EECS 6895 Final Report:
A11. Automatic Storytelling on Public Event

Xinluan Tian (xt2233)
Jiayao Wang (jw3514)
Cognitive Machine
03/20/2020

Abstract:

Social media is a new platform where people can learn about the news. However, information on social media can be overwhelmed and fragmented. It's very important to have a platform that can aggregate all information on one public event and automatically generate stories that describe the event in human style. Here we use the online streaming of Twitter on a particular public event and use NLP techniques to extract key information about the event. Finally, human-write style stories are generated by extracting key information from tweets.

Background Introduction:

In the past, information was collected, curated by journalists and published. What we learned from journals or newspapers are filtered, well-written, sometimes even biased stories on certain topics. With the rise of social media, the web has become a vibrant and lively Social Media realm in which billions of individuals all around the globe interact, share, post, and conduct numerous daily activities. Collecting information on social media gives us a chance to get a huge amount of raw, on time, and comprehensive information. The combination of social media and big data has created a new area of research, namely social media mining, which is similar to data mining but limited to the worlds of Twitter, Facebook, Instagram, etc. Social media mining is "the process of representing, analyzing, and extracting operational patterns from social media data."[1] In simple terms, social media mining occurs when a company or organization collects data about social media users and analyzes it for rendering.

Automatic storytelling is a process that involves using artificial intelligence (AI) to create written stories. Given a topic and a storyline, machines should generate a story in the human written style which is easy to read by humans. Although automatic storytelling is still far away from building truly creative and insightful novels, it has been steadily improving while working on basic applications. Automated storytelling tools combine AI, machine learning, and big data to create content. In general, automatic storytelling can be used to write headlines, financial reports, and weather updates, as well as anything from screenwriters or short stories.Today, the practical use of automated storytelling is to use the process to "write" more technical headlines or reports and allow human writers to focus their time on more creative stories that may be less structured. Automated narratives begin by collecting large amounts of data into a

database. This data may include information such as hundreds or thousands of different stories or titles. Tools such as Natural Language Processing (NLP) will then scan the data and parse it into structured data. Templates are created by humans, so AI can replace the information in the template with its own template. Templates can be lower-level, meaning they can be simple points that replace data values with AI (e.g., avid reports), or higher-level ones that are intended for more complex and meaningful writing templates. Natural language generation (NLG) is used to automatically generate text-based summaries from a database.

Related Work:

There are some successful automatic storytelling algorithms. Yao et al present a plan-and-write hierarchical generation framework that first plans a storyline, and then generates a story based on the storyline[2]. Ammanabrolu et al propose a Neural network-based approach to automated story plot generation attempts to learn how to generate novel plots from a corpus of natural language plot summaries[3]. Guan et al propose to utilize commonsense knowledge from external knowledge bases to generate reasonable stories to avoid repetition, logic conflicts, and lack of long-range coherence in generated stories[4].

Another way to generate story is extract key information from a long article and produce a short story, which is more relevant to our purpose. Thise types of tasks are called "Text summarization". Text summarization methods can be divided into two main categories, extractive methods and abstractive methods. Extractive Summarization: extract the original content as a summary through keywords, location, and other characteristics. [5] These methods rely on extracting several parts, such as phrases and sentences, from a piece of text and stack them together to create a summary. Therefore, identifying the right sentences for summarization is of utmost importance in an extractive method. The most common extractive method is text rank, influenced by PageRank algorithm, these methods represent documents as a connected graph, where sentences form the vertices and edges between the sentences indicate how similar the two sentences are. The similarity of two sentences is measured with the help of cosine similarity with TFIDF weights for words and if it is greater than a certain threshold, these sentences are connected. This graph representation results in two outcomes: the sub-graphs included in the graph create topics covered in the documents, and the important sentences are identified. Sentences that are connected to many other sentences in a sub-graph are likely to be the center of the graph and will be included in the summary Since this method does not need language-specific linguistic processing, it can be applied to various languages. At the same time, such measuring only of the formal side of the sentence structure without the syntactic and semantic information limits the application of the method. Abstractive Summarization methods use advanced NLP techniques to generate an entirely new summary [6]. Some parts of this summary may not even appear in the original text. Abstractive summarization models learn a large amount of data through deep learning models to encode

and decode to generate abstract content. The source of abstract content is not limited to the original content.

Data Collection:

In our project, we perform social media data mining on Twitter data and generate stories from key information summarized from social media data. Spark streaming together with Twitter developer API [7] are used to retrieve tweets on specific topics.For method development purpose, we use the twitter news dataset [8]. The dataset contains 5234 news events from Twitter, as well as the tweets talking about those news events. All tweets were provided with twitter id and associated events. We use twitter developer API and tweepy package [9] to obtain the tweets on certain events we specified. We analyzed several events including Yosemite wildfire; Syria chemical weapons; the death of poet Maya Angelou and the 2014 World Cup.

Methods and Experiments:

1. Data preprocessing and text cleaning

We did some NLP related work to preprocess the obtained dataset. Here we used regular expressions to finish data cleaning. The specific procedures of data cleanings are tokenization, lowercases, removing URLs, removing punctuations, lemmatization, and normalization. To be more specific, when one user retweets, it will show 'RT' characters at the beginning of the tweets and URLs linked to the original tweets. Thus, we need to remove those useless items when we are dealing with reposted tweets.

In order to remove spam, that is unrelated tweets or tweets don't contain much useful information, we first remove all retweeted tweets and set a cutoff on the number of followers of the user who sends the tweets. We hypothesize that if a user has more number of followers, the user's tweets should contain more information and are less likely to be spam. Figure 1 shows the distribution of the number of followers of users on each tweet in Yosemite wildfire dataset. In the next steps, we only perform analysis on tweets sent by user with more than 10,000 followers.

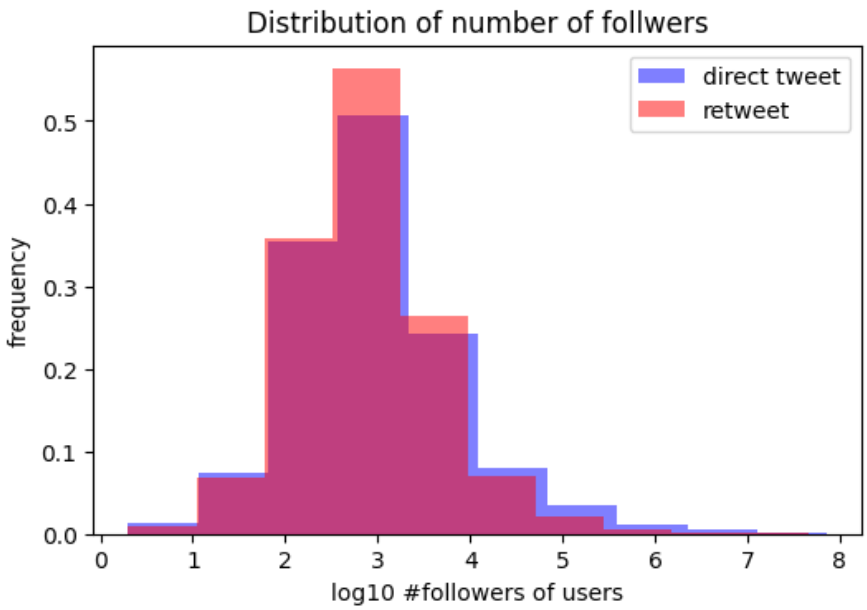

Figure 1. Distribution of the number of followers of tweets (Yosemite wildfire dataset)

2. Word embeddings

A log-bilinear regression model in GloVel that combines the advantages of the two major model global matrix factorization and local context window methods. A word co-occurrence matrix is constructed based on the corpus, and then a word vector is learned based on the co-occurrence matrix and the GloVe model. We use GloVe to get vectorized representations of words that contain semantic and syntactic information as much as possible.

3. Clustering

After mapping words to embeddings, we use PCA to perform dimension reduction on each word vector representation. By using PCA we hope to obtain a representation of each tweet in a lower dimension and from that, clustering the tweets to obtain the key information. With lower dimensions also can accelerate steps afterwards. Figure 2 shows the PCA results in word vectors. Figure 3 shows the variance explained by top N PCs. As we can see, in different datasets, we need different numbers of PCs to achieve the same level of variance explained. The world cup dataset has more tweets and more words, the original dimension is much higher than Yosemite dataset. We have to use top 80 PCs to achieve 70% variance explained. In Yosemite dataset, top 25 PCs can achieve 70% variance explained. In different datasets, we always pick a top N PC that can achieve 70% variance explained for hierarchical clustering.

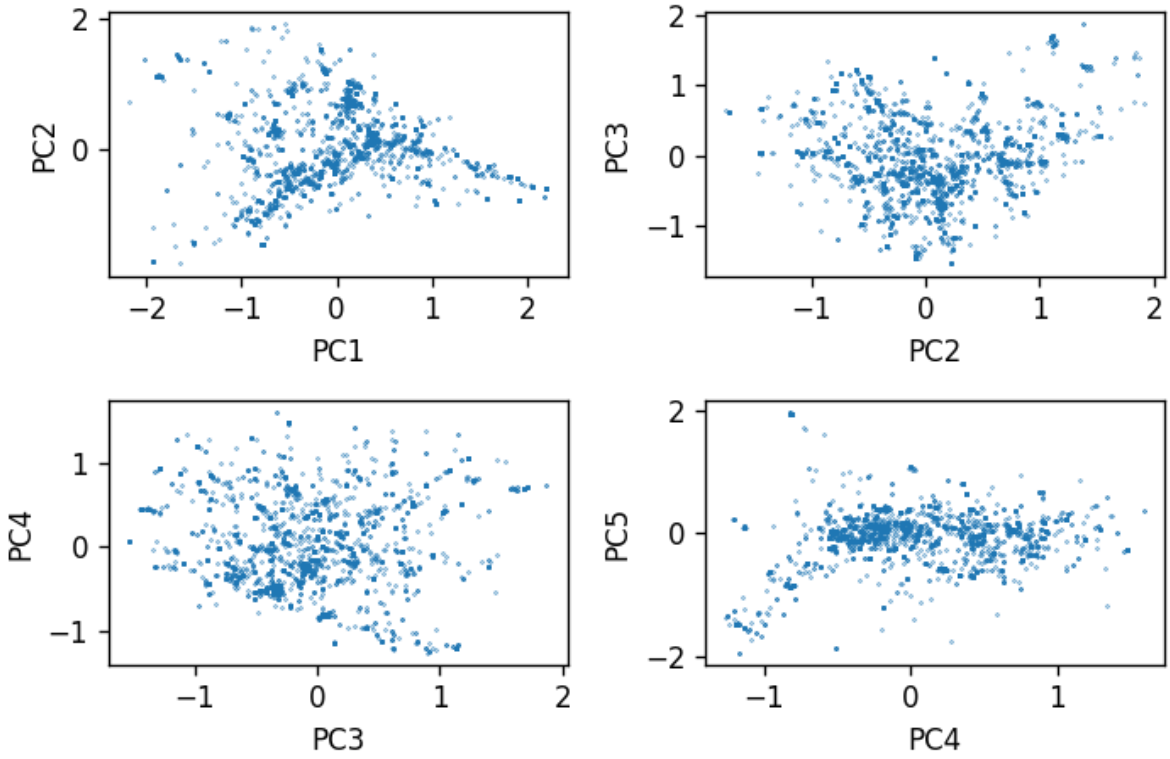

Figure 2. PCA of bag of words (Yosemite wildfire dataset)

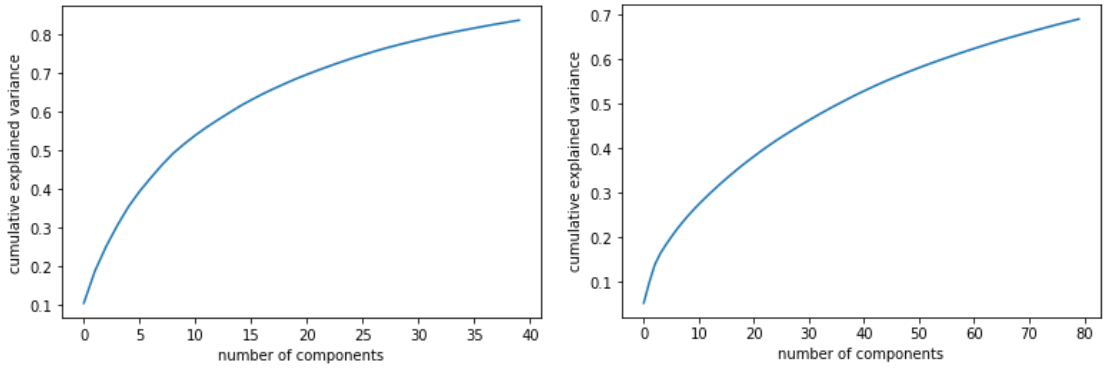

Figure 3. Variance explained vs number of components used. Left is Yosemite Wildfire data and right is 2014 world cup data.

After PCA, we use hierarchical clustering for clustering the tweets represented by top N PCs and identify key information. By doing this, we hope tweets that represent different sub-topics of the event could be clustered together. As shown in Figure 4, In the Yosemite dataset, we set 8 clusters, each of the clusters consisting of several hundred tweets talking about the same topic. In each cluster, we pick the tweets in the middle of the cluster and display the content. We can see each cluster does represent different information about the

event, like tourists fleeing, the damage made, and how fast the fire spread. Furthermore, we sort the key tweets by the time they were sent to form a timeline of the event (Figure 5).

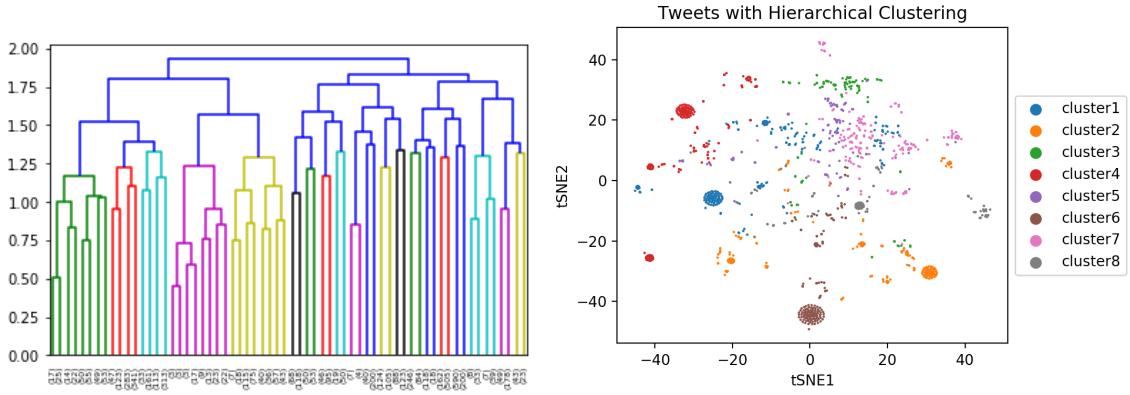

Figure 4. Dendrogram and tSNE plot of tweets of hierarchical clustering on tweets of Yosemite wildfire forms key information

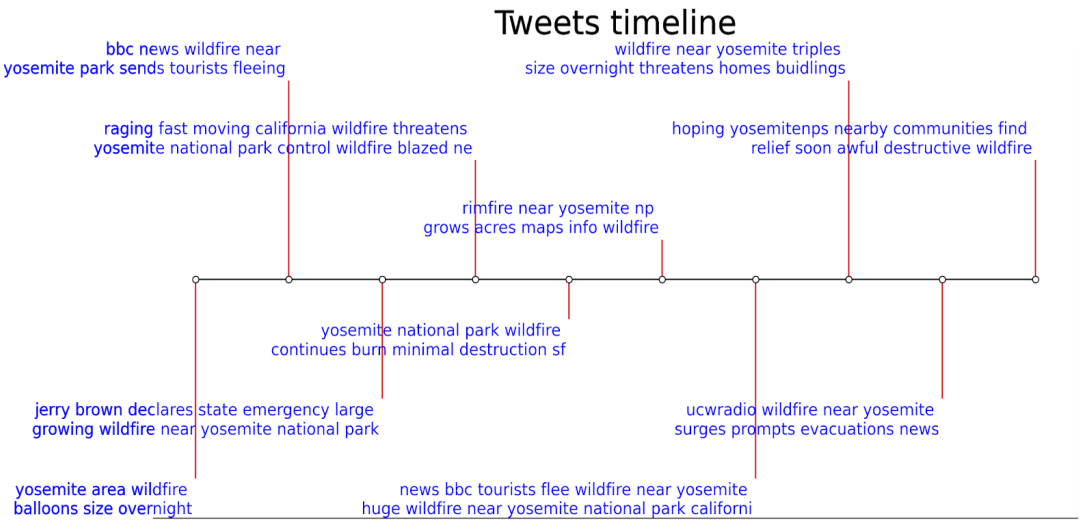

Figure 5. Event timeline represented by key information(Yosemite wildfire dataset)

4. Form of pseudo-article

When running extractive summarization methods, we need a pseudo-article to make it work. For each cluster obtained in the previous step, we calculate cosine similarity between all tweets belonging to the same cluster, remove highly similar (>0.8) tweets and only keep one for they should contain the same information. We also remove tweets if one has low similarity (<0.2) with all other tweets, which means it could talk about a different thing or not as credible as other tweets. Then we sort tweets by their time and get tweeted, so they will follow a timeline of happening. Also, all tweets go through a language check module to make sure they are grammar correct. Then we concatenate all processed tweets together to form a pseudo article, which also can be the longer version of our story.

5. Text summarization

After we got the pseudo article, we ran a bert-extractive summarizer on it to obtain the final story. Bidirectional Encoder Representations from Transformers (BERT) is a technique for NLP pre-training developed by Google. As opposed to directional models, which read the text input sequentially, the Transformer encoder reads the entire sequence of words at once. Therefore, it is considered bidirectional. This characteristic allows the model to learn the context of a word based on all of its surroundings (left and right of the word). Bert-extractive-summarizer utilizes the BERT model for text embeddings and K-Means clustering to identify sentences closest to the centroid for summary selection. BERT builds on top of the transformer architecture, but its objectives are specific for pre-training. On one step, it randomly masks out 10% to 15% of the words in the training data, attempting to predict the masked words, and the other step takes in an input sentence and a candidate sentence, predicting whether the candidate sentence properly follows the input sentence.

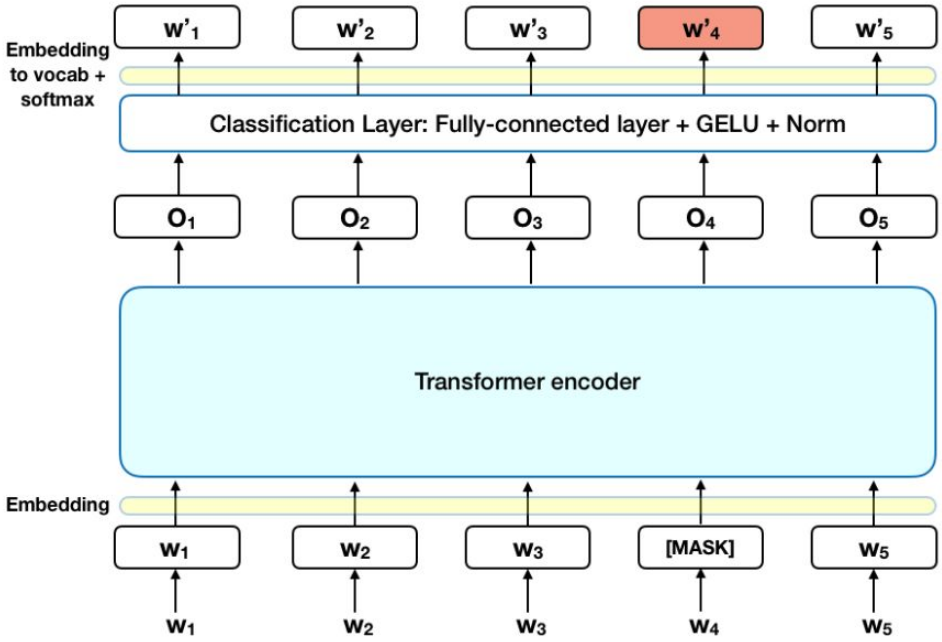

Fig 6. BERT Extractive Summarizer

Using the default pre-trained BERT model, one can select multiple layers for embeddings. Using the [cls] layer of BERT produces the necessary N x E matrix for clustering, where N is the number of sentences and E is the embeddings dimension. Once the embeddings were completed, the N x E matrix was ready for clustering. From the clusters, the sentences closest to the centroids were selected for the final summary.

System Overview:

Our model is implemented with python, NLP related work such as data cleaning was done by NLTK library [10]. Unsupervised clustering was done with sklearn [11]. To run the program, you

could run storytelling.py -i event.csv, event.csv contains the twitter and associated metadata, output will be a long and short version of the story. To download twitter dataset you can use storytelling.py --download eventid to get twitter data associated with an event. For data downloading you need to obtain twitter developer API credentials. Jupyter notebooks were also available to explore the functions.

Evaluation:

Here are the results on 5 different public events.

1. Yosemite wildfire: Wildfire near Yosemite grows to 25 square miles. Beyond that, Yosemite wildfire triples in size. Furthermore, " Huge wildfire spreads through Yosemite. Jerry brown declares state of emergency for large growing wildfire near Yosemite national park wildfire threatens homes and camps near Yosemite. yosemite-area wildfire balloons in size overnight. Abq yosemite-area wildfire triples in size overnight. In fact, Rimfireca fire fighters lose ground against growing wildfire camp closing info hard.
2. Billboards2014: Lorde s feathered frock an damp worst dressed at billboard awards. Brad paisley + Keisha are the cocaine and waffles of billboard awards bias. Moreover, Kendall + Kyle kinda matched at the bias. In addition, Looked fierce in Alexander McQueen at the 2014 bias. And Sorry Christmas used all the tinsel at the bias. Moreover, Just say no to hologram Michael Jackson. Michael Jackson hologram moonwalks at billboard awards mjxscape.
3. World Cup 2014: A Dutch soccer fan drove nearly 13K miles from SF to Brazil in his '55 Chevy. Rio and So Paul shine bright. Also, Eating and drinking in So Paul during the WorldCup2014. In fact, Louis van Goal's new broom has Holland flying under radar in Brazil. And, The World Cup opening ceremony is underway in So Paul Brazil. Psychic Turtle Predicts World Cup Win For Brazil. In addition, 5% of WorldCup players make their professional careers in Europe. Besides, Brazil has spent billions on the World Cup but only about five hundred quid on the opening ceremony. In addition, Brazil 60% possession Croatia 40% the ref 100%. In fact, Pope Francis sends message to World Cup opening. Moreover, Footballers Lionel Mess Christian Ronald. In fact, Brazil World Cup mascot carved on watermelons by Chinese. Nike debuts 3D-printed duffer bag for the World Cup. In addition, Brazil has had 7 years and spent over £7bn to plan for World Cup 2014. Rio de Canard airport workers declare strike on eve of World Cup. In addition, Huge security presence on streets of So Paul police everywhere you look. Beyond that, Baghdad inhabitants fear the enemy is at the gate. England players pose for series with lad from violent fa vela who blagged into hotel. Moreover, Authorities in Brazil are taking unprecedented steps to avoid drought-related water shortages during the World Cup.
4. Death of Maya Angelou: We recall the beautiful farewell words said about data media by Maya Angelou. No sun outlasts its sunset but will rise again and bring the dawn. And, Harlem remembers Maya Angelou as literary icon neighbor. Besides, What Maya Angelou meant to winston-salem. Maya Angelou is remembered in flint for her visits to the whiting and library. Also, Transportation advocates hail Maya Angelou as streetcar pioneer. Beyond that, Poet Maya Angelou shown speaking to a sold-out Jesse hall crowd has died at 86. And, Maya Angelou reading at the 1993 presidential inauguration ceremony. We may encounter many defeats but we

> must not be defeated. In addition, Photos: maya angelou on screen: 8 powerful TV and film appearances. In fact, Poet maya angelou on leadership politics and race. Moreover, Maya Angelou's legacy will live on in the efforts of all those who fight for freedom, dignity and humanity. In addition, Beyonce mourns maya angelou with this heartfelt tribute!.

For evaluation of the results, since there is no good metric for such tasks, we performed a self-evaluation by reading those stories. We evaluate our model based on comprehension, logic, cleanness, sentiment and credibility. We score 4 / 5 for comprehension, since the cluster algorithm works fine and gives a pretty comprehensive summary of the event, including many different subtopics and aspects. We score 2 / 5 for logic, Since we didn't imply a relationship between tweets/sentences, which itself is a very difficult task, for even if the storyline can't be well represented by time the tweets get tweeted, the summary doesn't get a good logic. For the cleanliness of the story, we score 2 / 5. Twitter data are pretty messy and have many irregular usages. Stories from tweets based on extractive methods can't solve this issue since they directly use the tweets to form the story. Our language check module works but not very well. The Sentiment of the story we score 4 / 5 since they convey the sentiment the original tweets expressed. For credibility of the story, we score 4 / 5 since we set many filters to make sure the tweets we have in the final story are pretty credible, like filters on the number of followers of the user who sent the tweet and tweets they mention the same thing based on similarity.

Conclusion and Discussion:

We implemented a lightweight version of an automatic storytelling system. It gets twitter data on a public event and tells a story in an extractive way. The biggest advantage is it's fast and doesn't involve any training. It utilizes pre-trained NLP embeddings such as GLOVE twitter model and BERT to represent semantic and syntactic meaning of sentences and use unsupervised clustering to get key information from tens of thousands of tweets. However, it also has many challenges and limitations. The biggest challenge comes from the nature of the data: Twitter data are short and messy, which makes them hard to analyze. We use unsupervised learning to aggregate and summarize them but performance is still not good enough. It's hard to connect tweets in a logical way except we can concatenate them by date, but the date of the tweets doesn't represent their underlying timeline in many cases. This limitation also limits the usage of extractive methods. The extractive methods usually work with well-structured articles and its results rely on logic within the original article. If the original article doesn't structured well, the result in summary is also not good. Also extractive methods can't solve the problem of the tweets convey the meaning but in bad grammar since they only use the original sentence and don't have the ability to rewrite the sentence.

Abstractive methods perform better in ordinary text summarization methods, they use advanced recurrent neural networks such as seq2seq models to learn the embedding between long article and short summary and are able to produce sentences that are not in original

articles. However these models need extensive training data and the training data must be able to represent the data we use here. However, existing training data such as CNN news and daily mails are not the same as twitter data we are facing. Models trained on CNN news dataset work badly on twitter data since CNN news are well-structured articles and their topics are very narrow. However, if there can be a good training dataset, say a large number of public event news articles and associated tweets, we could train a good seq 2 seq model for story telling. Gathering such a big dataset itself is a huge project and beyond our scope.

Reference:

[1] R Zafarani, MA Abbasi, H Liu. Social media mining: an introduction.

[2] Lili Yao, Nanyun Peng, Ralph Weischedel, Kevin Knight, Dongyan Zhao, Rui Yan. Plan-And-Write: Towards Better Automatic Storytelling. (2019)

[3] Ammanabrolu, Prithviraj, Ethan Tien, Wesley Cheung, Zhaochen Luo, William Ma, Lara J. Martin and Mark O. Riedl. "Guided Neural Language Generation for Automated Storytelling." (2019).

[4] Guan, Jian, Fei Huang, Zhihao Zhao, Xiaoyan Zhu and Minlie Huang. "A Knowledge-Enhanced Pretraining Model for Commonsense Story Generation." ArXiv abs/2001.05139 (2020): n. pag.

[5] Mehdi Allahyari, Seyedamin Pouriyeh, Mehdi Assef et. al. "Text Summarization Techniques: A Brief Survey". arXiv:1707.02268v3 (2017)

[6] Ramesh Nallapati, Bowen Zhou, Cicero Nogueira dos santos, Caglar Gulcehre, Bing Xiang. "Abstractive Text Summarization Using Sequence-to-Sequence RNNs and Beyond". The SIGNLL Conference on Computational Natural Language Learning (CoNLL), (2016)

[7] https://www.tweepy.org/

[8] Quezada, Mauricio; jkalyana@ucsd.edu; bpoblete@dcc.uchile.cl; gert@ece.ucsd.edu (2016): Twitter News Dataset. figshare. Dataset. https://doi.org/10.6084/m9.figshare.3465974.v2

[9] https://developer.twitter.com/en

[10] https://www.nltk.org/

[11] https://scikit-learn.org/stable/