ConfTracker: Imminent Event Extraction from Twitter in Closed Domain

Ravi Shankar
Arizona State University
Computer Sci & Engineering
ravs@asu.edu

Anil Kuncham
Arizona State University
Computer Sci & Engineering
akuncham@asu.edu

Abstract

Twitter has become a major platform for publicizing any happening or forthcoming events. Technical events, Comic cons, meet ups , Hackathon and various events of diverse genre promote their event by the means of tweets and hash-tags[1]. Previous work on extracting events from social media feeds has been mainly on either news text or open domain events (Ritter et. al). We present in this paper that extracting closed domain event data from Twitter is feasible and can be achieved with fairly good results using the hashtags and acronyms present in the tweet. We present two models: SVM (using unlabeled tweets) and Naive Bayes Classifier(using labeled tweets) and show that extraction of events in closed domain can be achieved.

Related Work

"Previous work in event extraction has focused largely on news articles, as historically this genre of text has been the best source of information on current events" as stated by A. Ritter et al[1]. Although their model achieves 90% accuracy it still cannot be used to extract events in closed domain. Simple classifiers like SVM[2] and Naive Bayes Classifier, unlike the complex techniques used in [1], can provide satisfactory results when operated on closed domain event extraction. We will show that Support Vector Machine with no-POS documents can still produce better results than Naive Bayes Classifier with annotated documents.

[1] https://blog.twitter.com/2013/your-pass-to-comic-con

Preparing the Data

In order to get tweets related to conferences, we prepared a Twitter list[2] of Twitter handles of such conferences. From the timeline of the list we extracted the tweet corpus, which served as our basic corpus for training and testing the models.

Since we worked on two different models: SVM and NBC, both required training data in different format. For our SVM model, we manually labeled each tweet (~800) based on the content whether the tweet contains any information about imminent conference or such event. While for our second model, NBC, we manually annotated our tweet corpus. Since it would have been tiring task, we adopted augmented manual tagging: we started with annotating first 100 tweets, then use the tags to annotate next 100 tweets and then perform manual annotation on this set. We followed this iteration until our corpus was completely annotated. We used our own POS tags which is inspired from PennTreeBank[3].

Model

We used two models to demonstrate that extracting events in closed domain from Twitter stream is feasible and can give satisfactory results.

First model we used is SVM(Support Vector Machine)[2]. We started with manually labeling each tweet (+1,-1) depending on the content of the tweet. We made our decision based on presence of 4-tuples in tweet: 1. *Conference entity*, name of the conference, 2. *Temporal data*, since we were doing it manually format did not matter in this model, 3. *Geographical data*, location of the event, and last optional feature 4. *HyperLink or URL*, link to the site of conference. We marked the tweet with +1 label if the tweet had 1 and 2 and (3 or 4).

We used SVMLight[3] as our tool for SVM model. We prepared the training and test data as per the format of SVMLight's specifications[4]. Our train and test data consist of ~400 labeled tweets in each set. Figure 1 shows the flow of the process which was carried out while using

[2] https://support.twitter.com/articles/76460-using-twitter-lists

[3] http://www.cis.upenn.edu/~treebank/

[4] www.cs.uic.edu/~liub/teach/cs583-spring-05/SVMLight.ppt

SVMLight as the learning and classification module. As presented in Table 1, with training and test corpus with just the proper labeling we achieved ~78% of accuracy with SVM model.

Second model we used is Naive Bayes Classifier, we implemented Supervised Classification model in this phase. The classifier is built on training corpora containing the correct label for each input. We analyzed number of tweets relevant to conferences and understood the usage pattern. Based on this knowledge we decided on feature vectors which represent a category of unique words. We then manually annotated the tweets and prepared corresponding corpus to each feature vector.The content of the tweet is very less and often people use different short hand notations to express their point. Also, people tend to be succinct in expressing their intent of the tweet. This makes it tough to analyze a single tweet and get some meaning out of it. The temporal information that has to be extracted is in different formats and most of the time it is relative to the time of the post. So all such scenarios have to be handled.

The feature vectors we selected narrow the analysis of tweet to closed domain. For example the occurrence of word "conf or event" or words like "call for papers" etc partly convey some meaning relevant to conferences.

The feature vectors used are :

NNCONF = Nouns like conference, workshop etc.

VBCONF = Verbs like attending, register etc.

HTCONF = Hashtags which represent a user for a particular conference”#jsconf"

USRCONF = Handle for a particular conference “@RubyConf"

POSITIVE = This feature vector is added based on experiment results to amplify the weight of the relevant tweet.

NEGATIVE = This feature vector is also added based on experiment results to amplify the occurrence of a past event or feedback related to happened event.

URL = We used regular expressions to extract the URL. If url is present, then the tweet is much more informative.

TEMPORAL = We used a lot of pattern recognition and came up with regular expressions which would match the tweeting style of different users.

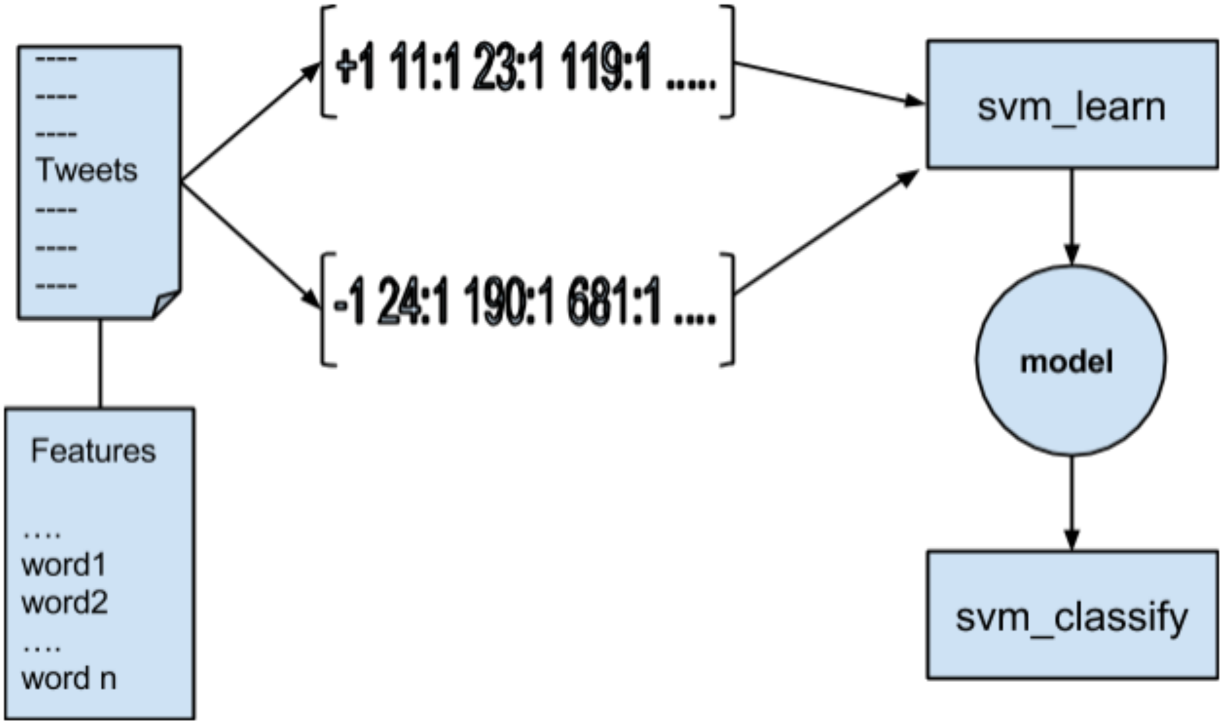

Figure 1: Processing tweet corpus into format acceptable by SVMLight and then passing it to svm_learn which creates the model for the classifier.

Table 1: Result using SVMLight

	Train Data (XiAlpha-Estimates)	Test Data
Precision	**>= 52.68%**	**92.75%**
Recal	**>= 46.83%**	**42.67%**
Accuracy	**>= 70.00%**	**78.54%**

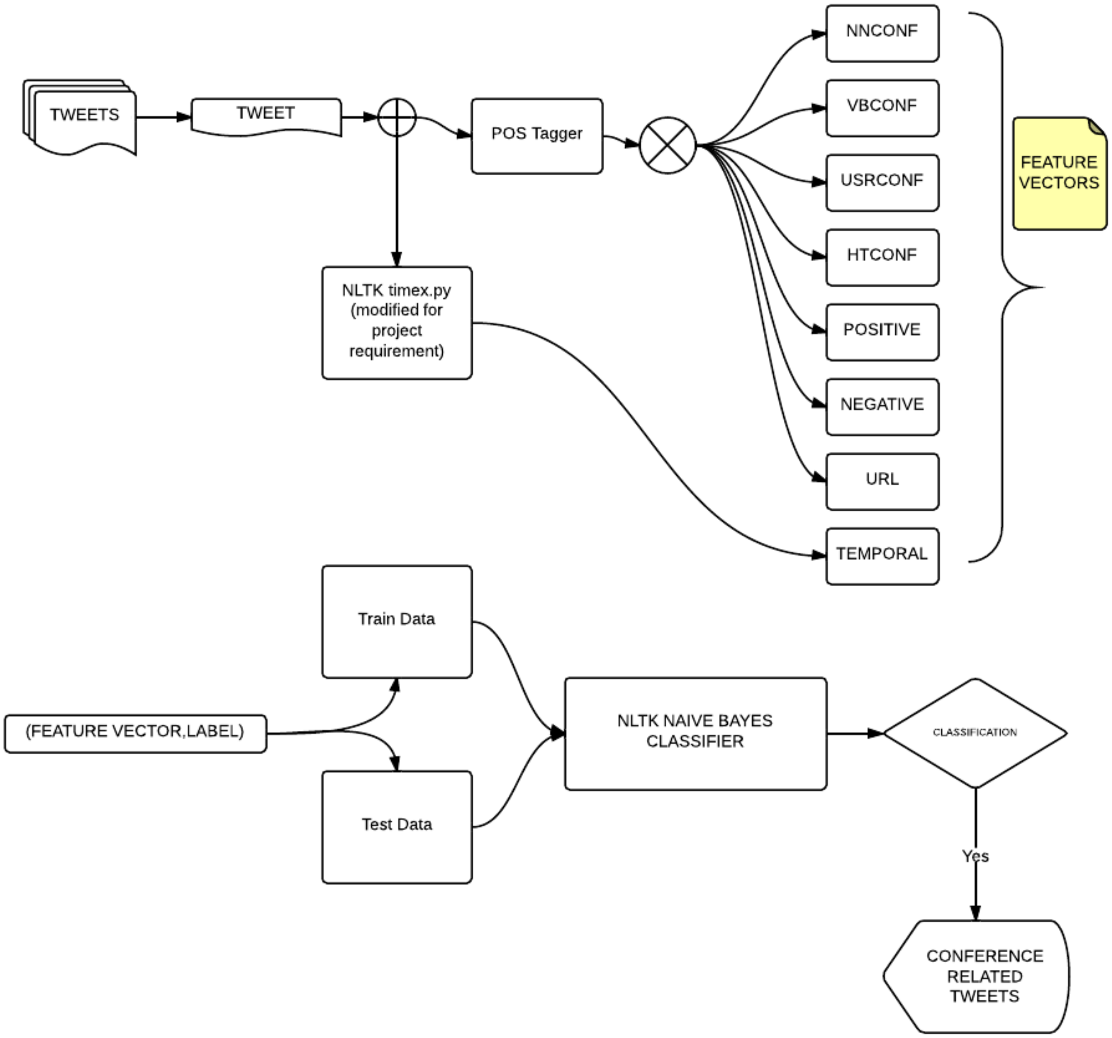

Figure 2: Process flow using NBC

Table 2: Result using NBC(NLTK Python Impl)

	Training Set(# of instances)	Test Set(# of instances)	Accuracy
1	200	200	34.5%
2	200	400	34.75%
3	400	400	41.75%
4	600	200	42%

Conclusion and Future Work

We presented the methods which can be used to learn and classify the tweets based on the Conference entity, geographic data, temporal data, and/or external links. Once we have classified the tweet, we can apply POS tagger to extract Named entity, geographic data and temporal data, and present the information to the user. We demonstrated the results using two well know models: SVMs and NBC. We achieved ~78% of accuracy with SVM which is far superior than NBC in our experiment.

We can use the methods being used by Ritter et al[1] to achieve 90% accuracy. We can further apply sentiment analysis method to rate the events and present the rating to our users. We can also implement methods to find trending events in local and global geographies and recommend events based on user's interest. There is a lot of space for idea in ConfTracker and give right amount of time and enthusiasm we will implement all or most of them and make it live at http://www.conftracker.com

Reference

[1] A. Ritter, Mausam, O. Etzioni, S. Clark. Open Domain Event Extraction from Twitter. In KDD'12, August 12–16, 2012, Beijing, China.

[2] B. E. Boser, I. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. In Proceedings of the Fifth Annual Workshop on Computational Learning Theory, pages 144-152. ACM Press, 1992.

[3] T. Joachims, Making large-Scale SVM Learning Practical. Advances in Kernel Methods - Support Vector Learning, B. Schölkopf and C. Burges and A. Smola (ed.), MIT-Press, 1999.