Event Detection using Social Media (Twitter) Text

Jay Chetankumar Patel
Pursuing Master of Computer Science
University of Ottawa
Ottawa, Canada
jpate186@uottawa.ca - 300288159

Jainam Vinaykumar Shah
Pursuing Master of Computer Science
University of Ottawa
Ottawa, Canada
jshah082@uottawa.ca - 300298777

***Abstract*— Social media generates a lot of data that provides insights about the current events around the world. However, manually detecting these events is very difficult, because of the high volume and fast-changing nature of data. Thus event detection is a popular research topic in text mining. Particularly in the context of social media data, Twitter has become a valuable source of information where events are discussed. Previous research on automated event detection is focused on statistical and syntactic features. We have implemented a system which uses tweets and hashtags to extract and filter segments of the context and finally detect events using a clustering method to group similar segments together. We tested our system on Events of 2012 October 12 corpus of 24 hours of tweets. The results show that the system can efficiently and effectively detect events, in comparison to manual detection.**

Keywords—Event Detection, Social Media, Twitter, NLP, segmentation, clustering, KNN

I. Introduction

Social media is increasingly being used to report different kinds of events such as emergencies, or receive information during disasters. So social media texts in a particular network are important and detecting events, topics and user profiles is a crucial task.

Twitter is a social media platform that allows the user to create a tweet of about 280 characters. Twitter is one such platform where the tweets of users often refer to a particular event. Since an enormous amount of data is produced on a social media platform, it becomes a crucial task to gain insights.

On Twitter, the user can not only publish what they think, but they can retweet some other users' posts too. This increases the reach count of that particular tweet. And, this leads to popularising or conveying some news or event. Let us take an example of the latest update in FIFA World Cup 2022, wherein Argentina won in penalty shoot-outs. Almost all the football fans congratulated team Argentina by posting tweets. Through this, people get to know what is going on currently in the world.

Twitter allows users to like the content or retweet or comment on the same post. People can follow each other on Twitter which allows them to have recent updates. The hashtags can help users to search some specific events by just typing the hashtag before the event title. The hashtags are applied in the tweets to give information about the tweet in short segments.

Twitter is one such platform where the tweets of users often refer to a particular event. Since an enormous amount of data is produced on a social media platform, it becomes a crucial task to gain insights.

Event detection is the process of fetching information about the event through any possible platform. This can be done by the analysis of what data is posted by the users, especially on social media.

Thus we have used the Twitter dataset and detected crucial events through the segmentation and clustering process.

The remainder of the project report discusses the related work in this domain in section II, followed by methodology in section III, section IV delivers the dataset used, and the following sections describe the evaluation methodology, results, conclusion, future work, and references.

II. Related Work

There has already been a lot of research done in the field of event detection from tweets using Natural Language Processing (NLP). We have referred to the following research papers to get insights about current works and techniques used that motivated us to do a project on event detection using social media text.

Research paper [1] describes how to detect localized events from the stream of Twitter tweets. They have extracted spatiotemporal characteristics that are used to get logical insights into events. With the help of clustering keywords by specific similarity, they were able to get event information. Once they introduced a scoring technique to give priority, they could get better results. The dataset used was tweets during the 2012 UEFA European Football Championship.

In the study done by [2], they proposed an event detection method called EventRadar, where they used a historical dataset. This method showed a dramatic improvement when compared to the other method where the precision rate was just 25.5% and this method had a precision rate of 68%.

The research work [3] says current methods have mostly concentrated on identifying peaks in clusters surrounding certain keywords. However, one of the primary problems with such methods is that it can be confusing when the same terms are used to describe several things. In this research, they offer a unique method for building periodic event graphs by using Named Entity occurrences in tweets and the entity context. Later on, they analyzed the graphs to find the clusters. This method can automatically detect events without previous ideas.

The extended work in the segmentation manner was done by the researchers [4]. They presented the event detection method called as SEDTWik. Their work's main fundamental concept is to extract bursty chunks from each tweet and hashtag, group them into clusters, and then summarise them. They benchmarked our performance against the renowned Events2012 corpus and attained cutting-edge results.

Automatic event detection in social media texts is quite important and there have been different methods and techniques proposed earlier such as graph theory, rule mining, clustering, and social aspect and in support were various text representations like tokens, vectors, n-grams, etc. which also was used for extraction of keywords such as named entities, noun phrases and hashtags.

In [5], they used a method where the most recurring word is extracted from a large volume of tweets to detect the event. Later on, they paired the words to identify the context of the event in a binary classification manner.

A novel method is proposed in [6] (called Embed2Detect) to detect the event. There, they combine characteristics in word embeddings and hierarchical agglomerative clustering. They tested their work on real social media datasets.

The paper [11] was written by Chenliang Li, Aixin Sun, and Anwitaman Datta, and it proposes a method for detecting events on Twitter using a segment-based approach. The authors aim to develop a system that can identify events in near real-time and handle the large amount of data that is developed on Twitter.

In their approach, the authors first segment the tweets into smaller units and then use a variety of features to represent the segments. These features include the segment itself, the user who tweeted the segment, and the time at which the tweet was posted. The authors also use additional features derived from the structure of the Twitter network, such as the number of followers and the number of retweets.

Once the segments have been represented using these features, the authors use a machine learning algorithm to classify the segments into different event categories. They evaluate the performance of their method on a dataset of tweets collected during the 2012 US presidential election and demonstrate that their approach is effective at detecting events on Twitter.

The paper [15] begins by introducing the problem of studying the public perception and opinion of the COVID-19 pandemic and what is the importance of social media platforms such as Twitter in this context. It then presents a method for collecting and processing Twitter data related to COVID-19 in Brazil and the USA and applying NLP and ML methods to classify and detect the topics and sentiments in this data.

The paper describes the results of the analysis, which shows that the most common topics discussed in relation to COVID-19 on Twitter in Brazil and the USA were related to the spread of the virus, the response of governments and health authorities, and the impact of the pandemic on society and the economy. The results also show that the overall sentiment expressed in these tweets was mostly negative, with a higher percentage of negative tweets in Brazil compared to the USA.

III. Description of the Dataset

The Wikipedia page titles dataset used in this study was a snapshot taken in March 2018 and contained 8,007,358 page titles, where we have titles from Wikipedia with unstemmed phrases.

We also used Wikipedia keyphrases values Q(s) from a dump released on January 30, 2010, which included 4,342,732 distinct entities that appeared as anchor text and use that as title probabilities. These values were previously used by [7].

A Twitter dataset created by [8] called Events2012 containing tweets from Oct 10 - Nov 7, 2012, to exclude spam took tweets with only less than 3 hashtags, 3 name mentions, and 2 URLs. This corpus has a list of events distributed among 8 categories. [4] developed a file with probabilities of segments that we use to extract bursty segments from all the segments.

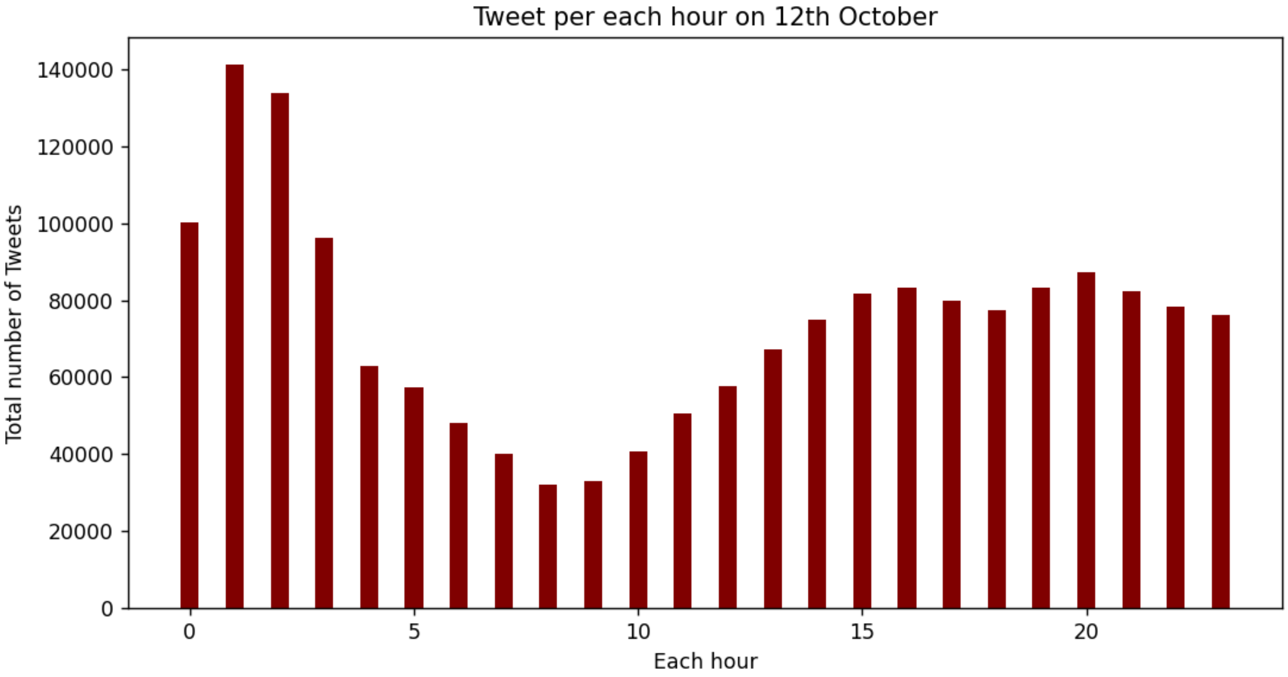
The Twitter dataset on which we detect events is the tweets from Oct 12, 2012 [4].The key attributes of the tweet dataset for October 12, 2012, is available shown in Table 1.

Table 1. Descriptions of the attributes of the tweets.

Keys	**Description**
created_at	Time and Date of the tweet
text	Content of the tweet
user_id	Twitter unique id
user_followers_count	Number of followers of the user

retweet_count	Number of retweets
entities_hashtags	List of hashtags included in the tweet
entities_user_mentions	The usernames of the users mentioned in the tweet

In the tweet segmentation phase, tweets are collected from the Twitter stream within a given time period (t). The tweets are then analyzed and divided into smaller segments, which can be based on a variety of factors such as the words used in the tweets, the hashtags included in the tweets, or the sentiment expressed in the tweets. These segments are indexed, which means that they are organized and stored in a way that makes it easy to retrieve and analyze them in the subsequent stages of the process.



The above figure shows the bar chart of the total number of tweets for each hour on the 12th of October 2012. The x-axis shows the number of hours and the y-axis represents the tweets count. It is clearly depicted that mornings 8 and 9 AM had the lowest count of tweets with 32034 and 33049, respectively, while mornings 2 and 3 AM had the highest count of tweets with 141187 and 133855 count respectively. We had a total of 1765856 tweets on that day.

IV. Methodology

The flowchart shown in figure 2 shows the working flow of the project. So, the system first processes the tweet to extract segments checking if they are present in the wiki titles (unstemmed phrases).

Where in we segment the tweeter text, then we do the bursty segmentation of the segments extracted in the previous step and assign a bursty score using the attributes of the tweets.

Before the last step of event summarization, clustering is done using the concept of K-nearest neighbours. We have referred to the research paper [4] for the same. The output we get is in the form of event clusters with different files.

Hashtags are typically given more weight in this phase because they are often used to indicate the topic or theme of a tweet and, therefore, can be helpful for grouping tweets together and identifying trends or events.

In the next two stages, the segments are analyzed based on various factors such as the probability distribution of the segments (how frequently each segment appears in the tweets), the number of retweets (how many times other users have shared a tweet), the diversity of users (how many different users are tweeting about a particular topic or event), and the popularity of users (how many followers a user has).

Based on this analysis, abnormally bursty segments (segments that show an unusual increase in activity) are identified and clustered together.

This process can be used to identify trends or events on social media platforms such as Twitter, and the resulting information can be helpful for a lot of user needs, such as marketing (to target specific audiences or promote products), news gathering (to identify breaking news or emerging trends), or social research (to study public opinion or behaviour).

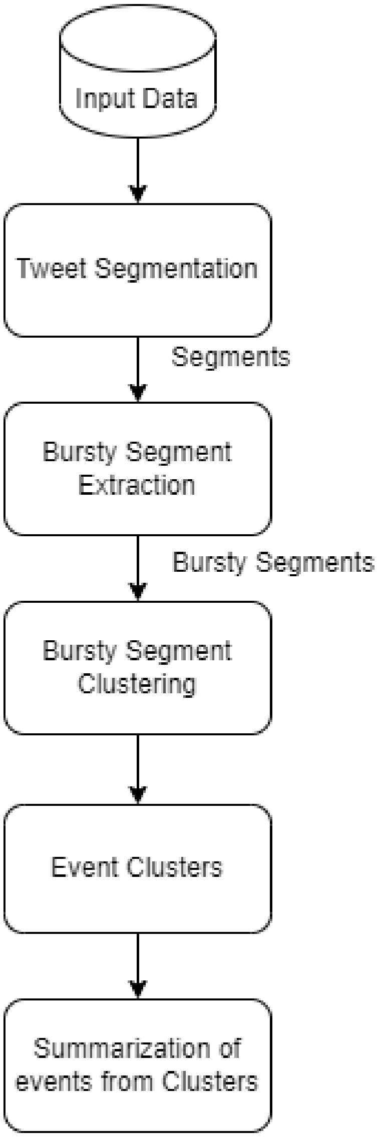

Input Data
Tweet Segmentation
Segments
Bursty Segment Extraction
Bursty Segments
Bursty Segment Clustering
Event Clusters
Summarization of events from Clusters

Fig. 2. Process flowchart

V. Implementation

This section talks in detail about the concepts behind each of the steps undertaken from tweet segmentation to getting the event clusters.

Tweet Segmentation:

- Tweet segmentation is the process of dividing a tweet into smaller segments that contain meaningful information. These segments can be either unigram (single words) or multi-grams (phrases). The primary purpose of tweet segmentation is to extract more specific and meaningful information from a tweet. A multi-gram segment, such as "argentina won fifa 2022," contains more detailed and meaningful information than the individual words that make it up ("argentina," "won," "fifa," "2022"), which may be in any random order.
- In tweet segmentation, three main components are typically emphasized: the tweet's text, name mentions (references to specific individuals), and hashtags. Analyzing these components makes it possible to identify trends or events on social media platforms like Twitter. For example, if there is a sharp gain in tweets containing the multi-gram segment "argentina won fifa 2022," it may indicate that Argentina has won a significant soccer tournament (FIFA 2022).
- To extract meaningful segments from the tweet text, it is common only to keep those segments present as a Wikipedia page title. This ensures that only named entities (such as "Narendra Modi") or significant components (such as "FIFA World Cup") are kept, and unnecessary words that would increase the noise in the event detection process are removed.
- Name mentions refer to specific individuals made using their username (e.g., "@jack07" for Jay Patel). In the tweet segmentation process, it is common to replace the username with the actual name of the person and consider it as a segment. This helps to identify the people being mentioned in the tweet.
- Hashtags are an essential component in event detection models because they contain much information in a concise form and are often used to group related tweets. Hashtags are created by adding the "#" symbol before a word or phrase (without any spaces).
- In event detection models, hashtags are often given more weight because they contain much information and are commonly used to indicate the topic or theme of a tweet. The H value can be increased to increase the weight of hashtags in the event detection process. For example, an H value of 2 means that all hashtags are duplicated in the segmentation process, resulting in a twice weight that would allow hashtags to become more bursty in the next stage. This also ensures that if a segment is not previously seen in the Wikipedia page titles, its use in a hashtag would still make the segment bursty.
- For example, if the hashtag "#FIFAWorldCup" is used in a tweet, it would be replaced by the "fifa world cup" segment in the event detection process.

Bursty Segment Extraction:

- It is computationally costly to cluster all unique segments within a day for event detection, as there may be hundreds of thousands of them. Therefore, it is common to only consider abnormally bursty segments (segments that show an unusual increase in activity) as potential events in the event detection process.
- These bursty segments are identified based on factors such as the probability distribution of the segments, the number of retweets, the diversity of users, and the users' popularity. Focusing on abnormally bursty segments makes it possible to identify potential events more efficiently and with fewer computational resources.
- Once the bursty segments have been identified, they can be further analyzed and clustered together to identify trends or events on social media platforms like Twitter.
- In tweet segmentation and event detection, several factors can be considered to improve the accuracy of the process. One of these factors is user diversity, which can be incorporated using a measure called user frequency. This measure denotes the number of distinct users using a particular segment (s) in a given time window (t).
- Now, several retweets a tweet receives are the next factor. A retweet is a copy of a tweet created by another user, and by analyzing the number of retweets a tweet receives, it is possible to identify tweets related to important events and give more weight to these tweets in the event detection process.
- This can be done by defining a segment retweet count, the sum of the retweet counts of all tweets containing a particular segment (s) in the time window (t).
- In addition to the number of retweets, the popularity of the user who posted a tweet can also be considered. Tweets by users with a large number of followers (such as celebrities or news pages) may be regarded as more critical than those with fewer followers.
- This can be incorporated by defining a segment follower count, which is the sum of the follower count of all users using a particular segment (s) in the time window (t).
- By giving more weight to tweets by popular users, it is possible to filter out spam or self-promoting tweets that might harm the accuracy of the event detection process.
- In event detection, bursty segments are identified based on their bursty weight, which measures their activity or importance in the event detection process.
- So, what we do is give more importance to the retweet count over the followers of the users. To do so, we performed the logarithmic operation on the followers count twice and once on the retweet counts.
- Then, multiplied them to make a score to decide importance. We added one to the current value so that we find the logarithmic of at least one base 10. However, if we do not do this, then we can be stuck with a logarithmic of 0. The following equation represents the formula for the same.

$$Score = A * B * C * D \qquad (1)$$

Where,
$A = \log(retweet\_count + 1)$,
$B = \log(1 + \log(1 + user\_followers\_count))$,
$C = \log(1 + user\_count)$,
D = Probability for segment s in time window t.

- Once the bursty segments have been identified, selecting the top K segments as the most relevant or significant ones is common. The value of K can be chosen based on the desired level of recall (the proportion of relevant events that are correctly identified) and the computational cost of the event detection process.
- If the value of K is small, the recall of events detected may be low, as important events may not be identified. On the other hand, if the value of K is too large, there may be more noise in the event detection process, leading to higher computational costs and potentially lower accuracy.
- Finding a balance between recall and computational cost is essential when selecting the value of K in event detection.

Bursty Segment Clustering:

- The methodology used by the researchers in [11] has been considered in this section. That is, in this process, the bursty segments that have been identified through an unsupervised approach (such as Burst Detection) are grouped into clusters.
- These clusters are then analyzed to determine which ones are related to events and which ones are not. Non-event clusters can be filtered out, leaving only the clusters that are likely to represent events.

- The temporal frequency of a segment refers to how often the segment appears in a stream of tweets over a particular period.
- By considering the temporal frequency of segments, it is possible to identify which segments currently receive much attention and are likely to be related to an event.
- The tweets containing a segment can also be used to calculate the similarity between two segments. Analyzing the words and phrases used in the tweets makes it possible to determine whether two segments are related to the same event or topic.
- By considering both the temporal frequency and the contents of the tweets, it is possible to identify which segments are likely to be related to events and group them into clusters for further analysis.
- The process uses the k-Nearest Neighbors (k-NN) method to determine whether two segments are related. This method identifies the k segments that are most similar to a given segment based on a similarity measure such as temporal frequency or the contents of the tweets that contain the segment.
- If a segment is one of the k-nearest neighbours of another segment, and vice versa, an edge is added between the two segments.
- After all possible edges have been added, the graph's connected components are considered candidate event clusters. Segments with no edges are dumped from additional processing, as they are not regarded as related to any events.
- After clustering the bursty segments, it may be observed that some clusters are not related to any events. For example, a cluster might contain segments that are related to a specific day of the week, such as "Boring Monday." While these segments might be bursty (showing an unusual increase in activity) on specific days, they are not likely to represent events in the traditional sense.
- To eliminate these types of clusters and focus on events that are truly noteworthy, external knowledge bases like Wikipedia can be used.
- By comparing the segments in a cluster to the information in these knowledge bases, it is possible to determine whether the cluster represents a real event or not. This can help to improve the accuracy of the event detection process and eliminate false positives.
- News events are typically considered to be more noteworthy or significant than other types of events. Therefore, one way to filter out non-event clusters and focus on real events is to consider the newsworthiness of the candidate events.
- In this process, the newsworthiness of each candidate event is compared to the maximum newsworthiness of any event in the current time window.
- If the ratio of the maximum newsworthiness to the current event's newsworthiness is less than a particular threshold, the event is considered to be realistic and is kept for further processing. Otherwise, it is discarded as unlikely to be a real event.

How Clustering is done?

- We have used the Machine Learning algorithm called K-Nearest Neighbour (KNN) for event clustering.
- The k-nearest cluster method groups related data points or segments into clusters. It is based on the idea of k-nearest neighbours, which is a method for identifying the k data points that are most similar to a given data point.
- To use the k-nearest cluster method, the data points or segments are first represented as nodes in a graph. An edge is then added between two nodes if they are considered to be related, based on a similarity measure. The similarity measure can be based on various characteristics of the data points, such as their content, temporal frequency, or other features.
- To determine the k-nearest neighbours of a given data point, the method calculates the similarity between the data point and all other data points in the dataset. The k data points with the highest similarity scores are considered to be the k-nearest neighbours of the given data point.
- Once the k-nearest neighbours of each data point have been identified, the method can be used to group the data points into clusters. To do this, the data points are connected to their k-nearest neighbours, creating a graph with multiple connected components. Each connected component is considered to be a cluster, and the data points within a cluster are considered to be related.
- The k-nearest cluster method is a helpful technique for grouping related data points or segments into clusters, and it is often used in applications such as event detection on social media platforms, recommendation systems, and image classification. The main advantage here is that it does not require the data points to be labelled in advance so that it can be used in an unsupervised learning setting. However, the method is sensitive to the value of k, which can affect the number and quality of the clusters formed.

- KNN can also be referred to as a distance-based algorithm because it finds the distance between the neighbouring data and the calculating variable to purify the closest neighbours.
- To quickly understand how it works, figure 3 shows how the five nearest neighbours are found to classify an unknown object as either a 'cat' or a 'dog.'

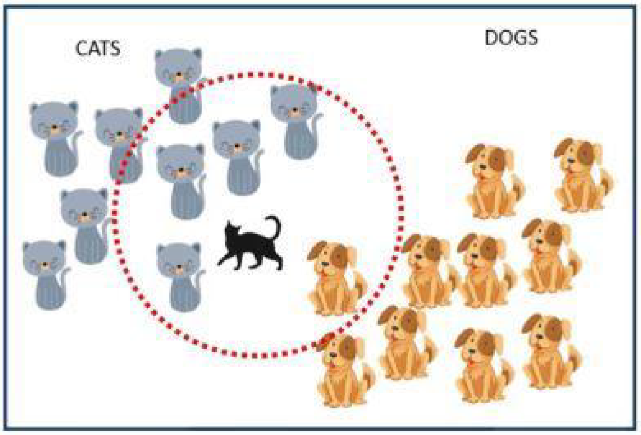


Fig. 3. K-Nearest Neighbour [13]

How did we summarize the Tweets from segment clusters?

- The LexRank algorithm [14] is a natural language processing technique that is used to generate summaries of text documents. It works by ranking the sentences in the document based on their importance or relevance and then selecting a subset of the top-ranking sentences to form the summary.
- In the context of event detection on social media platforms like Twitter, the LexRank algorithm can be used to summarize a cluster of segments that are associated with a particular event.
- To perform this, the algorithm takes as input all of the tweets in a given time window that contains the segments in the event cluster. It then ranks the sentences in these tweets based on their importance or relevance and selects a subset of the top-ranking sentences to form the summary of the event.
- The use of the LexRank algorithm to summarize event clusters can be helpful because it allows for the consolidation of information related to an event into a more concise form. This can make it easier to understand the key points or themes of the event and can also help to filter out extraneous or irrelevant information.
- However, the vital note here is the effectiveness of the LexRank algorithm in generating summaries of event clusters will depend on the quality and relevance of the input tweets.
- If the tweets do not provide a sufficient amount of information about the event or contain a large amount of noise, the summary generated by the algorithm may be incomplete or misleading.
- Additionally, the algorithm may be sensitive to the parameters used to rank the sentences, which can affect the quality of the summary produced.

VI. Evaluation

There are several evaluation metrics for evaluating the model prepared such as Precision, Recall, and many more. But, in our project, we have used human supervision for checking the event detection on a specific day, thus precision is a suitable metric to compare the manually annotated and the predicted events. We manually select the detected event and then summarize them and compare them with the event summarization of the system. The detected events for 12 October 2012 are manually annotated by looking at the segments [4], and we use this to compare our system's output to evaluate how precise is the system's event detection.

As shown in equation 2, The number of real true positives divided by the total of positive predictions is how precision is determined. Where, TP represents True-Positive, TN is True-Negative, FP is False-Positive, and FN is False-Negative for all the equations. If the precision value is 1, then the model predicts all true values.

$$\text{Precision} = (TP) / (TP + FP) \tag{2}$$

VII. Results and Discussions

We get results after each process ends in the system, and to checkpoint these stages, we save the output from each process.

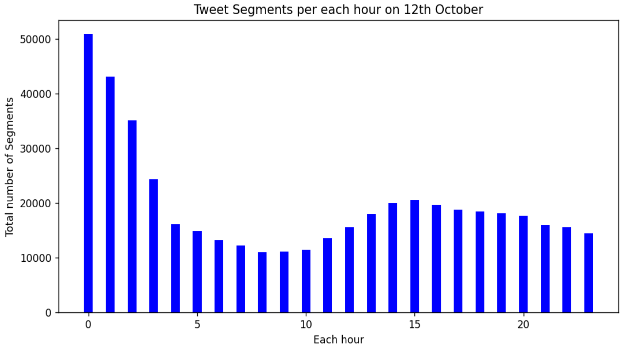


Fig. 4. Tweet segments per hour

The above bar graph illustrates the number of unique segments that are generated at each hour on 12th October 2012. The x-axis represents hours from 12 AM to 11:59 PM. Y-axis on the other hand shows the total number of segments each hour. Overall, it is clearly depicted that the

0th hour had maximum unique tweets of around 50,000. This then reduces continuously because of the uniqueness. This means we are storing the segments in a map that appends only new segments when it comes, else it will just overwrite existing segments. So, the number of unique segments reduces each hour because users would be twitting almost the same content.

The total number of segments we found was 470249 and its busty segments were about 1328.

Table 2 shows the top 10 segments and the similarity of it between all the bursty segments. We got this from all the 24 hours of data on the 12th of October. This is just a snippet of the busty segment similarity. The most similar segment to all the clusters is ‘vp debate’ with a similarity of 17.7020. This indicates there would be some important events related to this segment.

Table 2. Similarity among bursty segments

Segment	**Similarity**
vp debate	17.7020
joe biden	15.3149
vpdebate	15.1605
reasons why we dont get along	14.7204
v pdebate	14.4745
paul ryan	14.2069
debates	14.0192
wan na	13.7770
debate	13.7561
dont know	13.2206

The segment cluster obtained, for example, is: ['middle class', 'last thing', 'middle east', 'tax cuts', 'another war', 'million billion', 'bush tax cuts', 'tax cut', 'ground war',’ taxes’] with newsworthiness of 0.5797. Through this segment cluster, we found the related tweet “liberal allies already raised taxes middle class.” and “middle class pay less rich pay slightly taxes biden.”. Because this tweet contains segments ‘middle class’ and ‘taxes’ which we are looking for. This can be compared with the table 2 results. Wherein, ‘joe biden’ segment had a similarity of 15.3149. The clusters here are formed using 4-nearest neighbours.

Table 3. Segment clusters and tweets fetched

Tweet	**Segment clusters**
liberal allies already raised taxes middle class	['middle class', 'last thing', 'middle east', 'tax cuts', 'another war', 'million billion', 'bush tax cuts', 'tax cut', 'ground war']
middle class pay less rich pay slightly taxes biden	['middle class', 'last thing', 'middle east', 'tax cuts', 'another war', 'million billion', 'bush tax cuts', 'tax cut', 'ground war']

We collect all the tweets containing one or many segments which belong to the same cluster are combined into a file. This file is then passed to the LexRank method. The parameter is the string format of the file and the threshold is ‘none’. This later summarises the events from the input segment clusters. The below table shows the 6 events detected by our system.

Table 4. Detected events

Events detected	**Overlap with the manually annotated event.**
'im middle class dont want biden near'	Related to Bush Tax Cuts and Middle Class Tax Relief and Job Creation Act of 2012
'least biden ryan pro life'	Paul Ryan spoke for 40 of the 90 minutes during Thursday night's vice presidential debate and managed to tell at least 24 myths during that time
'behind vampire diaries'	The Vampire Diaries Season 04 aired this night
'another th inning game'	-
'lol rt ill get back'	-
'bus driver said'	-

Out of the 6 detected events three events overlap with the manually annotated ones. So considering the clustering of the system as error-free, we can say the precision of the system is 50%.

VIII. Conclusion and Future Work

Every social media platform generates a lot of useful information that can be used for event detection. Event detection is an important problem to pinpoint: emergencies, current disasters, or any major global events. However, considering Twitter a lot of data it generates can be noisy and may include informal sentences or unuseful sentences. So, in our work, we have implemented a segment extractor that tried to avoid noise in tweets and then clustered those segments together to detect the events.

We use LexRank to summarise the events, but an extension of that algorithm can be implemented to get a better summary of the segment clusters.

In future, this work can be extended by using an Online Model where events will be detected dynamically. So, this system will continuously extract segments and cluster them together according to the tweets posted by the users. This can help in immediate event detection.

In addition to this, we can have simple machine-learning models that are recurrent in a particular month of the year and increase their probability accordingly. This will help in effective event detection for each month.

We have used a list of Wiki titles dumped in the year 2018. So, a newer dataset with unstemmed phrases that represent the current global events would be more helpful to detect the events.

Due to computational limitations, we have used tweets with less than 3 hashtags and 2 user mentions. But if more computational power is feasible, then tweets with more attributes can be taken into consideration.

References

[1] Hamed Abdelhaq, Christian Sengstock, and Michael Gertz. 2013. Eventweet: Online localized event detection from twitter. Proc. VLDB Endow., 6(12):1326–1329.

[2] Alexander Boettcher and Dongman Lee. 2012. Eventradar: A real-time local event detection scheme using twitter stream. In 2012 IEEE International Conference on Green Computing and Communications, pages 358–367.

[3] Amosse Edouard, Elena Cabrio, Sara Tonelli, and Nhan Le Thanh. 2017. Graph-based event extraction from twitter. In RANLP.

[4] Keval Morabia, Neti Lalita Bhanu Murthy, Aruna Malapati, Surender Samant. 2019. SEDTWik: Segmentation-based Event Detection from Tweets Using Wikipedia. In the Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop

[5] Hossny, A. and Mitchell, L. "Event Detection in Twitter: A Keyword Volume Approach". In Social and Information Networks in 2019. https://doi.org/10.48550/arXiv.1901.00570

[6] Hettiarachchi, H., Adedoyin-Olowe, M., Bhogal, J. et al. Embed2Detect: temporally clustered embedded words for event detection in social media. Mach Learn 111, 49–87 (2022). https://doi.org/10.1007/s10994-021-05988-7

[7] Chenliang Li, Aixin Sun, and Anwitaman Datta. 2012a. Twevent: Segment-based event detection from tweets. In Proceedings of the 21st ACM International Conference on Information and Knowledge Management, CIKM '12, pages 155–164, New York, NY, USA. ACM

[8] Andrew J. McMinn, Yashar Moshfeghi, and Joemon M. Jose. 2013. Building a large-scale corpus for evaluating event detection on twitter. In Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management, CIKM '13, pages 409–418, New York, NY, USA. ACM.

[9] Chenliang Li, Jianshu Weng, Qi He, Yuxia Yao, Anwitaman Datta, Aixin Sun, and Bu-Sung Lee. 2012b. Twiner: Named entity recognition in targeted twitter stream. In Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '12, pages 721–730, New York, NY, USA. ACM.

[10] Chenliang Li, Aixin Sun, Jianshu Weng, and Qi He. 2015. Tweet segmentation and its application to named entity recognition. IEEE Transactions on Knowledge and Data Engineering, 27:558–570.

[11] Chenliang Li, Aixin Sun, and Anwitaman Datta. 2012a. Twevent: Segment-based event detection from tweets. In Proceedings of the 21st ACM International Conference on Information and Knowledge Management, CIKM '12, pages 155–164, New York, NY, USA. ACM.

[12] Raymond A. Jarvis and Edward A. Patrick. 1973. Clustering using a similarity measure based on shared near neighbors. IEEE Transactions on Computers, C-22(11):1025–1034.

[13] Support Vector Machine Algorithm: https://www.geeksforgeeks.org/support-vector-machine-algorithm/

[14] G¨unes Erkan and Dragomir R. Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. J. Artif. Int. Res., 22(1):457–479.

[15] Santos, L. D., & Gómez-Zamalloa, M. (2021). Topic detection and sentiment analysis in Twitter content related to COVID-19 from Brazil and the USA. Journal of Ambient Intelligence and Humanized Computing, 12(5), 1423-1436.