# TwiBiNG: A Bipartite News Generator Using Twitter

Yashvardhan Sharma
Department of Computer Science
Birla Institute of Technology & Science
Pilani, India 333 031
yash@pilani.bits-pilani.ac.in

Divyansh Bhatia
Department of Computer Science
Birla Institute of Technology & Science
Pilani, India 333 031
h2009399@pilani.bits-pilani.ac.in

Vivek Kishore Choudhary
Department of Computer Science
Birla Institute of Technology & Science
Pilani, India 333 031
f2012650@pilani.bits-pilani.ac.in

## Abstract

Online Journalism is being seen as future of Journalism. News Professionals are vying to capture newsworthy stories that emerge from crowd. Live Social Media especially Twitter is generating enormous volumes of data every minute. It becomes difficult to select credible and relevant tweets that may form quality news among others. The problem intensifies due to the freedom of Twitter being an informal language. Generating headlines by solving this problem may still not be relevant and may face the question of authenticity. Given a set of keywords and a time period this problem becomes manageable and can be solved efficiently. We propose a bipartite algorithm that clusters authentic tweets based on key phrases and ranks the clusters based on trends in each timeslot. Finally, we present an approach to select those topics which have sufficient content to form a story

## 1  Introduction

Journalism is the state of art that disseminates information and provides analysis of news to the general

public. With the advent of Web 2.0 most of the journalism has gone the online way innovating the term "Online Journalism". Since users of the web are ready to share each and every activity they do in their lives due to the free nature of the world, this has made professionals content hungry. Twitter generates an amount of information that can outrun the storage space of many servers in a few months. Developing a user centered tool that can process this information in real time has become need of the day for professional journalists.

From the Arab Spring to the Oscars 2014 Selfie tweets have changed the way the world shares information. Scholars today can predict election results better than ever before [Ocon10]. The "#" Hashtag feature in Twitter has made event stories easier to capture [Zan11]. As a result social network mining, originally loaded with clustering and classification of online worlds, is leveraging on understanding evolution of real-world events [Dom05].Adding another feather to its cap is the fact that newspaper and magazines have started publishing content on social media sites like Twitter and Facebook. To summarize, news no longer breaks it tweets (Solis)[Sol10].

The goal of this paper is to demonstrate the use of Twitter to monitor headlines online and generate news stories. We propose a standalone system TwiBiNG to extract tweets related to user defined keywords and propose ranked news summaries based on trend and relevance of tweets they contain. The key novelty behind TwiBiNG is generation of Bi-partitite clusters of tweet intentions and use of Longest common-subsequence(LCS) algorithm along with a few tweet creator's details to separate relevant tweets from irrele-

vant ones. This approach not only produces better clusters but also generates stories that are authentic, contains less spam and more importantly are distinct from each other. Also since we base our approach on intention of tweets it makes it language independent. Readers should note that by intention we refer to the general subject of tweet; not the intention of the user posting it. The selected datasets were developed from tweets collected between Tue 25 Feb, 18:00 GMT and Wed 26 Feb, 18:00 GMT based on keywords "Syria","Ukraine","Terror","Bitcoin". We collected 1,041,062 unique tweets from 556,295 users which included 648,651 retweets and 135,141 replies. The crawl also included messages sent from or to a set of around 5000 journalists/commentators.

In short our contributions can be summarized as:

- We incorporated retweets in BNgrams clustering [Aie13] and hence improved upon the trend ranking of keywords.

- We clustered our tweets based on bipartitite graph thereby clubbing similar intention tweets together.

- We reduced the effect of informal text in Twitter by using LCS based similarity score while dealing with keywords.

- We presented news headlines by ranking clustered tweets based on relevance to the clustered keyword set and use 'Part Of Speech' tagger to make them readable.

The remainder of the paper is organized as follows: In Section 2 we take a look at existing algorithms and approaches.Section 3 details about proposed methodologies and approaches. Section 4 provides a discussion of results. Section 5 concludes the work by laying a foundation for future work.

## 2 Related Work

The work of generating headlines using social media can be seen as a combination of two branches 1) Information Retrieval and Text Mining and 2) Natural Language Processing. Scholars have worked extensively on Twitter data using both the fields. Here we present an overview of existing approaches in both fields:

### 2.1 Text Mining on Twitter Content

Twitter has its own conventions for language while (@) is used to mention user, (#) is used to identify events and "RT" is used to represent a retweet. Bifet and Frank [Bif10] use these features for opinion mining. Zhao et al.[Zha11] develop a Twitter-LDA model through content analysis. The restricted length (140 characters) and informal text are some issues that pose problems to many text mining researchers (Hong and Davison [Hon10]). Bollen et al. [Bol11] used terms expressing positive and negative behavior for sentiment analysis on Twitter.

Text Clustering is another where scholars have worked for content analysis. Goyal and Mehala [Goy13] presented an approach to find conceptually related queries by clustering on bipartite and tripartite graphs. We try to propose a similar approach for Twitter content analysis using Bipartite graph. [Aie13] proposes trend based tweet clustering approaches. We present an approach that uses a modified BNgram clustering approach, which has motivation from original approach of [Aie13]. Phuvipadawat and Murata [Phu10] present a breaking news prediction algorithm that clusters tweets based on First Story detection after segmenting different stories. TwitterStand [San09] develops a "leader-follower" text clustering algorithm.

### 2.2 Natural Language Processing

Headline Generation has been active area of research among NLP researchers. Most of the scholars work here by selecting a proper set of keywords and finding a way to combine them in a way that forms a grammatically coherent and meaningful sentence. In Banko et al.[Ban00] authors present a statistical approach to term selection and term ordering process that depicts the power of non-extractive summarization whereas Jin and Hauptman [Jin01] presents an approach for extractive summarization along with a Bayesian approach. They also discuss various issues in keyword selection for headline generation. We use Part of speech tagging along with most relevant tweet identification to generate meaningful user readable headline.

## 3 Methodology

We divide our process in four phases 1) Data preparation, 2) Data Clustering 3) Cluster Ranking, 4) Tweet Ranking and Headline generation. We will now describe our TwiBiNG system phase by phase:

### 3.1 Data Preparation

Once the data set for a given timeslot is ready by extracting tweets related to a given set of seeds and keywords, we tag entities in tweets using Stanford's Part-of-Speech Tagger and extract nouns, HashTags, Users. We ignore other parts of speech, thereby concentrating more on the subject than the predicate. This is because in a given timeslot, it is difficult for predicate to change rapidly for the same subject while the reverse may not be true. These tagged words are referred as key phrases (KP) from now on. We now decide on trending keywords.

We rank keywords using a modified df-idft [Aie13] score by incorporating retweets:

$$R(k_i) = \frac{R_i - R_{i-1}}{max(R_i, R_{i-1})}$$

$$Score(k_i) = t_i * log(1 + \frac{R(k_i+1)}{t_{i-1}+1})$$

Here $R_i$ represents number of retweets for keyword $k$ in timeslot $i$ and $t_i$ represents number of tweets for keyword $k$. Since a keyword may be related to unbounded number of tweets and retweets in a timeslot deciding on threshold is difficult. Therefore, we decided to normalize the score for each keyword using min-max normalization. Let $< K >$ be the set of tweets in a slot $i$ then normalized score is given by:

$$NormalizedScore(NK_i) =$$
$$\frac{Score(k_i) - min(Score(< K >))}{max(Score(< K >)) - min(Score(< K >))}$$

The threshold for these normalized keywords was decided to be 0.0075 through experiments. We select the keywords above this threshold and store them in a set $(S_i)$. We observed that for each timeslot at this threshold we get around 800-875 trending keywords. Once this set was ready we assigned tweets to each keyword, i.e. we reversed the bipartite graph of Figure 1. We now filter the tweets based on user details specifically number of followers and status counts. This step is necessary in order to increase authenticity and reduce tweets containing spamming content. Since clustering is based on tweet intention, not performing the previous step may hamper clustering performance. Also the generated stories may not be considered quality news. Our experiments based on (Hutto et. al. [Hut13]) decided that users with a follower count>600 and tweet count>6000 may be considered authentic and considering tweets by these users alone will significantly improve system performance.
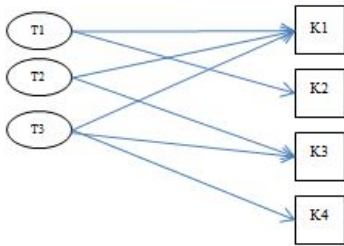


Figure 1: Bipartite Graph for Tweets(T) and Keywords(K)

Now since we are building a user centered news generator we want tweets related to the keywords defined by user to improve relevancy. For this purpose we scan all keywords in $(S_i)$ and compute their Similarity with user-defined keywords $(U_i)$.

$$LCS(S_i, U_i) = LongestCommonSubsequence(S_i, U_i)$$

If any $LCS(S_i, U_i)$ contains $U_i$ then we include all the tweets related to $S_i$ in set $< TU_i >$ which contains tweet ids related to user centered keywords. We scan the database for the timeslot again and remove those tweets which are not contained in $< TU_i >$ (user-centric tweets). At the end of this stage we end up with a set of tweets and related keywords that can be considered authentic for a news story.

## 3.2 Intention based Tweet Clustering

We use the approach used in [Goy13] to use bipartite clustering of tweets. The basic aim here is to get real intention of tweets in clusters. Algorithm 1 presents an incremental bipartite algorithm to cluster tweets and keywords. Once we have a set of clusters we know the intention of tweets. As can be seen the threshold is kept $> 0.5$, which signifies that keywords merged should have an intention similarity of more than 50%. Readers requiring more specific tweets to be clustered together may increase the similarity but this comes at a cost of duplicate tweets being merged together. As can be observed in Algorithm 1, since the clustering is on basis of basis of Intersection($T_i$,$T_j$) there will be duplicate tweets in cluster but a news story containing a lot of duplicate tweets would be considered of poor quality. So removing duplicate content becomes a prime task now.

**Data**: I$< S_i, < TS_i >> S_i$ and $TS_i$ denotes a set of keywords and related tweets
**Result**: O$< CS_i, < CTS_i >>$ clustered set of tweets
Let S: represent set of unique keywords
**while** *clusters exist with similarity $>$ threshold* **do**
    flag=0;
    **while** $s_i$ *in* $S$ **do**
        j=i+1;
        **while** $t_j$ *in* $T$ **do**
            Sim($s_i$,$s_j$)
            =Intersection($Ts_i$,$Ts_j$)/Union($Ts_i$,$Ts_j$);
            **if** $Sim\ (s_i, s_j) > 0.5$ **then**
                I$< s_i, < Ts_i >> =$
                I$< s_i = s_j, < Union(Ts_i, Ts_j) >>$
                Remove $s_j$ from I flag=1;
            **end**
        **end**
        **if** *flag=0* **then**
            b
        **end**
        reak;
    **end**
**end**

Algorithm 1: Bipartite Clustering of Tweets using Keywords

In Algorithm 2 we present an algorithm to remove du-

plicate tweets from cluster:

**Data**: $< CS_i, < CTS_i >>$ Set of tweets in a cluster of keywords $CS_i$

**Result**: : $< CS_i, < FTS_i >>$ Final Set of tweets and clusters

**while** $cs_i$ *in* $CS_i$ **do**
    **while** $t_i$ *in* $CTS_i$ **do**
        j=i+1
        **if** $< D_i >.contains< t_j > = false$ **then**
            **while** $t_j$ *in* $CTS_j$ **do**
                $sim(t_i, t_j)=$
                $LCS(t_i, t_j)/Min(t_i.length, t_j.length)$
                **if** $sim(t_i, t_j) > 0.65$ **then**
                    $< D_i >.add(t_j);$
                **end**
            **end**
        **end**
    **end**
    $< FTS_i > = < CTS_i >-< D_i >;$
    $< CS_i, < CTS_i >> =< CS_i, < FTS_i >>$
**end**

Algorithm 2: To remove Duplicate Tweets from Cluster

The motivation behind threshold of 0.65 in Algorithm 2 can be observed in O'Connor [Oco10]. We end this phase with a cluster of keywords and their relevant set of tweets. So now we know the intention of our keywords and we are ready to rank them.

### 3.3 Cluster Ranking

Up until this phase we have obtained required set of clusters. We now need to rank them. Although different authors [Yaj12][Hav03][Shu11] have proposed efficient topic ranking methods they have a common feature that relevance to considered keywords is considered an important issue. We make use of this fact and of normalized trend score to generate a ranking score for clusters. Since we are vying for a user centric tool our clusters should be most relevant to their intention. Also since we have to generate headlines trend needs a special attention. Keeping the above two facts we present our cluster ranking methodology. Using $< U_i >$ we collected tweets for relevant keywords in section 3.1 as set $< TU_i >$. We calculate Relevancy of cluster $CS_i$ having tweets $< FS_i >$ as:

$$RCS_i = Relevancy(CS_i) = \frac{Max(Intersection(U_i, FS_i))}{Union(U_i, FS_i)}$$

This relevancy score gives us an indication about the relation of cluster to the user's intention.

$$TCS_i = Trend(CS_i) = e^{-Max(NormalizedScoreofCSi)}$$

This factor indicates that how much a cluster is trending. The idea of taking Max(Normalized Score of $CS_i$)

has its Motivation from BNgram clustering approach used in [Aie13]. Readers can think of $TCS_i$ as a boost factor for relevance.

$$ClusterScore(CScr_i) = RCS_i * TCS_i$$

We now rank the clusters based on ($CScr_i$ ). At the end of this phase we have ranked our clusters and to avoid any confusion further we now refer them as $< CS_{ir}, < FTS_{ir} >>$.

### 3.4 Tweet Ranking in Clusters

Now once clusters are ranked we need to rank tweets contained in them in order to present them in most relevant order. Before introducing ranking calculations we need to introduce expanded keyword set. This can be seen as a prerequisite in the step of headline formation. This step is necessary and relevant since some of the clusters may contain a small number of keywords and need sufficient information to generate a story. We represent the expanded cluster set as $< ECS >$ . Let set $< K_t >$ represent set of keywords for tweet $T_i$. Then relevance score for $T_i$ is calculated as

$$Score(Ti) = \frac{Intersection(< K_t >, < ECS_i >)}{Union(< K_t >, < ECS_i >)}$$

Now we rank our tweets based on Score($T_i$). At the end of this phase, we filter out tweets which have a score($T_i$) ¡ 0.3. The threshold 0.3 is based on the results of our experiments, as described in Table 2. Increasing the threshold provides better quality stories but reduces the number of stories at a high rate. Hence, readers requiring more focused stories may increase the threshold.

### 3.5 Cluster Selection and Headline Generation

In this phase we provide an approach to decide which clusters can form news. As can be observed not all clusters form a story, we must judiciously decide on clusters to form news. By experiments, we observed the following Heuristic may be used to select quality clusters: H3.5.1: Those clusters tend to form quality stories which contain at least four keywords, one Hashtag keyword, and is related to at least three tweets .Further , number of non Hashtag keywords should be more than Hashtag keywords.

The rationale behind this approach can be explained. The clusters having excessive amounts of hashtags as keywords are usually related to tweets with almost similar content. Having a hashtag allows users to easily identify events and more than three distinct tweets allows us to form a sequence of events. Since, we are needed to identify a fixed number of topics, we

follow H3.5.1 and scan all the clusters in $< Cs_{ir} >$ up until the specified number of clusters in each timeslot. Hence, we follow a dynamic approach that is independent of cluster count.

For Headline Generation we order the keywords in accordance to top ranked tweet in cluster and use POS tagger to connect the keywords. We believe that better approaches to form headlines exist, but we were dealing with informal language so we need to take support from tweet intent to form them. Readers may improve upon this aspect by considering statistical techniques mentioned in section 2.2.

## 4    Results and discussion

Table 1 depicts human evaluation of results as carried out by authors. The official evaluation results of our method in the Data Challenge are included in snow2014dc [Pap14]. The language content shows that our topics were evenly distributed between English and non-English tweets. This is probably due to selection of keywords related to Syria and Ukraine, which allowed foreign phrases to come in the dataset. News Headline Readability being a highly subjective attribute, needs to be evaluated manually. A News Headline is considered readable if majority of the users accessing the system can comprehend it without the use of other resources. Further, it can be observed that 81.60% of our topics were labeled readable by language experts. The images related to the extracted tweets were found to symbolize the news story with 97.67% accuracy.

Table 2 represents the number of topical clusters with increasing score(Ti) threshold. As can be observed, number of clusters decrease at a high rate with respect to the threshold value. Thereby, allowing us to select 0.3 as our base threshold.

Table 1: Human Evaluation of topics

| Language | English | 282 |
|---|---|---|
| | Non-English | 256 |
| News Headline Readability | Good | 439 |
| | Bad | 99 |
| Topics with images | Related | 84 |
| | Unrelated | 2 |

Table 2: Number of clusters v/s Score(Ti) Threshold

| Threshold | 0.25 | 0.30 | 0.35 | 0.40 |
|---|---|---|---|---|
| No. of Clusters | 754 | 538 | 467 | 261 |

Table 3 represents sample topics along with Headline, timestamp, related tweets and set of keywords. The readers should note that not all the tweets in the story are covered, but only the most relevant are shown for clarity.These results show an improved performance over previously existing systems. A limitation of this system is not including user's community which may have allowed us to form tripartite clustering, thereby improving clustering quality at a low cost. Use of better known String matching algorithms may improve cluster quality. Our use of bipartite clustering algorithm can allow future researchers to explore more into this field.

## 5    Acknowledgement

## References

[Ocon10] O'Connor, B., Balasubramanyan, R., Routledge, B. R., & Smith, N. A. (2010), *From tweets to polls: Linking text sentiment to public opinion time series*, ICWSM,11,122-129 .

[Zan11] Zangerle, E., Gassler, W., & Specht, G. (2011). *Recommending#-tags in Twitter.* In Proceedings of the Workshop on Semantic Adaptive Social Web (SASWeb 2011). CEUR Workshop Proceedings (Vol. 730, pp. 67-78).

[Dom05] Domingos, P. (2005). *Mining social networks for viral marketing.* IEEE Intelligent Systems, 20(1), 80-82.

[Sol10] Solis, B. (2010). *The information divide between traditional and new media*, http://www.briansolis.com/2010/02/the-information-divide-the-socialization-of-news-and-dissemination/, Internet Draft (last accessed March 16, 2014)

[Aie13] Aiello, L., Petkos, G., Martin, C., Corney, D., Papadopoulos, S., Skraba, R., Goker, A., Kompatsiaris, I., Jaimes, A. (2013) *Sensing trending topics in Twitter.* Multimedia, IEEE Transactions on 15(6) 2681282.

[Bif10] Bifet, A., & Frank, E. (2010).*Sentiment knowledge discovery in Twitter streaming data.* In Discovery Science (pp. 1-15). Springer Berlin Heidelberg.

[Zha11] Zhao, W. X., Jiang, J., Weng, J., He, J., Lim, E. P., Yan, H., & Li, X. (2011).*Comparing Twitter and traditional media using topic models.* In Advances in Information Retrieval (pp. 338-349). Springer Berlin Heidelberg.

[Hon10] Hong, L., & Davison, B. D. (2010).*Empirical study of topic modeling in Twitter.* In Proceedings of the First Workshop on Social Media Analytics (pp. 80-88). ACM.

[Bol11] Bollen, J., Mao, H., & Pepe, A. (2011). *Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena.* In ICWSM.

[Goy13] Goyal, P., Mehala, N., & Bansal, A. (2013). *A robust approach for finding conceptually related queries using feature selection and tripartite graph structure.* Journal of Information Science, 39(5), 575-592.

[Phu10] Phuvipadawat, S., & Murata, T. (2010). *Breaking news detection and tracking in Twitter*. In Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on (Vol. 3, pp. 120-123). IEEE.

[San09] Sankaranarayanan, J., Samet, H., Teitler, B. E., Lieberman, M. D., & Sperling, J. (2009). *Twitterstand: news in tweets*. In Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (pp. 42-51). ACM.

[Ban00] Banko, M., Mittal, V. O., & Witbrock, M. J. (2000). *Headline generation based on statistical translation*. In Proceedings of the 38th Annual Meeting on Association for Computational Linguistics (pp. 318-325). Association for Computational Linguistics.

[Jin01] Jin, R., & Hauptmann, A. G. (2001). *Generation Using a Training Corpus*. In Computational Linguistics and Intelligent Text Processing (pp. 208-215). Springer Berlin Heidelberg.

[Yaj12] YaJuan, D. U. A. N., WEIF uRu, C. Z., Heung, Z. M., & Shum Y. (2012). *Twitter topic summarization by ranking tweets using social influence and content quality*. In Proceedings of the 24th International Conference on Computational Linguistics (pp. 763-780).

[Hav03] Haveliwala, T. H. (2003). *Topic-sensitive pagerank: A context-sensitive ranking algorithm for web search*. Knowledge and Data Engineering, IEEE Transactions on, 15(4), 784-796.

[Shu11] Shubhankar, K., Singh, A. P., & Pudi, V. (2011). *An efficient algorithm for topic ranking and modeling topic evolution*. In Database and Expert Systems Applications (pp. 320-330). Springer Berlin Heidelberg.

[Pap14] Papadopoulos S., Corney D., Aiello L. (2014). *SNOW 2014 Data Challenge: Assessing the Performance of News Topic Detection Methods in Social Media.* In Proceedings of the SNOW 2014 Data Challenge.

[Hut13] Hutto, C. J., Yardi, S., & Gilbert, E. (2013). *A longitudinal study of follow predictors on twitter*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (pp. 821-830). ACM.

[Oco10] O'Connor, B., Krieger, M., & Ahn, D. (2010). *Tweet-Motif: Exploratory Search and Topic Summarization for Twitter*. In Proceedings of the 4th Int'l AAAI Conference on Weblogs and Social Media.

Table 3: Sample Headlines and Stories

| HEADLINE | KEYWORDS | TWEETS |
|---|---|---|
| Syria alQaeda leader gives rivals ultimatum. <br><br> (25-02-14 19:00) | Rivals, alQaeda, #Syria, group, ultimatum | 1) #Syria #Homs #Aleppo Leader of Syrian militant group challenges rivals <br> 2) RT: Top al-qaeda leader abu khalid al-Suri was reportedly killed by a rival.#Syria |
| Ukraine parliament wants Yanukovich tried international court <br> (25-02-14 18:45) | Yanukovich, Ukraine, parliament, court | #ukraine Rada says try Yanukovich before Int Crime Court. Should be tried by Ukrainians for crimes against Ukrainians! <br> 2) Yanukovich papers:Snipers who killed dozens of protesters came from Ukraine's "omega" special forces.#euromaiden |
| Russian President Vladimir Putin ordered test combat readiness for troops stationed region that touches Ukraines northern border <br> (26-02-14 17:30) | Vladimir, Putin, Yanukovych, Russia, Ukraine | 1) Putin orders troops to prepare in case of 'a crisis' in Ukraine as tensions step up. Report on The 530 now @tv3News <br> 2) Russia puts troops on alert amid Ukraine tension. |
| Ukraine leaders disband riot police who kneel down ask forgiveness from the people <br> (26-02-14 17:45) | riot, Ukraine, police, unit, crackdown, Kiev, protesters | Not in my wildest dreams I'd imagine Arab police doing so #Ukraine riot police asking forgiveness from protesters |
| Bitcoin turmoil rumoured 375m theft closes major exchange. <br> (26-02-14 03:30) | theft, Gox, Bitcoin, exchange | The equivalent of war when states are in danger. <br><br> Bitcoin exchange fears $400m theft #bitcoin |
| bitcoin major turning point mtgox exchange abruptly stops trading 774000 bitcoin reported missing <br> (26-02-14 03:45) | time, website, transactions, being, Bitcoin | # Business?Exchange closes as virtual money vanishes: ANGRY Bitcoin enthusiasts are protesting the collapse |