Proceedings of the Twenty-Eighth International Florida Artificial Intelligence Research Society Conference

Event Analysis in Social Media Using Clustering of Heterogeneous Information Networks

Narumol Prangnawarat, Ioana Hulpuş, Conor Hayes
The Insight Centre for Data Analytics
National University of Ireland, Galway
{narumol.prangnawarat, ioana.hulpus, conor.hayes}@insight-centre.org

Abstract

In this paper, we propose a novel approach for social media event finding in order to support fast access to information that users find relevant. While there are many approaches related to this problem, they mainly focus on homogeneous data, such as either the text of the posts, or the network of users. Our research focuses on combining multiple types of data from social media in a heterogeneous network. We propose different graph-based models using users, posts, and concepts extracted from the post content to represent the social media network. We analyse the resulted heterogeneous network, and use it in order to cluster posts by different topics and events. Our preliminary results show improvement over the methods that typically use only one type of data.

Introduction

Social media has a great influence in our daily lives. People share their opinions, stories, news, and broadcast events using social media. This results in great amounts of information in social media. Methods to organise social media posts to support more informative views of data to users are needed so that users can easily find groups of posts that they are interested in. For example, clustering relevant topics together allows business users to go directly to the cluster of business related events.

Many approaches for data mining and analysis for clustering and event detection in social media have been researched, but most of them consider content-based analysis or analysis using one type of data as a homogeneous network. Lau et al. (Lau, Collier, and Baldwin 2012) used LDA (Latent Dirichlet Allocation) topic models over only the Tweets content for grouping and detection of events. Benhardus et al. (Benhardus and Kalita 2013) used term frequency-inverse document frequency (tf-idf) and normalised term frequency analysis to detect streaming trends in Twitter. Ifrim et al. (Ifrim, Shi, and Brigadir 2014) proposed a topic detection method in Twitter streams based on aggressive term filtering and hierarchical clustering of Tweets on the tweet-term matrix. On the other hand, Cataldi et al. (Cataldi, Di Caro, and Schifanella 2010) studied relationships between users to find importance of contents, and detected emerging topics by modelling the term life cycle of contents extracted from Tweets in the specified time interval. Hromic et al. (Hromic et al. 2012) proposed a methodology for filtering, grouping and ranking Twitter streams and providing breaking news to end-users using user interaction networks.

In our research, we leverage the idea of using relations between users, posts and concepts generated from texts in graph-based analysis, all at the same time. Furthermore, we analyse the impact of the choice of node types over clustering result and the model to represent the heterogeneous network of social media. Then, we show that clustering of the heterogeneous network can be used to group posts into relevant topics or events.

In the next section, we describe the models we use to construct the heterogeneous social media graph as well as our proposed approaches. Afterwards, we explain our preliminary experimental results. The conclusion is given in the last section.

Proposed Approaches

Our hypothesis is that the analysis of interaction between users and posts, together with interconnection of posts' content, can bring benefits to post clustering and ranking. Twitter is representative of social media used in this research.

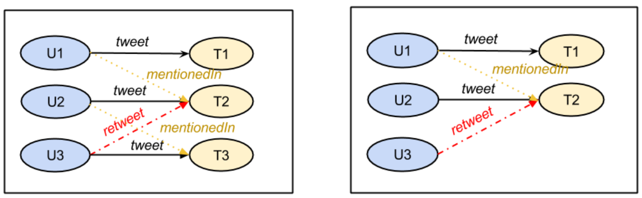
We model the Twitter network in two different types of networks in order to find the appropriate node types to use in heterogeneous information network. The first network type is a bipartite graph between users and Tweets. The second network type adds *concepts* from the Tweets' content as a concept type of nodes in the network. Three different types of concepts are taken into account as described in the detail of the model.

User-Tweet (U-T) Model

The bipartite graph between users and Tweets is built by aggregating *tweet, retweet, reply* and *mentionedIn* relationships into weighted edges.

In the case of *retweets*, the original Twitter data contains new tweets corresponding to the retweet action, as illustrated in Figure 1a. In the example in the figure, user $U3$ retweeted $T2$ originally created by user $U2$. This action triggers in the data the creation of tweet $T3$. Therefore $T3$ and $T2$ have the same content, $T3$ being just a copy of $T2$, made by a different user. To eliminate this duplication of content, in our graph representation we omit $T3$ and only represent tweet $T2$, but we add the $retweet$ edge between $U3$ and $T2$ as shown in Figure 1b.

With respect to edge types, for simplicity, in this work we currently ignore them, and we give all edge types a weight of 1. In the case of multiple edge types occurring between the same pair of nodes, (for example, if a user retweets a tweet he is mentioned in), we set the weight of the edge between the nodes as the count of relationships between the nodes.



(a) Interaction in Twitter data

(b) User-Tweet model

Figure 1: Interactions of users and Tweets in Twitter data and User-Tweet model

Definition 1. *A* ***User-Tweet graph (U-T graph)*** *is a directed weighted graph $G_{UT} = (V, E, T, \phi, \pi)$ where V is the set of vertices, T is the set of node types in this case $\{User, Tweet\}$, E is the set of edges connecting nodes of type $User$ to nodes of type $Tweet$, $\phi : V \rightarrow T$ is a function mapping vertices in V to types in T, and $\pi : E \rightarrow \mathbb{R}^+$ is the weighting function mapping edges to real positive numbers.*

User-Tweet-Concept (U-T-C) Models

U-T-C model extends the U-T model by adding concept nodes extracted from Tweets content into the network. We analyse three kinds of concepts to build three different U-T-C models:

- User-Tweet-Hashtag (U-T-$\mathrm{C^H}$) model extends the U-T model by using hashtags, the word or phrase starting with a hash sign(#) to identify specific topic in Tweets, as the concept type.
- User-Tweet-Entity (U-T-$\mathrm{C^E}$) model extends the U-T model by extracting named entities from Tweet texts as the concept type. Since Tweets might not have a hashtag, adding named entities as concept type can ensure that relevant Tweets become connected.
- User-Tweet-MixedConcept (U-T-$\mathrm{C^M}$) model extends the U-T model by extracting named entities from Tweet texts as well as hashtags used in the Tweets as the concept type.

Definition 2. *A* ***User-Tweet-Concept graph (U-T-C graph)*** *is a directed weighted graph $G_{UTC} = (V, E, T, \phi, \pi)$ where V is the set of vertices, T is the set of node types in this case $\{User, Tweet, Concept\}$, E is the set of directed edges connecting nodes of type $User$ to nodes of type $Tweet$, and nodes of type $Tweet$ to nodes of type $Concept$, $\phi : V \rightarrow T$ is a function mapping vertices in V to types in T, and $\pi : E \rightarrow \mathbb{R}^+$ is the weighting function mapping edges to real positive numbers.*

Entities extracted from the text of Twitter posts can be polysemous - the same word can have multiple different meanings. Another problem when linking entities from text is synonymy - different words can bare the same meaning. Adding semantic knowledge can solve this problem. We propose to link concepts extracted from posts to DBpedia in order to eliminate ambiguous and confusion in words.

Posts Clustering and Ranking

We assume that an interaction that has only one or two people involved in the discussion does not qualify as an event or a topic since the group of discussion is too small. Therefore, we analyse the graphs after removing the small connected components that consist of only one or two users from all models.

Then, we extend the state-of-the-art RankClus (Sun et al. 2009) algorithm designed for heterogeneous networks to apply to modelled Twitter data. The algorithm is also scalable for big datasets. RankClus integrates clustering and ranking by using rank distribution as the feature of clustering. In our experiment, HITS (Kleinberg 1999) algorithm is used as the ranking function.

Given the graph and the number of clusters as input, the results are clusters of each node type with ranking of nodes in the clusters. After that, we filter out low ranked Tweets and concepts in each cluster. For this, we set a threshold equal to the ranking score of a tweet in a cluster under a uniform distribution assumption. Thus, we remove the Tweets with lower rank than the ratio of $1/|Tweets|$. The same assumption can be applied for the concept type, as follows:

$$TopTweets = \{t; t \in Cluster \wedge R_t \geq 1/|Tweets|\}$$

$$TopConcepts = \{c; c \in Cluster \wedge R_c \geq 1/|concepts|\}$$

Where R_t is the ranking score of the Tweet t, R_c is ranking score of the concept c, $|Tweets|$ is number of Tweets in the cluster and $|concepts|$ is number of concepts in the cluster. Concepts in each cluster are used as labels to represent the topic of the clusters.

Preliminary Results

We analysed how well the obtained clusters of posts correspond to events. For this, we analysed two clustering methods: modularity-based clustering with Louvain method (Blondel et al. 2008), and the RankClus algorithm described in the previous section.

Dataset and Model

We use the dataset from (McMinn, Moshfeghi, and Jose 2013) which proposed a Twitter data corpus using state-of-the-art event detection approaches and Wikipedia Current Events Portal[1] to generate a set of twitter events. The data was also manually judged by crowd-sourcing to ensure integrity of the result. Tweets and users can be annotated to more than one event. Also, the dataset does not include reply Tweets. Two datasets, as described in the following table, are generated from the corpus with the same annotated Tweets covering 57 events but different number of background Tweets, Tweets that we do not annotate as events.

[1] http://en.wikipedia.org/wiki/Portal:Current_events

	Total Users	Total Tweets	Background Tweets
dataset1	19,127	15,461	9,640
dataset2	108,770	98,065	92,244

Network Components

We first analysed how users and Tweets interact to form connected components (CCs) in the network. We found that there are several small connected components that do not interact with the others in the networks. As mentioned in previous section, we removed connected components with only one or two users from all models. This removes a large number of nodes and speeds up the processing time while we still can capture discussions among many users. Figure 2 shows the number of connected components in each models from the dataset2. The same trend applies in the dataset1.

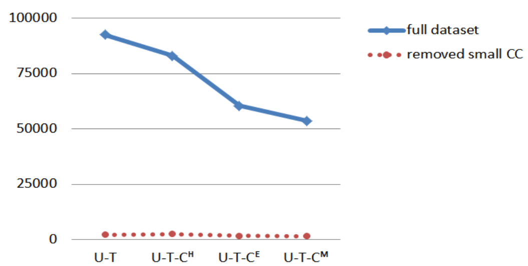


Figure 2: Graph comparing number of connected components of each models.

In the U-T model, components that consist of one event usually appear to have a user in the middle of the component as a key person or a user who initially posted about an event and got re-tweeted by other users. For example: Kendrik Lama(@kendriklama) was mentioned in several Tweets when he won the award so he become the centre of the component of the event "lyricist of the year". Another behaviour is when Tweets are re-tweeted by several people and appear to be in the centre of components. On the other hand, components that consist of multiple events usually have a user in the middle as a creator of Tweets. This user appears to tweet about many different topics. Our analysis indicates that such users that are central in components that contain many events are news channels. This behaviour brings challenges for graph clustering of Tweets into correct events.

We similarly analysed the U-T-C models. The same behaviours as U-T model still can be captured plus concepts appear as hubs connecting different User-Tweet interactions together. From the interaction of users and Tweets in U-T model, when a middle node is a user who is a news agency or a reporter, concepts in the connection will not be in common but will join common topics of discussion together, making concepts become middle nodes of the components in U-T-C model. On the other hand, when the middle node is a user mentioned in many Tweets, as mentioned before in @kendriklama case, the concepts in the connection are in common and remain the user as middle of the component.

Local Events and Global Events

After analysing networks represented by different models in both datasets, we found common interesting properties of events in the networks on how users and Tweets, as well as concepts in U-T-C models, interact when events occur. Events can be classified in two different types of events by interaction within the network.

Local Events are events occurred in local areas or in specific user communities. This kind of events are discussed among small group of people and are not discussed widely in other communities. An example of this kind of event is a discussion about new CEO in a small company. Local Events can usually be captured within the same connected component in the U-T model.

Global Events are events that occur and discussed in different communities. Users from different communities may not interact with others even if they discuss about the same topics. World Cup is an example of a Global Event discussed around the world. User-Tweet relations alone usually cannot capture Global Events within the same connected component since users do not have a connection with people from different regions of the network. U-T-C models that consider concepts help connect different interaction communities discussing the same topics together. Figure 3 shows entities as hubs connecting different User-Tweet interaction together. In the figure, Tweets are coloured according to the events in the dataset. Users are coloured in white and concepts are coloured in black.

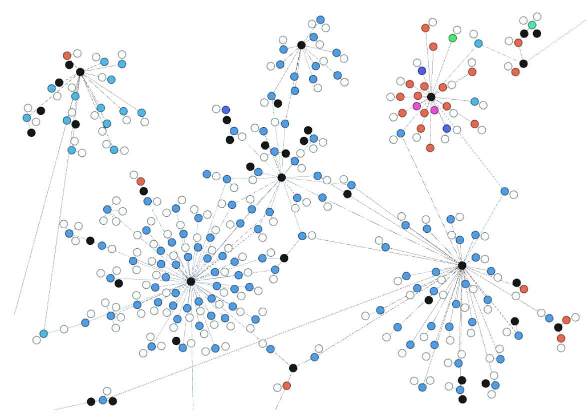


Figure 3: A part of network showing concepts as hubs to connect Tweets from the same events which do not have interaction to each others. Tweets are coloured according to the events in the dataset. Users are coloured in white and concepts are coloured in black.

Graph Clustering for Event Identification

In order to identify events and discussion topics, we experiment with two graph clustering algorithms, Louvain algorithm in homogeneous information networks and RankClus algorithm in heterogeneous information network.

Modularity-Based Clustering We applied modularity-based clustering with Louvain method to all models in both networks, with and without connected components with only one or two users, to find how the networks decompose into modular communities. In this approach, all nodes are processed as the same type in homogeneous information network.

Figure 4 and Figure 5 show network modularity and number of clusters, respectively, comparing between different models in dataset2. The removal of components with one or two users slightly reduces the network modularity, but also significantly decreases the number of modularity-based clusters found in the networks. When Tweets are clustered into too many clusters, Tweets are too isolated so that Tweets from the same event are assigned to different clusters.

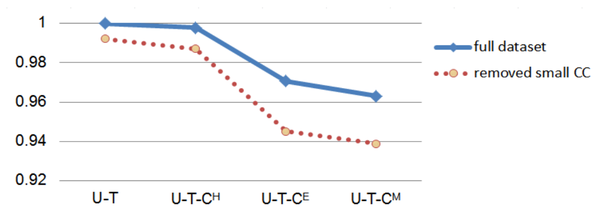


Figure 4: Network modularity comparing between different models

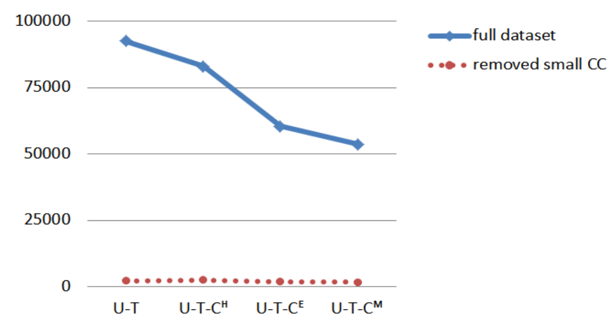


Figure 5: Number of Louvain clusters comparing between different models

Network modularities are slightly lower in the U-T-C models while the number of clusters is much lower in U-T-C^{E} model and U-T-C^{M} model. This means U-T model is clustered into more clusters which causes more dense connections between nodes within the same cluster and more sparse connections between nodes in different clusters than the U-T-C models which slightly loses connection density after connecting components together using concepts. The result shows that U-T-C models can significantly connect isolated interactions together while still preserve dense connection between users and Tweets.

Ranking-Based Clustering We applied RankClus algorithm to the U-T model in all dataset using different predefined numbers of clusters (k) which are 5, 10, 15, 20, 25, 30 and k=$|events|$ where $|events|$ is the number of events found in the annotated dataset. In this approach, all nodes have their own types as heterogeneous information network. We measured BCubed Recall and Precision (Amigó et al. 2009) of the results and found that different k values affect the BCubed Recall and Precision. This is because, Tweets from the same event have higher chance to spread out to different clusters when there are more clusters. On the other hand, related Tweets have more chances to be in the same clusters when there are less clusters. However, having fewer clusters cannot give much information about different events. We also found that filtering to get top ranked Tweets in each clusters, as per the method described in the previous section, also significantly improved the measures since much of the noise is removed.

Comparison We compared results from RankClus algorithm with Louvain clustering results stated above. BCubed precision in modularity-based clusters of both networks, with and without connected components with one or two users, is higher. However their BCubed recall is very low making their F1-Score much lower than using RankClus after filtering out to get only top ranked Tweets.

The results show that the best F1-Score is achieved by using RankClus after removing connected components with one or two users and get only top ranked Tweets. Figure 6 show the comparisons of Louvain modularity clustering which uses homogeneous information network, all nodes are processed as one type, and RankClus algorithm which uses heterogeneous information network, considering types. In this comparison, $k = 30$ is used for RankClus which is closer to the number of clusters in Louvain clustering.

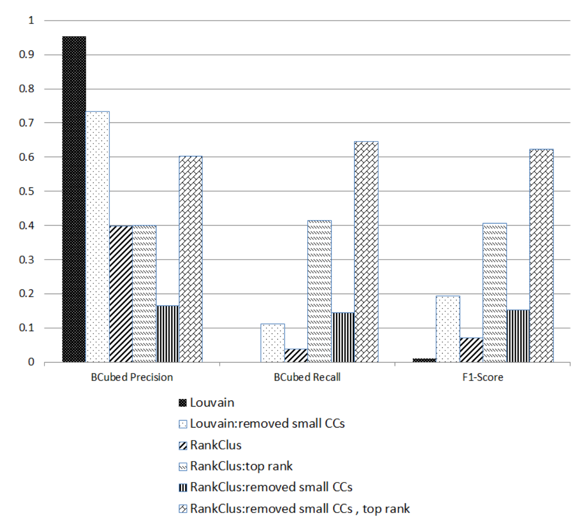


Figure 6: Comparison of BCubed Precision, BCubed Recall and F1-Score in different approaches

Using RankClus algorithm to cluster Tweets based on User-Tweet relations alone may not be sufficient to create complete event clusters especially when the network consists of small connected components. RankClus algorithm cannot capture global events where users are not related. This results in Tweets related to global events being clustered separately as shown in Figure 7. Also, RankClus sometimes groups together posts that are not related to the same events as shown in Figure 8. In both figures, Tweets are coloured according to the events in the dataset and users are coloured in white.

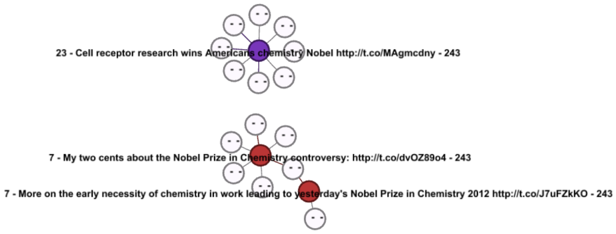


Figure 7: Example of Tweets of the same event but clustered into different clusters. Tweets are coloured according to the events in the dataset. Users are coloured in white.

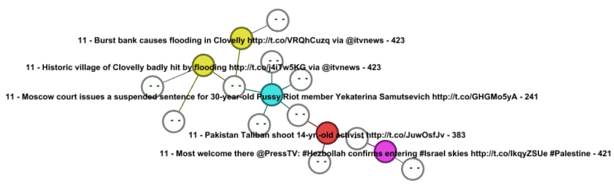


Figure 8: Example of Tweets of different events but clustered into the same cluster. Tweets are coloured according to the events in the dataset. Users are coloured in white.

Conclusion

In this paper we analyse Twitter data based on different models, which are U-T model and U-T-C models, as heterogeneous information network. The result shows improvement of using heterogeneous network over the method which uses one type of data. Also, using U-T-C models can connect data together more than using the interaction between users and Tweets alone. We are working on experiment to extend clustering and ranking on U-T-C models as well as applying to bigger dataset. We believe that considering the network using more data types will improve social media event clustering beyond current state-of-the art.

Acknowledgements

This research supported is funded by Science Foundation Ireland (SFI) under Grant Numbers SFI/12/RC/2289 (Insight) and SFI/08/SRC/I1407 (Clique).

References

Amigó, E.; Gonzalo, J.; Artiles, J.; and Verdejo, F. 2009. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Inf. Retr.* 12(4):461–486.

Benhardus, J., and Kalita, J. 2013. Streaming trend detection in twitter. *Int. J. Web Based Communities* 9(1):122–139.

Blondel, V. D.; Guillaume, J.-L.; Lambiotte, R.; and Lefebvre, E. 2008. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* 2008(10):P10008.

Cataldi, M.; Di Caro, L.; and Schifanella, C. 2010. Emerging topic detection on twitter based on temporal and social terms evaluation. In *Proceedings of the Tenth International Workshop on Multimedia Data Mining*, MDMKDD '10, 4:1–4:10. New York, NY, USA: ACM.

Hromic, H.; Karnstedt, M.; Wang, M.; Hogan, A.; Belak, V.; and Hayes, C. 2012. Event planning in a stream of big data. *LWA Workshop on Knowledge Discovery, Data Mining and Machine Learning*.

Ifrim, G.; Shi, B.; and Brigadir, I. 2014. Event detection in twitter using aggressive filtering and hierarchical tweet clustering. *Proceedings of the SNOW 2014 Data Challenge*.

Kleinberg, J. M. 1999. Authoritative sources in a hyperlinked environment. *J. ACM* 46(5):604–632.

Lau, J.; Collier, N.; and Baldwin, T. 2012. On-line trend analysis with topic models: #twitter trends detection topic model. In *International Conference on Computational Linguistics (COLING)*, 16.

McMinn, A. J.; Moshfeghi, Y.; and Jose, J. M. 2013. Building a large-scale corpus for evaluating event detection on twitter. In *Proceedings of the 22Nd ACM International Conference on Conference on Information & Knowledge Management*, CIKM '13, 409–418. New York, NY, USA: ACM.

Sun, Y.; Han, J.; Zhao, P.; Yin, Z.; Cheng, H.; and Wu, T. 2009. Rankclus: Integrating clustering with ranking for heterogeneous information network analysis. In *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*, EDBT '09, 565–576. New York, NY, USA: ACM.