A Topical Crawler for Uncovering Hidden Communities of Extremist Micro-Bloggers on Tumblr

Swati Agarwal
Indraprastha Institute of Information Technology,
Delhi (IIIT-D), India
swatia@iiitd.ac.in

Ashish Sureka
Software Analytics Research Lab
(SARL), India
ashish@iiitd.ac.in

ABSTRACT

Research shows that microblogging websites such as Tumblr are being misused as a platform to disseminate hate and extremism. We formulate the problem of locating such extremist communities as a graph search problem. We propose a topical crawler based approach performing several tasks: searching for a blogger, computing its similarity against exemplary documents, filtering hate promoting bloggers, navigating through links to other bloggers and managing a queue of such bloggers for social network analysis. We conduct experiments on real world dataset and examine the effectiveness of 'like' and 'reblog' features as links between bloggers. Experimental results demonstrates that the proposed solution approach is effective with an F-score of 0.80.

Keywords

Mining User Generated Content, Online Radicalization, Social Media Analytics

1. PROBLEM DEFINITION & SOLUTION

Tumblr is a popular and widely-used micro-blogging website. Previous research shows that such websites are used as a platform for disseminating hate and extremism (due to low barrier to publication and anonymity) [1][2][3][4][5]. Automatic identification of hate and extremism promoting posts and bloggers is an important (from the perspective of the website moderators and law enforcement agencies) and a technically challenging problem. Large volume of data on Tumblr, free-form text and noisy content makes automated analysis technically challenging [1][2][3][4][5]. Our aim is to investigate the application of a topical crawling based algorithm for retrieving hate promoting bloggers on Tumblr. Our objective is to examine the effectiveness of a random-walk based approach in social network graph traversal. Furthermore, our goal is to examine the effectiveness of *re-blogging* and *like* on a post as the links between two bloggers and conduct experiments on large real world dataset to demonstrate the effectiveness of our approach.

Published as part of the #Microposts2015 Workshop proceedings, available online as CEUR Vol-1395 (http://ceur-ws.org/Vol-1395)

#Microposts2015, May 18th, 2015, Florence, Italy.

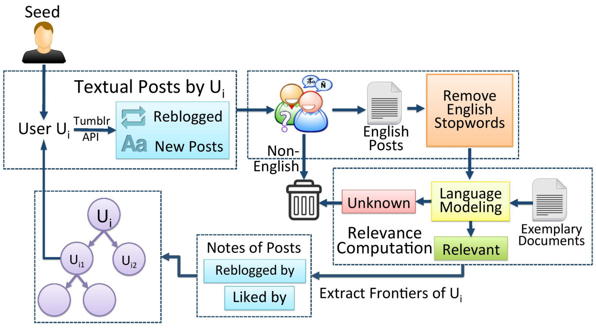

Figure 1: Proposed Architecture for Extremist Community Detection

In a graph traversal, a topical crawler returns relevant nodes to a specific topic. To define the relevance of a node, it learns the characteristics and features of given topic and computes the extent of similarity against a bunch of exemplary documents. To collect training examples, we perform an iterative search on Tumblr using keyword based flagging, where keyword is a search tag; for example, jihad, anti-Islam and hate. We perform a case study on Jihad and by manual search on Tumblr posts we collect several relevant tags that are commonly used by extremist bloggers. We use these tags to initiate our process and collect all textual posts (avoiding picture, audio, video and URLs), tags (associated with resultant posts) and linked bloggers (post reblogged by and liked by) with no redundancy. We perform a manual inspection on resultant posts and posts made by linked bloggers to filter relevant (hate promoting) and unknown results. We further extract more posts and linked bloggers from related tags and run this framework recursively to collect our exemplary documents (400 hate promoting posts). These training examples contain the body and caption of only positive class (hate and extremism promoting content) posts which is used to train the model.

Figure 1 illustrates the design and architecture of topical crawler to locate extremist communities. As shown in Figure 1, our proposed solution framework is an iterative multi-step process primarily consisting of five phases: features (posts) extraction, data pre-processing, classification, frontier extraction and graph traversal. In phase 1, we initiate our process using a positive class (hate promoting) blogger U_i called as 'seed'. We use Tumblr API [1] to fetch the URLs of n number of textual posts and by using Jsoup Java library [2] we extract the content and caption of these posts (used as contextual metadata). These posts can be either re-blogged from other users or originally posted by the user U_i. These posts consist of multiple langauges. Therefore, in phase 2, we perform data pre-processing and filter English and non-English posts using language detection library[3]. We perform data pre-processing on these posts and remove English stopwords. In phase 3, we build a statistical model from the exemplary documents collected separately by semi-automatic process. To compute the relevance of each blogger, we use character level n-gram language modeling approach. We find the extent of similarity between metadata and exemplary documents using LingPipe API [4] - applying joint probability-based classification of character sequences . We implement a one class classifier and filter extremism promoting bloggers from unknown bloggers. In phase 4, we extract the notes associated with the posts (collected in phase 1) of relevant bloggers. These notes contain the list of bloggers who liked and re-blogged a particular post. The number of notes represent the popularity of a post and indicate the similar interest between original poster and other bloggers in the list who may or may not be the direct followers of each other. We use notes to extract frontier nodes of a blogger because of two reasons: 1) due to the privacy policies Tumblr API does not allow developers to extract followers and following blogs of Tumblr users. 2) Tumblr facilitates bloggers to track any number of tags so that whenever there is a new post published publicly on Tumblr containing any of these tags, it automatically appears in a menu on user's dashboard. They can spread that post among their followers by re-blogging it. Tracked tags allow bloggers to form a virtual community without following each other. For each frontier extracted in phase 4, we compute the relevance score against exemplary documents and discard unknown bloggers. In phase 5, we manage a queue of relevant bloggers and perform directed graph traversal using random walk algorithm. To expand our graph we select the next blogger in uniform distribution and extract it's frontiers. We execute our focused crawler for each frontier without revisiting a blogger. This traversal results in a connected graph, where nodes represents a blogger (hate promoting) and edges represent the links (re-blog and like) between two bloggers. We perform social network analysis on the resultant graph and locate extreme right communities of hate promoting bloggers.

2. RESULTS & CONCLUSION

We execute our topical crawler for a given seed blogger and traverse through Tumblr network using random walk algorithm. For every new blogger, we compute its relevance and classify it as hate promoting or unknown using one class classifier. To examine the effectiveness of our classifier, we compute its accuracy using standard information retrieval techniques. In one execution of our topical crawler, we were able to collect 600 bloggers. We hired 30 graduate students as volunteers from different department to label these bloggers as hate promoting or unknown according to their published posts and given guidelines for annotation. To avoid the biasness and to collect correct annotated results we perform a horizontal and vertical partition on nodes and arrange these 600 bloggers into a 2D matrix where rows are the numbers of annotators grouped in 10 sets, 3 members each. Columns of the matrix are the number of bloggers assigned to each member for annotation i.e. 60. We use majority voting approach for final annotation, the class of a blogger is the one which is voted by at least two annotators. Based upon the validation results we evaluate the accuracy of our model. Table 1(a) shows the confusion matrix for one class classification. Table 1(a) reveals that our model predicts 382 (290+92) bloggers as hate promoting and 218 (173+45) bloggers as unknown. Table 1(a) shows that there is a missclassification of 13% and 34% in predicting hate promoting and unknown bloggers. Table 1(b) shows the accuracy results of our classifier. Results shows that the precision, recall and f-score are reasonably high and we are able to predict hate promoting bloggers with an accuracy of 77%. Our experimental analysis reveals that re-blogging is a good indicator of connection between two bloggers. We locate users who are central and influential among all and play major role in the discovered communities. We perform independent social network analysis on like and re-blog links among bloggers and conclude that re-blogging is a discriminatory feature to identify the communities of extremist bloggers sharing a common agenda.

Table 1: Confusion Matrix and Accuracy Results for One Class Classifier

(a) Confusion Matrix

		Predicted	
		Positive	**Unknown**
Actual	**Positive**	290	45
	Unknown	92	173

(b) Accuracy Results

Precision	**Recall**	**F-Score**	**Accuracy**
0.75	0.86	0.80	0.77

3. REFERENCES

[1] S. Agarwal and A. Sureka. Using knn and svm based one-class classifier for detecting online radicalization on twitter. In *Distributed Computing and Internet Technology (ICDCIT)*, pages 431–442, 2015.

[2] E. A. Cano Basave, Y. He, K. Liu, and J. Zhao. A weakly supervised bayesian model for violence detection in social media. In *Sixth International Joint Conference on Natural Language Processing*, pages 109–117, 2013.

[3] S. Kumar, F. Morstatter, R. Zafarani, and H. Liu. Whom should i follow?: Identifying relevant users during crises. In *ACM Conference on Hypertext and Social Media (HT)*, pages 139–147, 2013.

[4] A. Sureka and S. Agarwal. Learning to classify hate and extremism promoting tweets. In *Joint Conference in Intelligence Security Informatics (JISIC)*, pages 320–320. IEEE, 2014.

[5] J. Xu, T.-C. Lu, R. Compton, and D. Allen. Civil unrest prediction: A tumblr-based exploration. In *Social Computing, Behavioral-Cultural Modeling and Prediction*, pages 403–411. Springer, 2014.

[1] https://www.tumblr.com/docs/en/api/v2

[2] http://jsoup.org/apidocs/

[3] https://code.google.com/p/language-detection/

[4] http://alias-i.com/lingpipe/index.html