An Unsupervised Approach for Summarizing the Plot of Twitter Trends

Jaya P A[1 2], Srinath Srinivasa[1], and Dipti Deodhare[2]

[1] International Institute of Information Technology,
Electronic City, Bangalore, Karnataka
jaya.p.a@iiitb.org, sri@iiitb.ac.in
[2] Center for Artificial Intelligence and Robotics,
C V Raman Nagar, Bangalore, Karnataka
dipti.deodhare@cair.drdo.in

Abstract. Trending topics on social media sites depict large amount of user interest around some event or issue. User activities and opinions often mask the primary "plot" behind the trend. In this paper, we investigate the problem of summarizing the plots of Twitter trends. We define a plot as comprising of two primary elements: key entities and their characterization, and key actions. We present a system that takes a collection of tweets as input and generates a plot summary using an unsupervised approach. The entire tweet collection is first represented as a collection of noun phrases representing entities. Then, using Markov Random Field based factor computation techniques, key actions are identified by connecting entities with verbs, to form activity triples. These triples and corresponding scores are used to identify key sentences from tweets and cluster them. The activity triples and corresponding sentence clusters help to focus only on a limited set of sentences rather than going through the entire collection of tweets. The proposed system, is assessed on a collection of tweets related to various trending topics. Experiments show that the proposed approach is effective in characterizing the plot and identifying the key sentences.

Keywords: Tweet summarization, Markov Random Field, Social Media Analytics

1 Introduction

Trending topics on social media sites like Twitter, contain actionable knowledge for several stakeholders like law enforcement agencies, policy-makers and executives. However, trending topic datasets are also noisy with a number of user reactions diluting the main message(s) behind the trend, or its "plot". Automated methods for summarization of the underlying plot is hence an important need.

Automatic tweet summarization has attracted significant research interest [1, 8, 9, 17, 19]. Most existing work in tweet summarization have focused on identifying few key tweets as summary by using various extractive summarization techniques. Salience of tweets are computed using lexical features such as term frequency, term overlapping, term positions and Twitter-specific features such as followers, re-tweet, hyper-links and time-line informations. Inouye and Kalita [10] provides a comprehensive list of extractive summarization algorithms and compares the performance on micro-blog posts. Approaches towards summarizing tweets have taken varied perspectives. These include simple frequency-based techniques, various graph-based representations, and machine learning approaches.

In this paper, we explore a generic, unsupervised approach to identify key actors and relationships between them, that form the plot for a trending topic. Our approach is based on creating an abstract model of entities and interactions between them. A set of activity triples points derived from the model is used to identify key sentences from the tweets and cluster them. The key contribution in this work is the identification of abstract elements like actors (including non-person entities), their characterizations and relationships between actors, as part of a Twitter trend. These elements can be stored and retrieved separately for comparing actions involving actors across different trends.

An actor is defined as a central role-player entity of the story narrated by the trend. Actors could also refer to non-person entities like cities, countries, organizations, etc. that play a central role in the trend. Characterizations for an actor represent the set of qualifier terms that can be attached to an actor for this trend. Relationship between actors are in the form of a predicate that best connects two actors in this trend.

For example, in one of the datasets about a Twitter trend involving a bull-fight festival called "Jallikattu", some of the actors identified were entities like: *sr counsel parasaran*, *karnatakahc*, *kambala verdict*, etc. Some of the characterizations of the actor *sr counsel parasaran* were terms like "defend," "opposes," etc. and some of the characterizations of *karnatakahc* were terms like "approached," "deferred,"etc. Similarly the actors *karnatakahc* and *kambala verdict* were linked by a predicate called "deferred."

The summarization itself involves extracting sentences from the tweets based on the above abstractions. This will not only help the user to focus on a limited set of sentences rather than going through thousands of tweets, but the abstractions of actors, their characterizations and relationships can also be stored for posterity and compared across trends. The proposed technique does not require a domain ontology or training data, and uses only shallow (lexical) NLP techniques, thus making it the generic, first line of attack to address the larger problem of obtaining actionable insights from social media.

2 Related Literature

There are two main approaches for automatic summarization of text: extractive and abstractive methods. Extractive methods select a subset of existing words, phrases, or sentences as a summary from the original text without altering them.

In contrast, abstractive methods build an internal semantic representation and then use natural language generation techniques to create a summary that is closer to what a human might generate.

In the case of short documents like tweets, research on summarization to date has primarily focused on extractive methods. Inouye and Kalita [10] compares various algorithms used for extractive summarization and applying the same on micro-blog posts.

Sharifi et al. [17] proposes two algorithms *phrase reinforcement* and *hybrid tf-idf* for generating single and multi-document summaries. The Phrase Reinforcement (PR) algorithm builds a weighted and directed acyclic graph, and generate summary by searching for the most weighted paths through the graph. The hybrid tf-idf method computes the score for each of the tweets and the ones having highest score are considered for the summary.

Xu et al. [19] uses information extraction techniques to identify events from Twitter trends, and build an event graph from the document collection. A PageRank-like algorithm is used to partition the events from event graph and generate variable length summaries. This approach returns a bag of words and the tweets containing those words are chosen as a summary.

Chang et al. [3] proposes a time-line based summarization framework called Timeline-Sumy, which aims to provide details of an entity from social media by displaying a set of episodes about the entity in a chronological order. For time-line summarization the system first identifies the key time-line episodes and then summarizes each episode. Time-line episodes are detected by explicitly modeling temporal information with life cycle models, since episodes usually exhibit sudden-rise-and-heavy-tail patterns in time series. The episode summarizer uses feature based extraction methods and ranks social media posts in each episode via a learning-to-rank approach. Huang et al. [9] performs participant based sub-event detection and summarization for sports games using online clustering and temporal-context mixture models.

Chakrabarti and Punera [2] explore tweet summarization of highly structured and recurring events, such as sports. They have used Hidden Markov Model to learn the underlying hidden state representation of the events. Dutta et al. [4] construct a tweet similarity graph based on term level similarity and semantic similarity between tweets. Further, community detection algorithms are used to find the tweet nodes that are densely connected. Within a community, the tweet node which represents the cluster centre or higher weighted degree or the longest tweet are considered to form the summary.

Liu et al. [14] proposes a graph-based multi-tweet summarization system based on the popular graph summarization algorithm called LexRank [5], which selects a subset of tweets from the input tweet collection. It leverages social network features such as the number of followers and the number of times re-tweeted, readability and user diversity for selecting representative tweets. He et al. [8] utilize social and temporal aspects of tweets and experimented as part of LexRank and SumBasic [18] summarization systems.

To contrast our work with current literature, the focus on our work is to identify the primary “plot” behind a trending topic. We define a plot as comprising of key actors, their characterizations, and their key actions. The proposed method is a *hybrid* of abstractive and extractive summarization methods. We build an abstract model of key actors and their actions using factor computation over a pair-wise Markov Random Field (MRF) [11]. We connect nodes of the MRF network with actions in a way that the joint probability of the network of interactions, is maximized. The set of factors so computed gives us an abstract model of the plot. However, rather than using sentence generation algorithms to create a purely abstractive summary, we use each edge from the network to extract the relevant sentences from tweets and cluster them. This way, we can establish connections between the abstract plot model and the actual dataset. The abstract model itself, can be used to list out key actors and actions from the plot.

A popular metric for calibrating summarization models is the Recall-Oriented Understudy for Gisting Evaluation (ROUGE) metric [12, 13], which is used to evaluate our approach as well. The ROUGE metric is based on comparing n-grams in the summary generated by the algorithm with a human-generated gold standard summary. To the best of our knowledge, there is no single standard dataset available for summary evaluation of tweets, given the large diversity in the underlying motivation for tweet summarization. Hence in our approach, we also conduct a controlled experiment with human evaluators to create a gold standard against which the model is compared. Four other existing summarization models are used as baseline and compared using ROUGE against the human-generated summaries.

3 The Hybrid Summarization Approach

We start with a collection of tweets T pertaining to a given trending topic. The details of tweets data used for the experiments are explained in Section 4. The first task is to pre-process the tweets to remove special characters, URLs and canonicalize the content. A generic dictionary consisting of common abbreviations and code words is then used for canonicalization – or converting different variations of well-known terms into a base, common form.

The canonicalized text is then processed through a series of NLP pipeline stages to perform Tokenization, Sentence Boundary Detection, Parts Of Speech (POS) tagging and Chunking. The sentences from the tweets are identified based on the standard sentence boundary markers, such as '.','!' and '?'. Then, POS tagging and chunking are performed on the tweet sentences using Twitter NLP toolkit[3]. The Twitter NLP uses IOB format for the tags, where each token is tagged with one of three special chunk tags: I (inside), O (outside), or B (begin). The B and I tags are suffixed with a noun or verb chunk type. Noun phrases are tagged with B_NP and I_NP tags, and verb phrases are tagged with B_VP and

[3] https://github.com/aritter/twitter_nlp

I_VP tags. The following regular expression based rules are used to collect the noun and verb phrases from tagged tweet sentences.

$$
\begin{aligned}
Noun\_Phrase &= ((B\_NP)(I\_NP)^*)\ . \\
Verb\_Phrase &= ((B\_VP)(I\_VP)^*)\ .
\end{aligned}
\qquad (1)
$$

Let N_T and V_T represent the sets of all noun and verb phrases respectively, that were identified from the dataset. The set of noun phrases identified from the dataset is then used to build an abstract model $\mathcal{H}$ that is used to represent the plot. Let $|N_T| = n$ and $m = \frac{n(n-1)}{2}$ denote the number of possible edges between pairs of noun phrases in N_T.

The set of noun phrases thus identified form the collection of candidate "actors" (including non-person actors) of the plot. Verb phrases that include characterization and actions, are attached to either a given actor or between pairs of actors in a way that the joint probability of all assignments is maximized. This is achieved by factorizing the graph as a pair-wise Markov Random Field (MRF) [11].

Given a set of actors and actions N_T and V_T, the terms $Val(N_T)$ and $Val(V_T)$ describe the set of all possible valencies for the elements of the set. In this work, valency is modeled as a binary variable having a value of either 0 or 1 indicating the absence or the presence of the given element.

We define a factor as a function of the form $\phi : Val(D) \rightarrow \Re$ where D is either N_T or V_T, that denotes the amount of evidence for a given valency of an element. In this work, the factor function is modeled by the frequency, or the number of tweets in the dataset that display the said valency. Marginal factor functions for every valency is denoted by specifying the valency as a subscript. Hence, for any $v \in V_T$, the functions $\phi_0(v)$ and $\phi_1(v)$ indicate the number of tweets not mentioning and mentioning v respectively.

Given the set of actors, we need to associate verbs (characteristics or actions) with actors. A verb assignment for verb v_i is indicated by the random variable X_i, where $val(X_i)$ denotes the valency of the nouns that are on the either ends of the verb. Hence, for a very v_i being assigned on an edge (n_1, n_2), its random variable X_i can take on four values $00, 01, 10$ and 11 corresponding to whether the verb v_i was seen in the vicinity of n_1 and n_2. For example, the probability $P[X_i = 01]$ indicates the probability of verb v_i occurring in a tweet that does not contain n_1, but contains n_2.

Any given assignment of verbs to the network is indicated by its joint probability as follows:

$$
P(X_1, \ldots, X_m) = \frac{1}{Z} P'(X_1, \ldots, X_m)\ . \qquad (2)
$$

where

$$
P'(X_1, \ldots, X_m) = -\sum_{i=1}^{m} [\log \phi_{val(X_i)}(v_i) + \log(\rho(v_i))]\ .
$$

and

$$Z = \sum_{X_1,\ldots,X_m} P'(X_1,\ldots,X_m) \ .$$

In the above equation, the term $\rho(v_i)$ represents the "verb prior" or the prior probability of the verb being considered. This is calculated as:

$$\rho(v_i) = \frac{\phi_1(v_i)}{\phi_0(v_i) + \phi_1(v_i)} \ . \tag{3}$$

The verb assignment is said to be optimal when the joint probability of assignment is maximized:

$$\hat{\mathcal{H}} = \arg\max_{X_1,\ldots,X_m} P(X_1,\ldots,X_m) \ . \tag{4}$$

This model is also said to maximize the pairwise factor potential of actors in the network. The set of factors so computed gives us an abstract model of the plot. Verb assignments denoting valencies of the form 01 and 10 denote occurrences of verb phrases in the absence of one of the noun phrases. When such assignments are found in the discovered model, these are considered as characterizations of the actor that is present in the assignment, rather than an action connecting two actors. Figure 1 shows some examples of actors and their characterizations for the trending topic #jallikattu. Assignments whose valencies are of the form 00 are removed from the discovered model.

Actor and characterizations
#jallikattu bill , passed
#jallikattu bill , cleared
#karnataka high court, deferred
#karnataka high court, to remain
#karnataka high court, to await
#jallikattu protesters, demands
#jallikattu protesters, arrest
#jallikattu protesters, diverting
#jallikattu protesters, wanted to stop
#kambala matter , adjourned
#kambala matter , to wait
#kambala verdict, deferred
honorable president, conceded
honorable president, permitted
the president, approved

Fig. 1. Actors and their characterizations for the trending topic #jallikattu

We then take assignments of valency 11 from the discovered model $\hat{\mathcal{H}}$. It represents the activity triples of the form $(noun, verb, noun)$ denoting interactions between actors. We use this to search for one or more sentences from the tweet

Cluster 1
Triple :: feb 9, announced, #jallikattu final dates
#jallikattu final dates announced! #avaniyapuram - feb 5, #palamedu - feb 9, and #alanganallur - feb 10.
#jallikattu final dates announced #avaniyapuram feb 5 #palamedu feb 9 #alanganallur feb 10 get ready.
#jallikattu final dates announced - #avaniyapuram feb 5 #palamedu feb 9 #alanganallur feb 10 @offl_lawrence @hiphoptamizha @rj_balaji.
Cluster 2
Triple :: #kambala, come, petitioner
#kambala matter adjourned for two weeks by #karnatakahc, and advised the petitioner to come back once #sc grants verdict on #jallikattu.
timesofindia: #kambala matter adjourned by karnataka hc for two weeks, petitioner advised to come back once sc grants verdict on #jallikattu.
karnataka hc adjourns #kambala matter, ask petitioner to come back after sc gives #jallikattu verdict.

Fig. 2. Top two clusters and their sentences for the trending topic #jallikattu

dataset that match such a triple. All matching sentences are clustered together to represent the said action.

Figure 2 shows the top two clusters and their extracted sentences for the trending topic #jallikattu. The clusters are ranked in descending order of their total cluster score, which in turn is given by:

$$clusterscore = k * [\log \phi_{val(X_i)}(v_i) + \log \rho(v_i)] \ . \tag{5}$$

where k is the number of sentences matching the triple represented by random variable X_i representing valency 11 for verb v_i.

The set of noun phrases form the actors of interest in the plot. They are in turn ranked in descending order according to an actor score. The actor score is computed based on combining the node prior along with all the action potentials, where the given node participates. Let $n_i \in N_T$ be an actor with node prior:

$$\rho(n_i) = \frac{\phi_1(n_i)}{\phi_0(n_i) + \phi_1(n_i)} \ . \tag{6}$$

Let X_{n_i} be the set of all verb assignments in the model that include n_i and have a valency 11. That is, X_{n_i} is the set of all interactions of n_i with other actors. Let $clusterscore(X_{n_i})$ denote the total cluster score of all clusters in X_{n_i}. The actor score of n_i is then calculated as follows:

$$actorscore(n_i) = clusterscore(X_{n_i}) + \log \rho(n_i) \ . \tag{7}$$

Some of the top actors identified from the #jallikattu dataset are “jallikattu,” “tamilnadu,” “india,” “kambala,” etc. Such actors, their characterizations and the actions form the summary plot model. The proposed technique does not require a domain ontology or training data, and uses only shallow (lexical) NLP techniques, thus making it generic. It can be used as the first line of attack towards obtaining actionable insights from trending topics on social media. In addition, most tweet summarization systems are tightly coupled with Twitter-specific features, as well as features specific to the particular structure of events. In contrast, our approach can be used in any set of social media posts, represented as short documents, where we can identify the key actors, their characterizations and the key actions.

4 Experimental setup

We assessed the proposed summarization method using tweets related to various trending topics, such as *#jallikattu*, *#wannacry*, etc. We collected tweets related to each hash-tag using the streamR[4] library for R. We considered the actual tweet content because want our model as generic enough so that it can be applied to any kind of short and noisy reports. We filtered duplicates and discarded tweets which contained only one or two words once the URL is removed. Finally, we obtained about 500 to 800 unique tweets about each hash-tag for the summarization.

We have considered manual and automatic methods for the summary evaluation. The quality of clusters identified by our approach is judged by a team of human evaluators. The evaluators were picked from the professional colleague network of the authors in their parent organizations. The evaluators were aware of the summarization problem, and also understood the significance of each of the trending topic in the collection.

Initially, the input set of tweets was given to evaluators to carefully read and comprehend the importance of each trending topic. Later, the output clusters were presented to them for evaluation. The evaluators scored each cluster with 0 or 1, based on the importance of the cluster with respect to the input trending topic. In order to measure the inter-rater agreement, we computed Cohen's Kappa score[5] on the cluster evaluations. Figure 3 shows a high overall inter-rater agreement for the trending topic #jallikattu.

#jallikattu
Overall Agreement 0.814815
Fixed Marginal Kappa: 0.629121
Free Marginal Kappa: 0.629630

Fig. 3. Cohen's kappa scores for inter-rater agreement

In the case of automatic summary evaluation, the system generated summary is compared against one or more manual summaries. In order to create a

[4] https://cran.r-project.org/package=streamR

[5] Online tool to measure inter-rater agreement: http://justusrandolph.net/kappa/

gold-standard summary for comparison, we formed a team of human evaluators. We selected the above mentioned trending topics, with the intention that the evaluators would be aware of these incidents and they will be able to produce a better summary. The evaluators were apprised of the problem and were provided the input set of tweets about each trending topic, and were asked to create 10 sentences each, for summarizing every trending topic. The evaluators were free to use their own words in creating the summary and need not extract actual tweet content.

For comparing a given summary against the gold standard, we used the well known automatic evaluation metric ROUGE [12]. A ROUGE-n metric counts common n-grams (sequence of n consecutive words) between the human generated summary and the gold standard. We have considered two human generated summaries of each trending topic for the automatic summary evaluation.

We compared our tweet summarization results with four baselines from existing literature. These comprises of three extractive summarization methods: SumBasic, Centroid and LexRank, and one abstractive summarization method called Opinosis. SumBasic[18], Centroid[16] and LexRank[5] are the well established generic text summarization systems, and being used as baselines across various summarization systems. Our model establishes associations between key entities and key actions. The sentences identified by our method captures these associations, and we wanted to check it against sentences generated by an abstractive method. We have considered Opinosis system[6, 7] which generates summary from highly redundant and short user reviews.

SumBasic is a generic system for multi-document summarization. It is motivated by the observation that words occurring frequently in the document cluster occur with higher probability in the human summaries than words occurring less frequently. SumBasic uses word probabilities with an update function to compute the best k posts.

Centroid is a well-known method for judging sentence centrality. The centrality of a sentence is defined in-terms of centrality of words that it contains. A common way of assessing word centrality is to look at the centroid of the document cluster – a pseudo-document in a vector space. The pseudo-document consists of words that have *tf-idf* scores above a predefined threshold. Centroid is used as a measure to check how close the sentences are to the centroid of the cluster.

LexRank is an extractive summarization method for computing sentence importance based on the concept of eigenvector centrality in a lexical similarity graph representation of sentences. LexRank accounts for information subsumption among sentences. i.e., it is naturally preferred to include the one that contains more information in the summary.

Opinosis The Opinosis summarizer[6] is considered a "shallow" abstractive summarizer, as it uses the original text itself to generate summaries (this makes it shallow). But, it can generate phrases that were previously not seen in the

[6] http://kavita-ganesan.com/opinosis

original text because of the way paths are explored and this makes it abstractive rather than purely extractive.

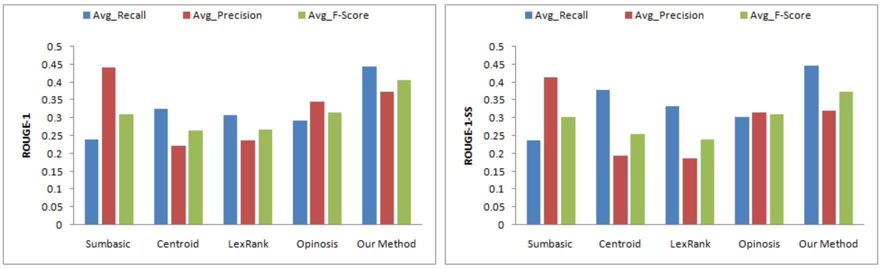

Fig. 4. Comparison of related Approaches on #jallikattu dataset : ROUGE-1 with and without Stop-word Removal and Stemming

For each of the trending topics considered, we generated summaries from all the four baseline methods and our proposed method. We used the tweet sentences (sentences identified from unique set of tweets) as input for each of these systems.

In the experimental setup, we considered all baseline methods with default settings and generated a summary in about 10 sentences. We used MEAD [15],[7] – a publicly available toolkit for extractive multi-document summarization. It consists of centroid-based summarization, as well as feature for LexRank based summarization. In the centroid based MEAD summary, we assigned position feature to be 0, so that all tweet sentences will get equal priority. We also used length feature, which is set to 3, to discard the tweet sentences which doesn't convey any meaning. Similar features are used to generate the LexRank MEAD summary. As there are multiple sentences in each cluster, and more than 10 total number of clusters are identified from our approach, we selected 10 sentences from the top-most 10 clusters as summary from our system. We computed the lexical overlap between summaries, in-terms of uni-gram (ROUGE-1) and bi-gram (ROUGE-2) scores with and without standard stop-word removal and stemming. We used the default stop-word list and english stemmer from ROUGE package[8]. The ROUGE evaluation results on #jallikattu dataset are shown in Figure 4 to Figure 5.

The ROUGE-1 score shows the overlap of information content with respect to the gold standard summaries and ROUGE-2 shows the readability of the system generated summary compared to the gold standard summary. We could achieve similar or better performance while comparing with all the baselines. The focus of our summarization approach is to assist in identifying the key entities and key actions of the plot along with key sentences. Each edge in the abstract

[7] http://www.summarization.com/
[8] http://www.rxnlp.com/rouge-2-0/

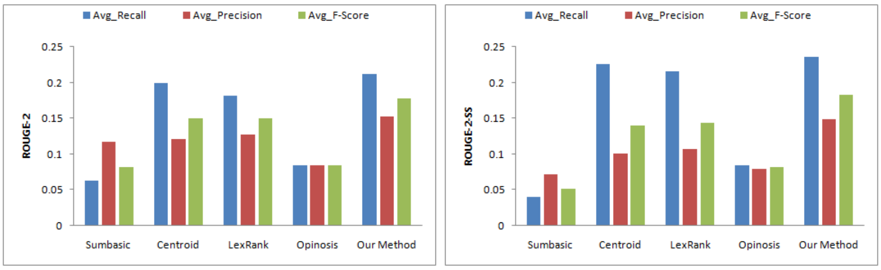

Fig. 5. Comparison of related Approaches on #jallikattu dataset : ROUGE-2 with and without Stop-word Removal and Stemming

model allows to focus only on a set of sentences, based on entities and actions of interest.

5 Conclusions

We have presented a novel unsupervised approach to extract key sentences from tweets related to trending topics. Using a pair-wise MRF factor computation technique, we were able to characterize the plot of a trending topic, in-terms of key entities and key-actions. The proposed method based on identifying activity triples, seems to be more effective for summarizing micro-documents like tweets as compared to sentence saliency computed based on centrality measures and longest subsequence based approaches.

References

[1] Cao, Z., Chen, C., Li, W., Li, S., Wei, F., Zhou, M.: Tgsum: Build tweet guided multi-document summarization dataset. In: AAAI. pp. 2906–2912 (2016)
[2] Chakrabarti, D., Punera, K.: Event summarization using tweets. ICWSM 11, 66–73 (2011)
[3] Chang, Y., Tang, J., Yin, D., Yamada, M., Liu, Y.: Timeline summarization from social media with life cycle models. In: IJCAI. pp. 3698–3704 (2016)
[4] Dutta, S., Ghatak, S., Roy, M., Ghosh, S., Das, A.K.: A graph based clustering technique for tweet summarization. In: Reliability, Infocom Technologies and Optimization (ICRITO)(Trends and Future Directions), 2015 4th International Conference on. pp. 1–6. IEEE (2015)
[5] Erkan, G., Radev, D.R.: Lexrank: Graph-based lexical centrality as salience in text summarization. Journal of Artificial Intelligence Research 22, 457–479 (2004)
[6] Ganesan, K., Zhai, C., Han, J.: Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions. In: Proceedings of the 23rd international conference on computational linguistics. pp. 340–348. Association for Computational Linguistics (2010)

[7] Ganesan, K., Zhai, C., Viegas, E.: Micropinion generation: an unsupervised approach to generating ultra-concise summaries of opinions. In: Proceedings of the 21st international conference on World Wide Web. pp. 869–878. ACM (2012)

[8] He, R., Liu, Y., Yu, G., Tang, J., Hu, Q., Dang, J.: Twitter summarization with social-temporal context. World Wide Web 20(2), 267–290 (2017)

[9] Huang, Y., Shen, C., Li, T.: Event summarization for sports games using twitter streams. World Wide Web pp. 1–19 (2017)

[10] Inouye, D., Kalita, J.K.: Comparing twitter summarization algorithms for multiple post summaries. In: Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on. pp. 298–306. IEEE (2011)

[11] Koller, D., Friedman, N., Getoor, L., Taskar, B.: Graphical models in a nutshell. Introduction to statistical relational learning pp. 13–55 (2007)

[12] Lin, C.Y.: Rouge: A package for automatic evaluation of summaries. In: Text summarization branches out: Proceedings of the ACL-04 workshop. vol. 8. Barcelona, Spain (2004)

[13] Lin, C.Y., Hovy, E.: Automatic evaluation of summaries using n-gram cooccurrence statistics. In: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1. pp. 71–78. Association for Computational Linguistics (2003)

[14] Liu, X., Li, Y., Wei, F., Zhou, M.: Graph-based multi-tweet summarization using social signals. In: COLING. vol. 12, pp. 1699–1714 (2012)

[15] Radev, D.R., Allison, T., Blair-Goldensohn, S., Blitzer, J., Celebi, A., Dimitrov, S., Drabek, E., Hakim, A., Lam, W., Liu, D., et al.: Mead-a platform for multidocument multilingual text summarization. In: LREC (2004)

[16] Radev, D.R., Jing, H., Styś, M., Tam, D.: Centroid-based summarization of multiple documents. Information Processing & Management 40(6), 919–938 (2004)

[17] Sharifi, B.P., Inouye, D.I., Kalita, J.K.: Summarization of twitter microblogs. The Computer Journal p. bxt109 (2013)

[18] Vanderwende, L., Suzuki, H., Brockett, C., Nenkova, A.: Beyond sumbasic: Task-focused summarization with sentence simplification and lexical expansion. Information Processing & Management 43(6), 1606–1618 (2007)

[19] Xu, W., Grishman, R., Meyers, A., Ritter, A.: A preliminary study of tweet summarization using information extraction. NAACL 2013 p. 20 (2013)