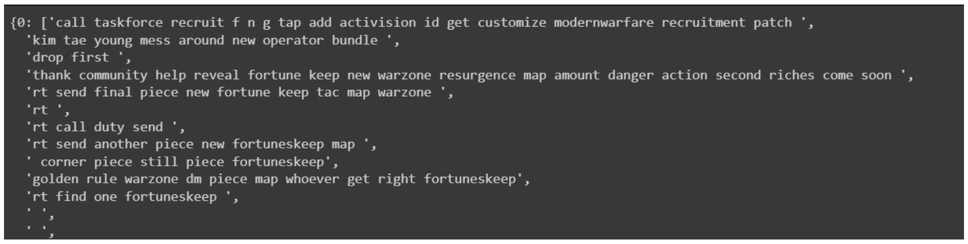
```
{0: ['@capybarafan14 Hello,\nPlease follow our Official Support page here on Twitter, so that we can reach out to you via DM and assist you appropriately. https://t.c
  '@Angelo89711626 Hello,\nPlease follow our Official Support page here on Twitter, so that we can reach out to you via DM and assist you appropriately. https://t.co/
  "RT @UbisoftSupport: We're aware of the issues affecting connectivity in Rainbow Six Siege and are working towards resolving this ASAP: http…",
  '@Abhis26575489 Hello,\n\nPlease follow our Official page on Twitter in order to get a Direct Message regarding the issue you are encountering.\n\nThank you!',
  '@VanWinkle2112 Please follow our official page on Twitter in order to be able to get a direct message regarding the issue you are encountering.',
  '@frank_john98 Please follow our Official Xbox Support page on Twitter in order to be able to get a Direct Message regarding the issue you are encountering https://
  '@JoelPlant1 Hello\nThank you for reaching Microsoft Xbox Support on Twitter.\nPlease follow our Official Xbox Support page on Twitter in order to be able to get a
  '@JulianStar18 Hello please  follow our Xbox page on Twitter in  order to Get a Direct Message regarding your issue . https://t.co/3ZhVDWKaYU',
  '@DrYooper_ Please follow our Official Xbox Support page on Twitter in order to be able to get a Direct Message regarding the issue you are encountering.',
  'RT @SeaOfThieves: It's time to take to the waves again, sailors - the Sea of Thieves servers are back online. Call your crew and get voting…',
  'Join us July 1-3 and see the fun for yourself! \n\nGet your badges here: https://t.co/ZCc8hxPVBY https://t.co/9Itaak0lm6',
  'What are you excited to see at #RTXAustin this year?',
  'RT @AlfredoPlays: Everything is fine. Everything is the egg.\n\nFirst members can catch Egg Jeopardy here: https://t.co/h89dfwrnFu https://t…',
```

- Random Nodes selected



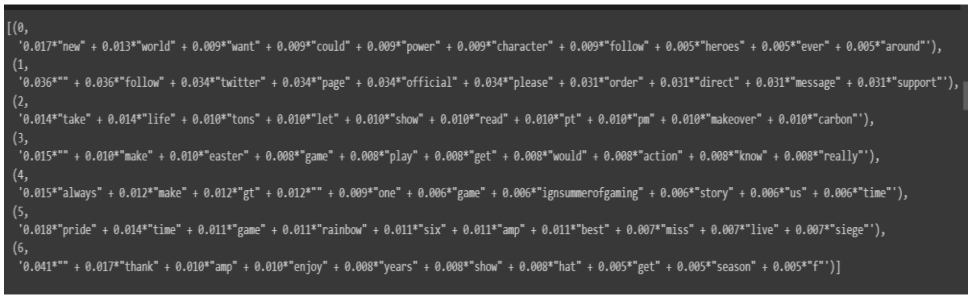
```
[402970625,
 203868162,
 14827526,
 16193542,
 24541192,
 11681802,
 26585095,
 15706128,
 2576401,
 21364753]
```

- Cleaned and preprocessed tweets



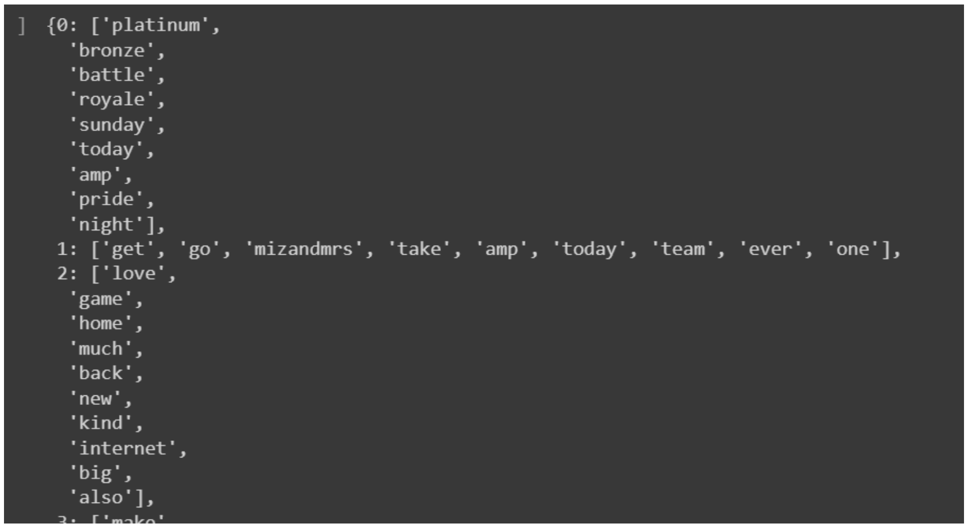
```
{0: ['call taskforce recruit f n g tap add activision id get customize modernwarfare recruitment patch ',
  'kim tae young mess around new operator bundle ',
  'drop first ',
  'thank community help reveal fortune keep new warzone resurgence map amount danger action second riches come soon ',
  'rt send final piece new fortune keep tac map warzone ',
  'rt ',
  'rt call duty send ',
  'rt send another piece new fortuneskeep map ',
  ' corner piece still piece fortuneskeep',
  'golden rule warzone dm piece map whoever get right fortuneskeep',
  'rt find one fortuneskeep ',
  ' ',
  ' ',
```

- 7 topics modelled using LDA



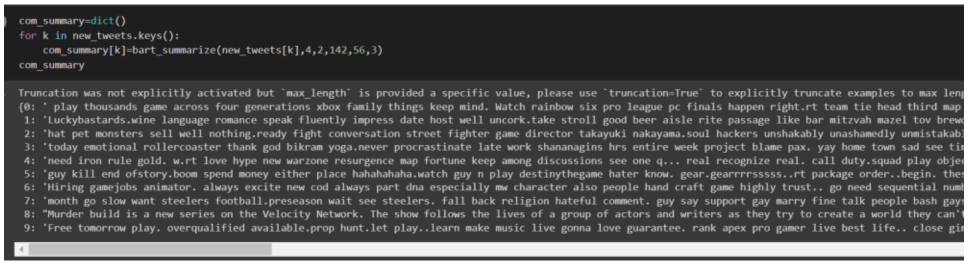
```
[(0,
  '0.017*"new" + 0.013*"world" + 0.009*"want" + 0.009*"could" + 0.009*"power" + 0.009*"character" + 0.009*"follow" + 0.005*"heroes" + 0.005*"ever" + 0.005*"around"'),
 (1,
  '0.036*"" + 0.036*"follow" + 0.034*"twitter" + 0.034*"page" + 0.034*"official" + 0.034*"please" + 0.031*"order" + 0.031*"direct" + 0.031*"message" + 0.031*"support"'),
 (2,
  '0.014*"take" + 0.014*"life" + 0.010*"tons" + 0.010*"let" + 0.010*"show" + 0.010*"read" + 0.010*"pt" + 0.010*"pm" + 0.010*"makeover" + 0.010*"carbon"'),
 (3,
  '0.015*"" + 0.010*"make" + 0.010*"easter" + 0.008*"game" + 0.008*"play" + 0.008*"get" + 0.008*"would" + 0.008*"action" + 0.008*"know" + 0.008*"really"'),
 (4,
  '0.015*"always" + 0.012*"make" + 0.012*"gt" + 0.012*"" + 0.009*"one" + 0.006*"game" + 0.006*"ignsummerofgaming" + 0.006*"story" + 0.006*"us" + 0.006*"time"'),
 (5,
  '0.018*"pride" + 0.014*"time" + 0.011*"game" + 0.011*"rainbow" + 0.011*"six" + 0.011*"amp" + 0.011*"best" + 0.007*"miss" + 0.007*"live" + 0.007*"siege"'),
 (6,
  '0.041*"" + 0.017*"thank" + 0.010*"amp" + 0.010*"enjoy" + 0.008*"years" + 0.008*"show" + 0.008*"hat" + 0.005*"get" + 0.005*"season" + 0.005*"f"')]
```

- Sample important words per community



```
]  {0: ['platinum',
     'bronze',
     'battle',
     'royale',
     'sunday',
     'today',
     'amp',
     'pride',
     'night'],
    1: ['get', 'go', 'mizandmrs', 'take', 'amp', 'today', 'team', 'ever', 'one'],
    2: ['love',
     'game',
     'home',
     'much',
     'back',
     'new',
     'kind',
     'internet',
     'big',
     'also'],
    3: ['make'
```

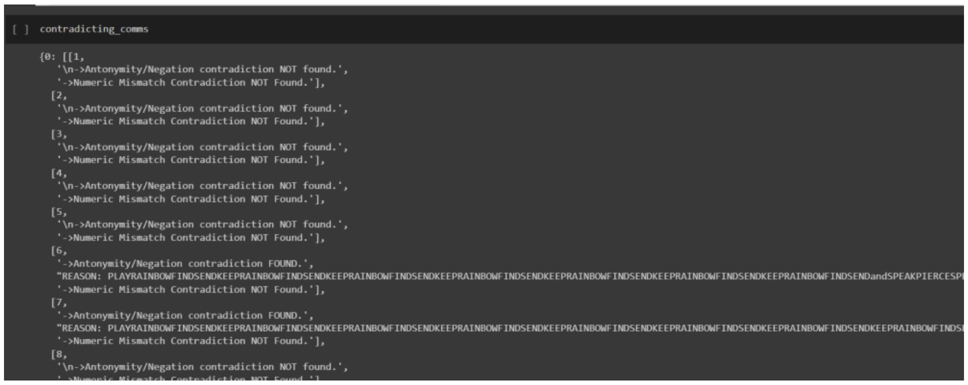
- BART Summary per community



```
com_summary=dict()
for k in new_tweets.keys():
    com_summary[k]=bart_summarize(new_tweets[k],4,2,142,56,3)
com_summary
```

```
Truncation was not explicitly activated but `max_length` is provided a specific value, please use `truncation=True` to explicitly truncate examples to max leng
{0: ' play thousands game across four generations xbox family things keep mind. Watch rainbow six pro league pc finals happen right.rt team tie head third map
 1: 'Luckybastards.wine language romance speak fluently impress date host well uncork.take stroll good beer aisle rite passage like bar mitzvah mazel tov brew
 2: 'hat pet monsters sell well nothing.ready fight conversation street fighter game director takayuki nakayama.soul hackers unshakably unashamedly unmistakab
 3: 'today emotional rollercoaster thank god bikram yoga.never procrastinate late work shananagins hrs entire week project blame pax. yay home town sad see ti
 4: 'need iron rule gold. w.rt love hype new warzone resurgence map fortune keep among discussions see one q... real recognize real. call duty.squad play obje
 5: 'guy kill end ofstory.boom spend money either place hahahahaha.watch guy n play destinythegame hater know. gear.gearrrrsssss..rt package order..begin. the
 6: 'Hiring gamejobs animator. always excite new cod always part dna especially mw character also people hand craft game highly trust.. go need sequential num
 7: 'month go slow want steelers football.preseason wait see steelers. fall back religion hateful comment. guy say support gay marry fine talk people bash gay
 8: "Murder build is a new series on the Velocity Network. The show follows the lives of a group of actors and writers as they try to create a world they can'
 9: 'Free tomorrow play. overqualified available.prop hunt.let play..learn make music live gonna love guarantee. rank apex pro gamer live best life.. close gi
```

- Conflicting communities detection



```
[ ]  contradicting_comms
```

```
{0: [[1,
   '\n->Antonymity/Negation contradiction NOT found.',
   '->Numeric Mismatch Contradiction NOT Found.'],
  [2,
   '\n->Antonymity/Negation contradiction NOT found.',
   '->Numeric Mismatch Contradiction NOT Found.'],
  [3,
   '\n->Antonymity/Negation contradiction NOT found.',
   '->Numeric Mismatch Contradiction NOT Found.'],
  [4,
   '\n->Antonymity/Negation contradiction NOT found.',
   '->Numeric Mismatch Contradiction NOT Found.'],
  [5,
   '\n->Antonymity/Negation contradiction NOT found.',
   '->Numeric Mismatch Contradiction NOT Found.'],
  [6,
   '->Antonymity/Negation contradiction FOUND.',
   "REASON: PLAYRAINBOWFINDSENDKEEPRAINBOWFINDSENDKEEPRAINBOWFINDSENDKEEPRAINBOWFINDSENDKEEPRAINBOWFINDSENDKEEPRAINBOWFINDSENDKEEPRAINBOWFINDSENDandSPEAKPIERCESP
   '->Numeric Mismatch Contradiction NOT Found.'],
  [7,
   '->Antonymity/Negation contradiction FOUND.',
   "REASON: PLAYRAINBOWFINDSENDKEEPRAINBOWFINDSENDKEEPRAINBOWFINDSENDKEEPRAINBOWFINDSENDKEEPRAINBOWFINDSENDKEEPRAINBOWFINDSENDKEEPRAINBOWFINDSENDKEEPRAINBOWFINDS
   '->Numeric Mismatch Contradiction NOT Found.'],
  [8,
   '\n->Antonymity/Negation contradiction NOT found.',
   '->Numeric Mismatch Contradiction NOT Found.']
```

The LDA modelling actually helped in retrieving relevant tweets compared to just regular crawling. This helped us in giving a meaning to the community theme. Louvain community detection has better resolution when it comes to medium-sized ego networks.. So the algorithm gave the output very quickly. Several experimentations concluded that the quality of the input to the summarizer algorithm increased as a result of using a keyword-based refined search. Initially, 20 tweets per person were crawled. The consolidated form of this input is provided to the LDA topic model. The number of topics was set to 7. Since the latest upgrade, some functions of the new API are yet to be released. So there could be an improvement in that aspect. BART summarizer is an extractive summarizer, hence it directly picks the important sentences. If there was a numerical mismatch contradiction or antonym-based contradiction between the summaries of the communities then the those two communities are considered as communities with conflicting ideas.

7. Conclusion and Future Work

To assist social networking platform developers to detect large-scale hate and offensive environment on their platform,, a complex yet sophisticated solution of using community detection,, tweet summarization, and contradiction detection were devised. The Twitter ego network data has been obtained from the SNAP website.10 random active nodes were selected and their ego networks have been taken into consideration for experimentation. In between, a small modification was made while collecting tweets. Using a Latent Dirichlet allocation method, provided us with a list of keywords that are important to classify the documents was used. This fact, provided a more richer output from the summarizer, compared to normal tweets concatenated as a single paragraph. Putting the ideology or theme of a community into a sentence has been made possible through this.However,this framework is not ready to deal with tweets in languages other than English for now, so once this method is fine-tuned, cross lingual solutions can be researched. Now, this input has been provided to the contradiction detection algorithm. In the future we could improve the results by testing out other methods to refine the tweet search and using abstractive text summarization instead of extractive could achieve a more semantically sound summary.

8. REFERENCES

[1] Uyheng, J. and Carley, K.M., 2021. Characterizing network dynamics of online hate communities around the COVID-19 pandemic. Applied Network Science, 6(1), pp.1-21.

[2] Mittal, R., & Bhatia, M. P. S. (2021). Classification and comparative evaluation of community detection algorithms. Archives of Computational Methods in Engineering, 28(3), 1417-1428.

[3] Lingam, V., Bhuria, S., Nair, M., Gurpreetsingh, D., Goyal, A., & Sureka, A. (2018). Deep learning for conflicting statements detection in text (No. e26589v1). PeerJ Preprints.

[4] Orbach, M., Bilu, Y., Toledo, A., Lahav, D., Jacovi, M., Aharonov, R., & Slonim, N. (2020). Out of the echo chamber: Detecting countering debate speeches. arXiv preprint arXiv:2005.01157.

[5] Burnap, P., Rana, O. F., Avis, N., Williams, M., Housley, W., Edwards, A., ... & Sloan, L. (2015). Detecting tension in online communities with computational Twitter analysis. Technological Forecasting and Social Change, 95, 96-108.

[6] Mossie, Z., & Wang, J. H. (2020). Vulnerable community identification using hate speech detection on social media. Information Processing & Management, 57(3), 102087.

[7] Client — tweepy 4.10.0 documentation

[8] Twitter Developers

[9] Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. Journal of machine Learning research, 3(Jan), 993-1022.

[10] Chen, Y., & Song, Q. (2021, March). News text summarization method based on bart-textrank model. In 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC) (Vol. 5, pp. 2005-2010). IEEE.