Report
(Sudiksha Aapan)

Assumption:

1) I assume that the class containing the #RestInPeace contains only this tag i.e. no other tag is present. Same I assume this for other tags also.
2) This way I divide the tweets in the three classes.

Q1)

Tweets with #MissYouYuvi = 4606

Tweets with #RestInPeace = 90

Tweets with both above tags = 0

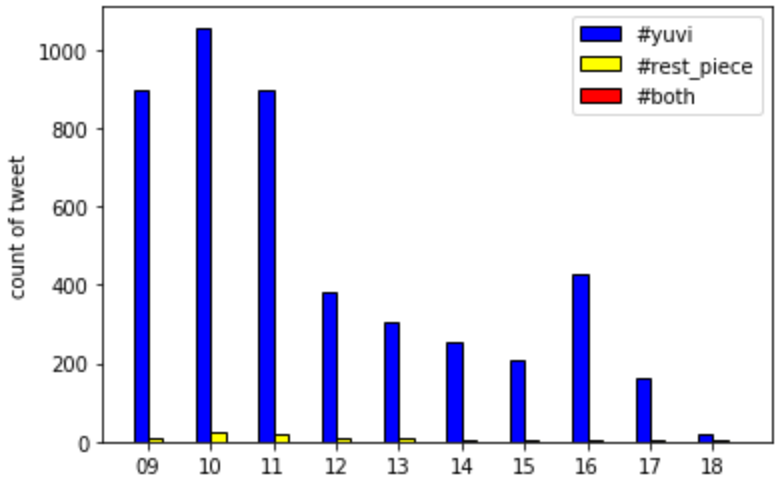
Maybe If I don't consider my assumption more tweets will be coming as more tweets may include more information.

Verified User #MissYouYuvi = 10 , Not verified Users = 1231

Verified User #RestInPeace = 1 , Not verified Users = 83

No users for both tag

Q2) As I observe that tweet is of the same date so I make the count of tweets with each hour.



It contain information of all the tags #MissYouYuvi has the highest tweets whereas #RestInPeace tag has very few tweets .

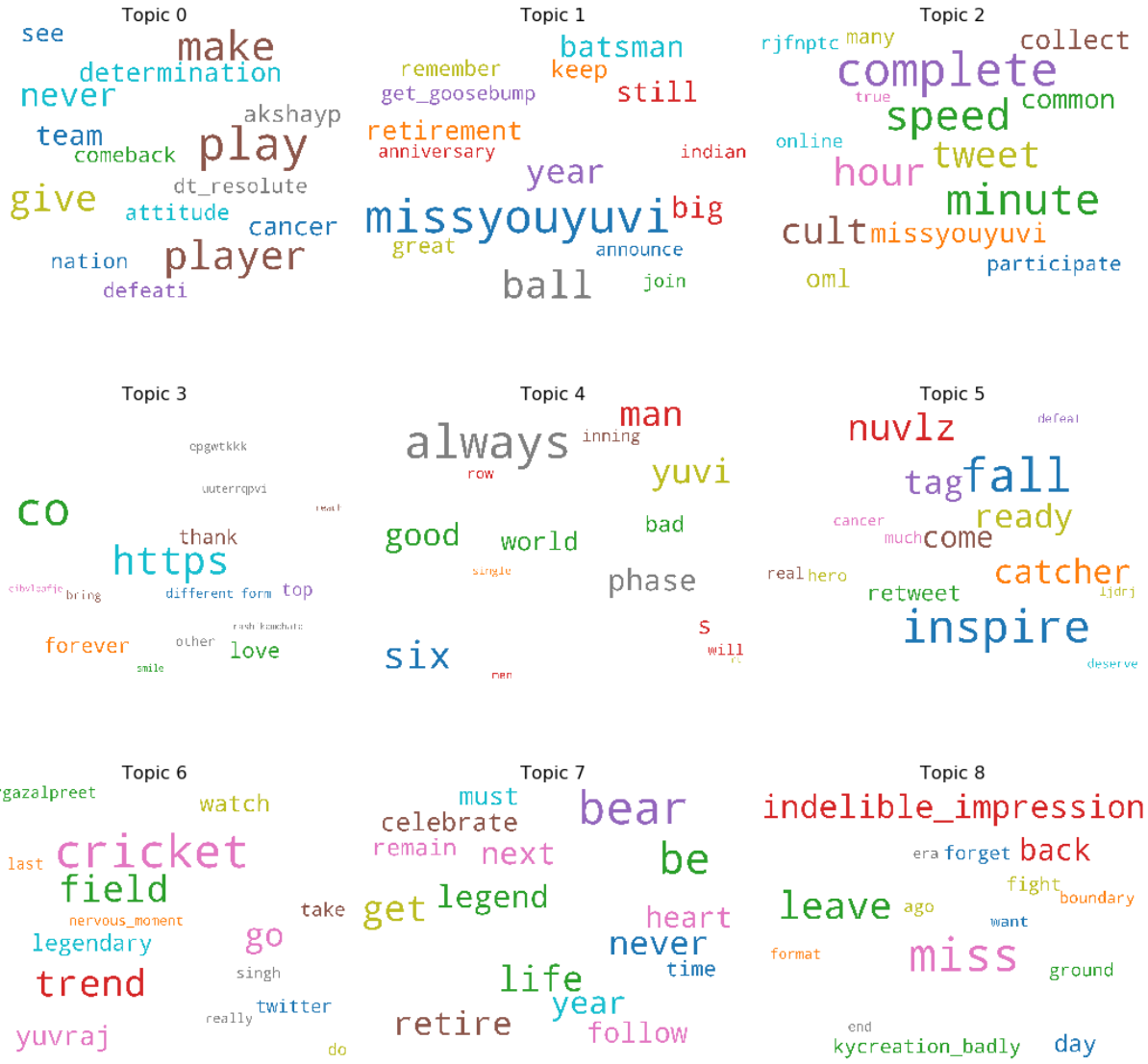
I observe that

As the time progresses Initially tweets increase but then the tweets decrease as most of the users tweet in the initial hour only.
Then after some time the count is almost constant and as the time passes the very less people tweet(means most of the people tweets)
Also this shows the popularity of yuvi is very much between the peoples.

Q3)

Preprocessing:

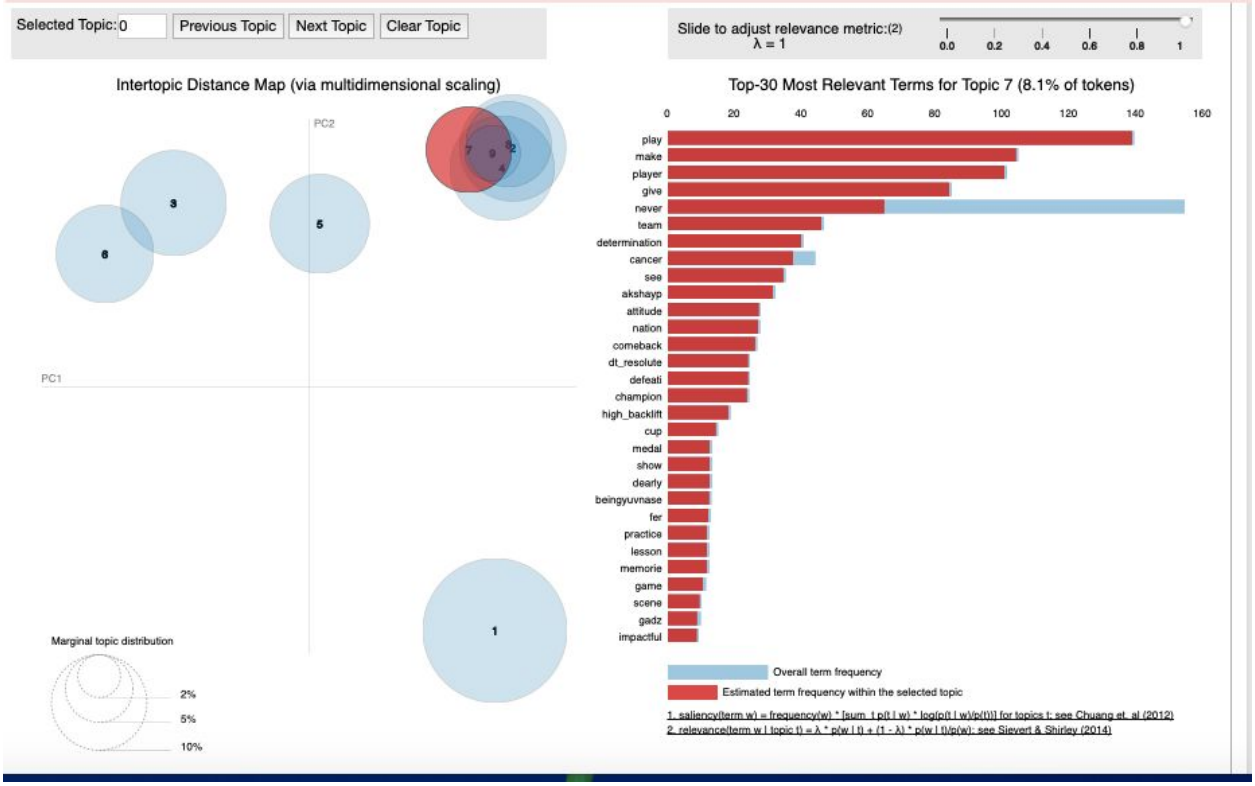
- Remove the stop words from the tweets.
- Remove the inverted commas from the tweet.
- Remove the punctuations
- Do the lemmatization and for this tag also perform POS tagging of the words.
- Also pass the bi gram words in the LDA process as it gives better topic distribution to the corpus.(TF-IDf doesn't improve the performance in the LDA)



These topic almost tell everything about Yuvi:

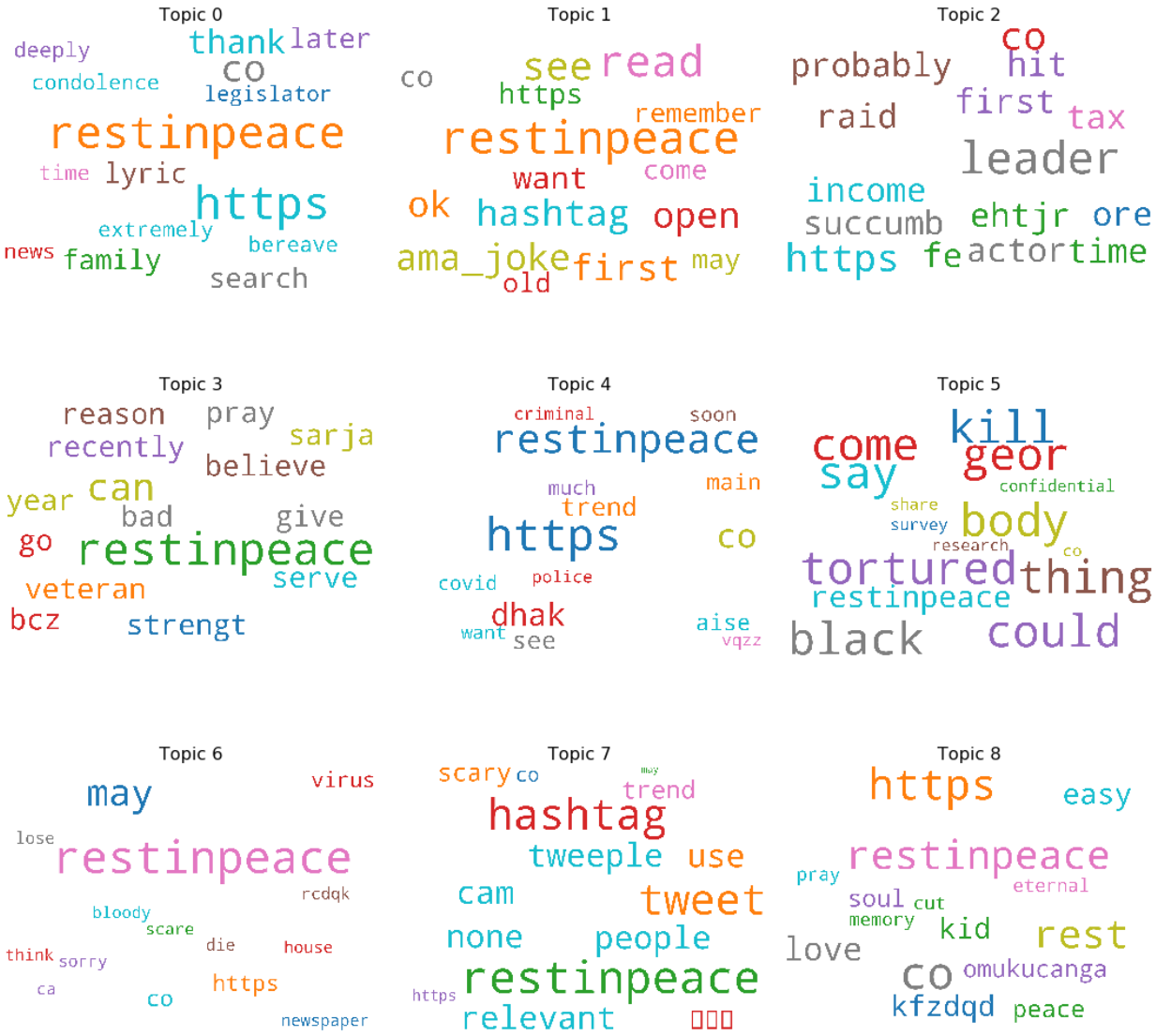
- He is a very determined cricketer.
- While batting once get goose_bump(very good player)
- All them misses him on his retirement day
- He played very good innings of sixes.
- He beats cancer and fight back for this
- He is a big legend and all of them remember them.

- He inspires every one which shows he is very popular.



It shows the word distribution of the topic.

6-7 topics are sufficient for this tag.



Analysis shows:

- People paid condolence to the death of the people and prayed for their soul.
- People are died due to the virus covid -19
- The death of the people due to some tortured to the black people and if we see the place also.
- Due to covid it hit income of the people also.

As I have only 90 tweets in which many Re Tweet also so I can't gather much of the information.

Sentiment Analysis

Tool use is : VADER

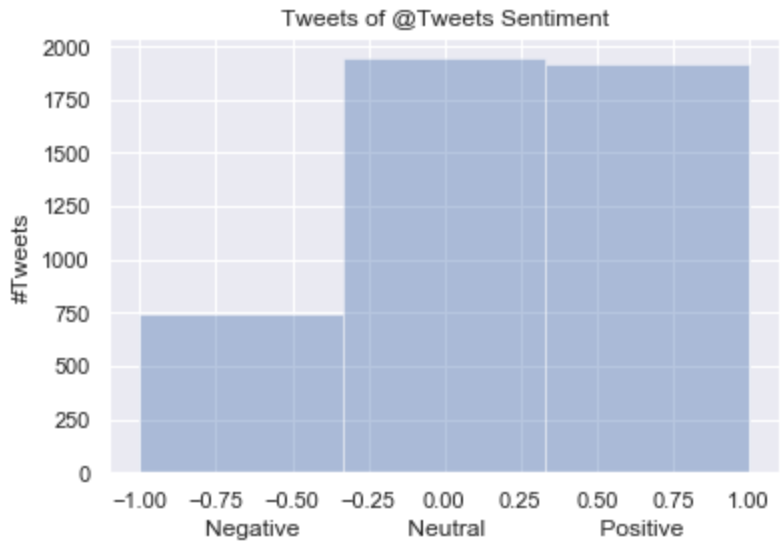
I observe VADER give better performance than TEXT BLOB.

Preprocessing of the tweet :

I have not done much preprocessing as the tool which I use for sentiment analysis shouldn't require much more preprocessing only punctuation is needed.

- remove twitter Return handles (RT @xxx:)
- # remove URL links (httpxxx)
- # remove special characters, numbers, punctuations (except for #)

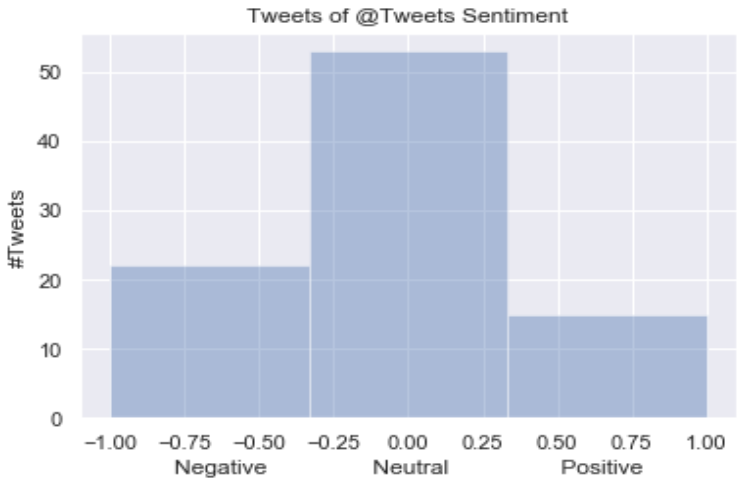
missyou Yuvi



Word Cloud of the tweets

Positive sentiments word cloud of the Yuvi Tag

restinpeace



Q4)

I didn't get any tweet which have both tags only:

- Firstly one is talking about the famous cricketer Yuvraj singh and other is talking about the death of the people so nothing is common between them.

- Secondly others are talking about the death of the black people , tortured on them and death of the people due to the covid so how it can match with the popularity of the cricketer.
- We can match the word cloud of both the tags, nothing common in both of them. So In my opinion that's why no tag is present which is common for them.

Q5)

I didn't able to get the correlation as I tried to extract location

1. From the place tag of the tweet object.
2. From the tweet text using the ner Technique but nothing help it out as I get only 3-4 states which is of very less frequency and some noisy words also.

```
for i in range(len(data)):
    for it in data[0]['entities']['hashtags']:
        if it['text']=='RestInPeace':
            if data[i]['place']!=None:
                print(data[i]['place']['name'])
            else:
                s = data[i]['full_text']
                entities= nlp(s)
                for ent in entities.ents:
                    if ent.label_=='GPE':
                        print(ent.text,ent.label_)
```

Eg.

Yuvi GPE
India GPE
India GPE
India GPE
St. Louis GPE
Seeman GPE
the Republic of Burundi GPE
St. Louis GPE
St. Louis GPE
Seeman GPE

India GPE
India GPE
INDIA GPE