I J C T A, 9(19) 2016, pp. 9151-9159

Twitter-based Semantic Approach to Multi-Class Classification

Indrajit Mukherjee* **and Kirti Vardhan Rathore***

ABSTRACT

Twitter has established itself as a platform for real-time information sharing over the last decade. As such, various marketing organizations seek to mine Twitter for information about what people think about their services and products. With 500 million tweets being generated in a day, data scientists have used tweet classification as a powerful organizational tool. However, traditional text classifiers are blind to the semantic information expressed in their subject text. While bag of words based approaches have effectively been used widely, the data points they draw inferences from – word frequencies- are ill-suited to an unstructured sparse data source such as Twitter. This paper delivers an intuitive approach to classify tweets based on their semantic features sets. The propose classifier has been designed from the ground up with language semantics and data sparseness in mind. It is found that the classification accuracy is improved over a minimal set of features compared to a bag of words based approach. The proposed method was also compared with the state-of-the-art baseline over one open data set, and reduced the classiûcation error by 29% comparatively.

Keywords: Twitter, POS, WordNet, Classification

I. INTRODUCTION

The growing popularity of social media has led to an unprecedented increase in user created content. In the recent past, Twitter has been widely used for checking the social pulse about real world events and entities. Twitter mining has helped analyze trends and monitor social response to various issues.

Twitter is inherently noisy, uses colloquial language and often provides approximate information. Part-of-speech (POS) tagging based approaches have been used to work with ambiguous data as they are able to predict the syntactic category of words in text with a high degree of accuracy. Gimpel et al.[1] first developed a POS tag set for Twitter. However their rich linguistic utility to microblog classification is a hitherto unexplored topic.

Classification of Tweets on the past research has focused on domain specific feature selection Sriram et al. [2] and standard BOW based classification techniques (Sankaranarayanan et al.[3]. While rule based systems are not scalable to microblogging data outside Twitter, standard classification techniques ill-suit microblog services due to the inherent sparseness of data - Twitter has a 140 character limit for each tweet. Also, the *meaning* of text is not taken into account in traditional word frequency based classification techniques.

The paper presents a semantic approach to the classification of Tweets in Twitter into predefined categories. All the terms contained in a Tweet do not contribute an equal amount of information towards determining the appropriate class for the Tweet. From a linguistic viewpoint, the main “building blocks” of a sentence are Noun Phrases (NP) and Verb Phrases (VP). Noun Phrases are usually the topics or objects in the sentence or in simple words – this is what the sentence is talking about, while Verb Phrases describe

* Department of Computer Science &Engg., Birla Institute of Technology, Mesra, Jharkhand, India, *E-mails: imukherjee@bitmesra.ac.in; kirtivr.bit@gmail.com*

some action between the objects in the sentence. With regard to classification, we are interested in NPs more than VPs.

Zhao et al. [4] demonstrated that the use of word hypernyms as additional features can lead to better categorization. To solve the data sparseness problem we propose the incorporation of semantic information about terms – such as their hypernyms into the system feature set.

The paper proposed multi class classifier that divides tweets into the following classes:-

a) News b) Events c) Meme d) Sports e) Other.

Assume that the above classes cover a reasonably large portion of the topics of discussion on Twitter.

The training dataset is composed of 2,200 tweets. The training and testing corpus split the dataset in a 7:3 ratio respectively. The system ultimately compares the micro-averaged F-measure of the proposed method on the training set against the popular BOW approach, demonstrating the superiority of the proposed approach.

II. PROPOSED METHODOLOGY

2.1. Key Challenges & General Strategy

Classification of Tweets poses a number of esoteric challenges. *Firstly,* due to Twitter's 140 characters limit an average tweets comprises of 15 to 28 words. A number of these words are connecting words such as prepositions, conjunctions and other parts of speech. Consequently, the words which identify the topic discussed by the Tweets are fewer. This makes Twitter data sparse and effectively the system must carefully separate the wheat from the chaff. *Secondly,* being an informal medium, colloquial speech, abbreviated words and slang are common in Twitter. *Thirdly,* it is important that we take into account certain Twitter-specific linguistic conventions into account. For example, text following hashtags (#) are more likely to present to us the topic of discussion on Twitter. On the other hand, "@" symbols are used to initiate or reply to ongoing conversations. *Fourthly,* the proposed method uses WordNet to enhance the noun phrases present in Tweets along with other semantic information such as word hypernyms. This may cause a new problem to creep in – the *curse of dimensionality.* This can be prevent by using a scoring system (discussed ahead) and extracting only the most representative features of a Tweet.

2.2. Methodology and explanation

The topic extraction problem can be divided into:

1. *Pre-processing*: Tweet pre-processing consists of Tweet selection (English tweets consisting of 50 or more characters are considered), tokenization, spelling correction using the edit distance algorithm, stop word detection & removal using Princeton's stop word list, word stemming using Porter's algorithm and expanding slang words. Pre-processing can sometimes result in the loss of contextual information. We include information about whether edit distance is used to correct spellings or slang words have been used as features in the feature vector.
2. *Feature Selection:* The selection of features in the feature vector (to be used to train the C4.5/SVM classifier) generally follows from the definitions of the classes. We started with a large number (500+) of keywords associated with the pre-defined classes. For example, 'tournament' 'hockey' 'goal' 'striker' are some keywords associated with sports. The collection of keywords is supplemented with their "concepts" or hypernyms. For example "athlete" is a direct hypernym of striker.

   Next proceed carefully to prune the feature set. While continuing to take sports as example, the steps followed were:-

- Discriminative features are preferred. The utility of features robust to data heterogeneity and their corresponding effect on classification accuracy has been earlier explored by Pekar et al. [5]. Terms that can distinguish between the classes are preferred. For example, 'team' may not necessarily correspond to a sports team.
- Terms with more than 5 corresponding synsets are avoided as they have been empirically found to be bad features, and so are not used.
- Features are inherently binary in nature. If an interesting term is present, its corresponding feature is 1 or else 0.
- Finally, a sufficiently large feature set is considered which define the feature vector used by the C4.5/SVM classifier.

3. *Extraction of keyphrases*: WordNet POS tagging and shallow parsing (OpenNLP sentence chunkers) are used to extract noun phrase unigrams, bigrams and trigrams from the tweets. It remove terms with over 5 synsets as they have empirically found to be bad features .Trigrams are precedent over bigrams which in turn are preferred to unigrams. This means that if a bigram also forms a trigram, the trigram is preferred and the bigram containing common words are disregarded. The justification of the precedence between n-grams is based upon empirical evidence and Santini's results where trigrams (82.6%) achieved the highest classification accuracy in the genre classification domain [6].

   Noun phrases have the regular expression – (Adjective|Noun)*(Preposition|Noun|Verb)? (Noun).

   The above rule effectively means that the noun phrase ends with a noun, starts with an adjective or a noun and may have a noun/verb/preposition in the middle. Text following hashtags (#) are also extracted and processed as unigrams. They often have a strong correlation with what the tweet is discussing.

4. *Selection of the most "prevalent" concepts*

   Now a vector of multiple n-grams is formed which represent the noun phrases extracted from the Tweet. It is found that most of the terms initially found in the tweet formed trigrams, or bigrams, and a few isolated unigrams.

   The tweet is internally represented as:-

   $T = \{n_1, n_2, n_3, n_{4...}\}$ where n_i is a uni, bi or trigram.

   Any term can have more than one meaning (or technically, be a part of one or more synsets). However, it need the correct contextual meaning of the terms contained in the n-grams.

   Thus,

   $n_i = \{t_j\}$ for j from 1 to 3, where n denotes an n-gram text and *t* denotes a singular word in the n gram

   $t_j = \{s_k\}$ for k from 1 to 5 (we had earlier removed terms with over 5 synsets) where *s* is a synset for the term *t*.

   The system uses a scoring system to disambiguate the meaning of all the terms t_i. The outline of the proposed approach is as follows:-

   - Firstly, disambiguate the terms contained within the trigrams using the scoring system that discuss next.
   - then disambiguate the bigrams using the scoring system. Now create a new vector space with the selected synsets from the trigrams and bigrams.
   - Trigrams and bigrams are disambiguated using their own contexts by the system scoring system. However, isolated unigrams need to use the overall context of the tweet. The new vector space is

used containing disambiguated terms from trigrams and bigrams as the context for disambiguation for the unigrams.

Disambiguation scoring system: In this section, explain about the score calculation of each term in the tweet. A simple synset disambiguation scheme is used which seeks to find the most pervasive synset of a term based on the other terms it shares context with. This system is conceptually based on Çelik et al [7]'s proposed system to derive the correct meaning of any term which is part of a larger document.

Once the synsets of a term are identified, calculate score of every synset by calculating the similarity of the synset with all other synsets contained in the context (for trigrams it will be the two other terms contained in them). The basis of this calculation is by comparing the hypernyms of the synsets. A hypernym denotes the root or a more general concept of any term. For example: "Phone" is part of the synset "telephone" whose level 1 hypernym is "electronic equipment" and level 2 hypernym is the more general term "equipment" – so the system will use "electronic equipment" and "equipment" to denote the concept of the synset "telephone". The proposed system will use hypernyms up to 2 levels for score calculation.

The score of a synset S_i is given by:

$$Score(S_i) = \sum_{j=0, i!=j}^{k} Similarity\ (S_i, S_j) \quad (1)$$

Where Score (S_i) denotes the scores of S_i in term T, the term is k-gram.

Similarity (S_i, S_j) is defined as:-

$$Similarity\ (S_i, S_j)$$
$$= CommonCount\ (Hypernyms\ (S_i), Hypernyms\ (S_j))$$
$$+ CommonCount\ (Hypernyms\ (Hypernyms\ (S_i)), Hypernyms\ (Hypernyms\ (S_j))) \quad (2)$$

Note that Hypernyms (S_i) denotes the hypernym term list for synset S_i and CommonCount (X, Y) denotes the number of common terms in the given two lists.

After scoring phase, we select the best synset which represents a term.

5. *Populating the feature space*: - Raw n-grams and term direct hypernyms (corresponding to the selected synset) are used to populate binary features in the feature vector used to train the multi-class classifier.
6. *Classification*: - Finally, the system will use the J48 implementation of the C4.5 algorithm (which use decision trees) by Weka to train a classifier using the training set. It is also use the Weka SVM classifier as a comparative classification tool and compare the results.

III. EXPERIMENT AND RESULT

3.1 EXPERIMENTAL SETUP

The choice of the size of the training and testing data set is very important. A sufficiently large dataset of 2,200 labelled tweets are considered. These tweets were acquired from 2 sources: - the UNED [8] dataset and manual annotation of 360 sports related Tweets. Studies have shown that a core 10% of the users on Twitter contribute 90% of the tweets on twitter [9]. So we manually identify a variety of users belonging to this core group to obtain training data from them. The labelled dataset was then randomly distributed among 6 students for verification and re-annotation.

There were cases where the tweets spanned multiple classes. For example the tweet "Girls' singles winner SofyaZhuk tells @Annabel_Croft and @mwilander she "played her best tennis" in the final" is a news story about a sporting event. Since news stories cover sporting events, we label the above tweet as

"news". Similarly, there are memes which portray sports or discuss events. These tweets are labelled "memes". While it would be desirable if the classes were more discriminative, we believe that would be unrealistic for an application seeking to operate on live data. Though 2,200 tweets may seem a small number the micro-averaged F-measure with 1,100 tweets is 71.45% - a decrease of only 3.24% over the 74.69% when the classifier is trained with the entire training data.

1540 tweets (70%) of the training set are used for training while 660 tweets (30%) are used for testing. For the bag of words approach, pre-process the tweets in the same way as the proposed classification method. Use then further tf-idf based weight assignment to the rank the words contained in the tweet. The following (standard) formula is used to obtain the most discriminative words from a tweet:-

$W_{ij} = tf_{ij} \log (N/Df_i)$;

W_{ij} is the weight of a term i in the tweet j,

Tf_{ij} denotes the frequency of the term i in tweet j,

Df_{ij} denotes the number of tweets in which a term i occurs in the training set of tweets,

N is the total number of tweets.

3.2. Performance Evaluation

Assume the Bag of Words model as the system baseline since it is popularly used for text classification. Compare the accuracy, precision, recall and the F measure of the results obtained through:-

a) the POS Tagging based feature Extraction Method + C4.5.

b) Bag of Words approach + C4.5.

c) the POS Tagging based feature Extraction Method + SVM.

d) Bag of Words approach + SVM.

The results is obtained through the application of the system Part of speech based classification method for all the classes (using the C4.5 decision tree based classifier). The testing set consists of 192 tweets which belonged to the class "News". The confusion matrix for "News" after testing looked like the below:-

Table 1
Testing results for News using the POS method

	Predicted Negative	*Predicted Positive*
Negative Cases	TN: 401	FP: 67
Positive Cases	FN: 24	TP: 168

TN – True negative; TP – True positive; FN – False negative; FP – False positive

Accuracy = (401 + 168)/660= 0.8621(or 86.21%) Precision = 168/ (168+67) = 0.7636 (or 76.36%)

Recall = 168/192 = 0.875 (or 87.5%)

Macro averaged F-score => 2 * (precision * recall) / (precision +recall) = 0.8155

We had 102 tweets in the testing set which belonged to the class "Meme".

The confusion matrix for "Meme" after testing looked like the below:-

Table 2
Testing results for Memes using the POS method

	Predicted Negative	*Predicted Positive*
Negative Cases	TN: 525	FP: 33
Positive Cases	FN: 15	TP: 87
Accuracy = (525 + 87)/660 = 0.9272	Precision = 87/ (87+33) = 0.725	
Recall = 87/102 = 0.8529	F-score = 0.7838	

The 124 tweets in the testing set which belonged to the class “Events”. The confusion matrix for “Events” after testing looked like the below:-

Table 3
Testing results for Events using the POS method

	Predicted Negative	*Predicted Positive*
Negative Cases	TN: 524	FP: 12
Positive Cases	FN: 42	TP: 82
Accuracy = (524+82)/660 = 0.9181	Precision = 82/ (82+12) = 0.8723	
Recall = 82/124 = 0.6612	F-score = 0.7522	

The 118 tweets in the testing set which belonged to the class “Sports”. The confusion matrix for “Sports” after testing looked like the below:-

Table 4
Testing results for Sports using the POS method

	Predicted Negative	*Predicted Positive*
Negative Cases	TN: 521	FP: 21
Positive Cases	**FN: 22**	**TP: 96**
Accuracy = (521+96) / 660 = 0.9348	Precision = 96/ (96+21) = 0.8205	
Recall = 96/118 = 0.8136	F-score = 0.8170	

The “Other” class had 124 labelled tweets in its training set. These consisted of tweets which did not belong to any of the above categories. The purpose of the “other” class (apart from being the default bucket) was also to act as a sanity check for the classifier, to detect over or under fitting. The confusion matrix for “Other” after testing looked like the below:-

Table 5
Testing results for “Others” using the POS method

	Predicted Negative	*Predicted Positive*
Negative Cases	TN: 504	FP: 34
Positive Cases	FN: 64	TP: 60
Accuracy = (504+60) / 660 = 0.8545	Precision = 60/ (60+34) = 0.6382	
Recall = 60/124 = 0.4838	F-score = 0.5504	

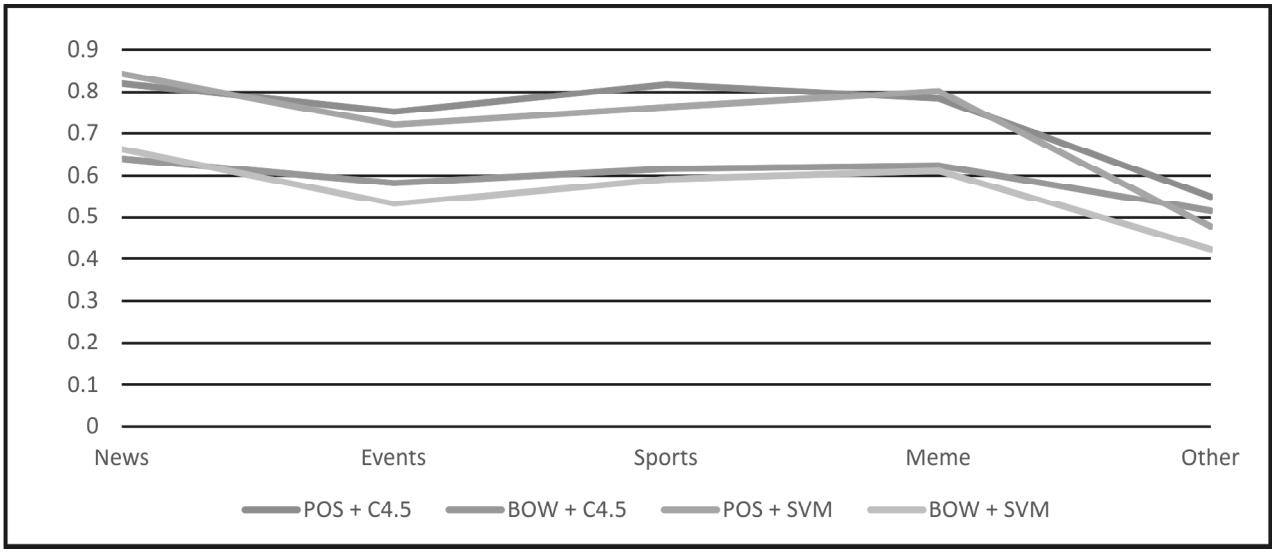

Figure 1: F-score comparison over categories

Next, obtain the overall micro F-measure for the classifier. The F measure is computed over all system classes. Precision and recall are obtained by summing over all the class-decisions.

$$\text{Overall Precision} = \frac{\Sigma_{i=1}^{M} TP_i}{\Sigma_{i=1}^{M} TP_i + FP}$$

= (168+87+82+96+60) / ((168+67) + (87+33) + (82+12) + (96+21) + (60+34)) = 493/660 = 0.7469

$$\text{Overall Recall} = \frac{\Sigma_{i=1}^{M} TP_i}{\Sigma_{i=1}^{M} TP_i + FN_i} = 493/660 = 0.7469$$

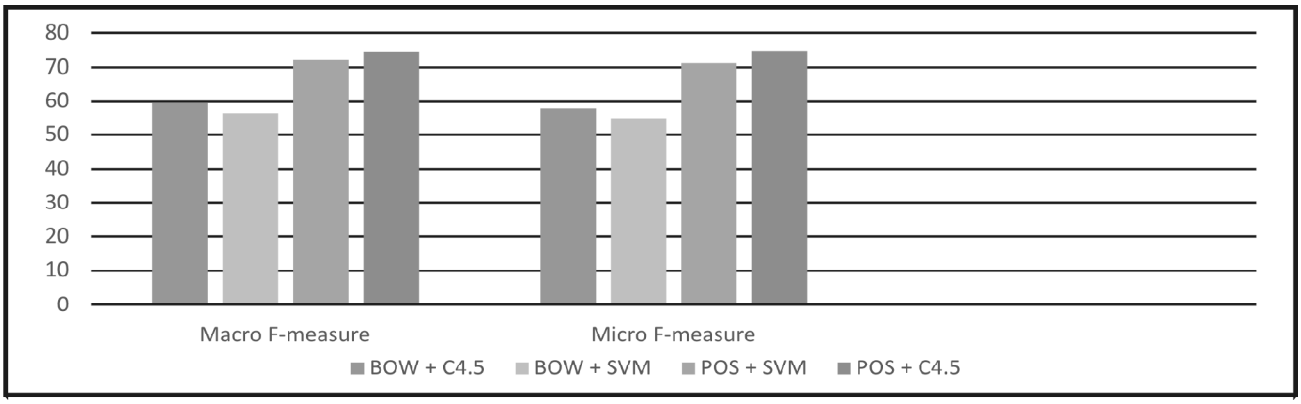

Figure 2: Contrasting the performance statistics of the BOW approach to the semantic approach based on Part of speech tags

Micro F-score = 2 * (precision * recall) / (precision +recall) = 0.7469.

$$\text{Macro F-score} = \frac{\Sigma_{i=1}^{M} F}{M} = (0.8155 + 0.7838 + 0.7522 + 0.8170 + 0.5504) / 5 = 0.74378$$

The best overall classification results were obtained using the POS-based approach in combination with the decision tree based C4.5 classifier. Taking the micro averaged F-measure as a performance measurement parameter, the proposed approach was a 29% improvement over the widely used tf-idf based Bag-of-words approach for text classification.

IV. CONCLUSION

Twitter first caught the imagination of data scientists as a real time platform for gaining insight to information shared by Twitter users across the world. Classification of tweets in particular has allowed us to use Twitter as a tool for sentiment analysis [10], in critical applications like disaster management [11], and even as a news source often swifter than traditional news aggregators. Such work has even led to the evolution of Twitter as a platform over time [12].

Traditional text classifiers are blind to the semantic information expressed in their subject text. While bag of words based approaches have effectively been used widely, the data points they draw inferences from – word frequencies- are ill-suited to an unstructured sparse data source such as Twitter.

The primary contribution through this paper is a part of speech based multi-class microblog classifier which has been designed from the ground up with language semantics and data sparseness in mind. We have worked exclusively with Twitter but it is believe that the basic approach can be applied to address other linguistic needs as they continue to arise in the era of social media and rapidly changing linguistic conventions.

It is also present a comparison of the Weka C4.5 & SVM implementations for the proposed technique. It is notice that there is no major difference between the two, though SVM performed better for News and Meme classes which have a larger associated feature set compared to other classes.

There is further scope for improvement in the proposed methodology.

Here are a few issues have been identified and plan to fix in the future:-

- In the online scenario, when to re-train the classifier model is an important question we need to answer. Re-training for every incoming tweet is inefficient. Alternately, we could consider re-training when a “Concept Drift” [13] is detected in the incoming data.
- Since the proposed system currently do not crawl tiny URLs which are sometimes appended to tweets. These URLs link to images, videos, or blogs. While crawling these URLs to obtain meaningful information may be expensive in the online scenario, it would allow us to better classify tweets containing tiny URLs.

REFERENCES

[1] Gimpel, Kevin, et al., “Part-of-speech tagging for twitter: Annotation, features, and experiments”, Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2. Association for Computational Linguistics, 2011.

[2] Sriram, Bharath, et al., “Short text classification in twitter to improve information filtering”, Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval ACM, 2010.

[3] Sankara Narayanan, Jagan, et al. “TwitterStand: news in tweets”, Proceedings of the 17th acmsig spatial international conference on advances in geographic information systems ACM, 2009.

[4] Li, J., Y. Zhao and B. Liu, “Fully Automatic Text Categorization by Exploiting WordNet”, Proceedings of the 5th Asia Information Retrieval Symposium on Information Retrieval Technology, Heidelberg, 2009.

[5] Pekar, V. et al., “Selecting Classification Features For Detection of Mass Emergency Events on Social Media”, Proceedings of the International Conference on Security and Management (SAM) and The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2016.

[6] Santini, Marina, “A shallow approach to syntactic feature extraction for genre classification”, Proceedings of the 7th Annual Colloquium for the UK Special Interest Group for Computational Linguistics. Birmingham, UK, 2004.

[7] Celik, Koray, and TungaGungor, “A comprehensive analysis of using semantic information in text categorization”, Innovations in Intelligent Systems and Applications (INISTA), 2013 IEEE International Symposium on IEEE, 2013.

[8] A. Zubiaga, D. Spina, V. Fresno, R. Martínez, “Real-Time Classification of Twitter Trends”, Journal of the American Society for Information Science and Technology (JASIST), 2014.

[9] B. Heil and M. Piskorski, “New Twitter research: Men follow men and nobody tweets”, http://blogs.harvardbusiness.org/cs/2009/06/new_twitter_research_men_follo.html Pub. June 1, 2009.

[10] Kouloumpis, Efthymios, Theresa Wilson, and Johanna D. Moore. “Twitter sentiment analysis: The good the bad and the omg!”, Icwsm 11 (2011): 538-541, 2011.

[11] Toriumi, Fujio, and Seigo Baba. “Real-time Tweet Classification In Disaster Situation”, Proceedings of the 25th International Conference Companion on World Wide Web and International World Wide Web Conferences Steering Committee, 2016.

[12] Hussain, Muhammad IrshadAlam, “Evaluation of graph centrality measures for tweet classification”, Proceedings of the International Conference on Computing, Electronic and Electrical Engineering (ICE Cube), 2016.

[13] Joao Gama, Indre Zliobaite, Albert Bifet, Mykola Pechenizkiy and Abdelhamid Bouchahia, “A Survey on Concept Drift Adaptation”, ACM Computing Surveys, Vol. 1, No. 1, Article 1, January 2013.

[14] Castillo, C. ; Mendoza, M. , and Poblete, B., “ Information credibility on twitter”, In Proceedings of the 20th International Conference on World Wide Web, WWW ’11, pages 675–684, New York, NY, USA, ACM, 2011.

[15] Kairam, S. R., Morris, M. R., Teevan, J. Liebling, D. , and Dumais, S., “Towards supporting search over trending events with social media”, In Proceedings of ICWSM 2013, the 7th International AAAI Conference on Weblogs and Social Media, 2013.