Classifying Typhoon Related Tweets

Alron Jan Lam, Ivan Paner, Jules Matthew Macatangay, Duke Danielle Delos Santos
College of Computer Studies, De La Salle University
2401 Taft Ave, Manila City
1004 Malate, Metro Manila, Philippines
+63 2 524 4611
{alron_lam, ivan_paner, jules_macatangay, duke_delossantos}@dlsu.ph

ABSTRACT

Organizations concerned with relief operations during disasters have turned to social media such as Twitter to look for valuable information. However, current processes for sorting through tweets mostly involve manual methods which overwhelm human resources. In these cases, automatic classifiers would be helpful. In this paper, we classified typhoon related tweets as: resource coordination, urgent rescue needed, urgent rescue resolution, damage reporting, missing people, and media storm coverage. These tweets, totaling 2,356, were represented using Bag of Words with the Term Frequency – Inverse Document Frequency (TF-IDF) weighting scheme, and were classified using Support Vector Machine (SVM) and Naïve Bayes classification. As test bed, we focused on Philippine typhoon related tweets because the Philippines is hit by at least 20 typhoons per year resulting in USD 35M worth of property loss per storm. Ten-fold cross-validation was used to evaluate the classifiers. Results show that the SVM classifier performed better with an F-score of 88.7% and a kappa statistic of 81.7% than the Naïve Bayes classifier with 77.3% and 62.6% respectively. Future work may involve using binary classifiers for each category, which would allow irrelevant tweets to be uncategorized and tweets containing multiple types of information to be classified into multiple categories.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications – *data mining, statistical databases.*

General Terms

Measurement, Performance, Reliability, Human Factors.

Keywords

Naïve Bayes, Philippines, SVM, Twitter, Typhoon.

1. INTRODUCTION

The Philippines is hit by an average of 20 typhoons annually. Some of these typhoons cause much damage. Disaster statistics from 1980 to 2010 point to an average of 144 people killed per storm, with an additional 446,210 lives affected. Economic damages are estimated at USD 31.5M or PHP 1.35B per storm[1]. During calamities, Filipinos air their grievances via social meida. As of August 2012, the Philippines has had 9.7 million Twitter users despite the fact that only 30 percent of Filipinos have Internet access[2]. This widespread use of Twitter by Filipinos could potentially provide involved agencies with relevant information during a typhoon.

This is true not only locally but also globally. Organizations have seen the importance of being able to quickly classify disaster-related tweets for appropriate dissemination. The Standby Volunteer Task Force (SBTF) under the Digital Humanitarian Network (DHN) in cooperation with the UN Office for the Coordination of Humanitarian Affairs (OCHA) has been categorizing tweets manually using crowdsourcing tools such as MicroMappers Volunteers are given tweets and they classify them manually into categories. However, the copious amount of data overwhelmed the volunteers. To solve this, techniques had to be developed to handle this big data [2].

There have been several efforts in this field of research, such as a classifier for Haiti earthquake-related text messages [1], and classifiers for disasters like Hurricane Sandy and Typhoon Bopha in 2012 [6]. However, it has been noted that despite the promising results of the classifiers, they are highly dependent on the type of disaster and the country in which the disaster is happening. Thus, it would be beneficial to have such classifiers for Philippine typhoon related tweets.

2. RELATED WORKS

Naïve Bayes classifiers were used for classifying disaster-related tweets [4, 10]. Naïve Bayes classifiers performed better than Support Vector Machines (SVM) [4], and have been known to perform well for the task of text classification [10]. Table 1 shows the performance of the system developed by Imran et al. [4].

Table 1. Naïve Bayes performance for the Imran et al. system

Class	Precision	Recall	F-score
Caution	0.859	0.765	0.809
Donation	0.726	0.716	0.721
Casualty	0.526	0.652	0.583
Information source	0.545	0.581	0.562

[1] From "EM-DAT: The OFDA/CRED" by Université catholique de Louvain, 2013, *International Disaster Database*.

[2] From "Philippines has 9.5M Twitter users, ranks 10th," by P. Montecillo, August 9, 2012, *Philippine Daily Inquirer*.

Proceedings of the 10th National Natural Language Processing Research Symposium, pages 32–35
De La Salle University, Manila, February 21-22, 2014.

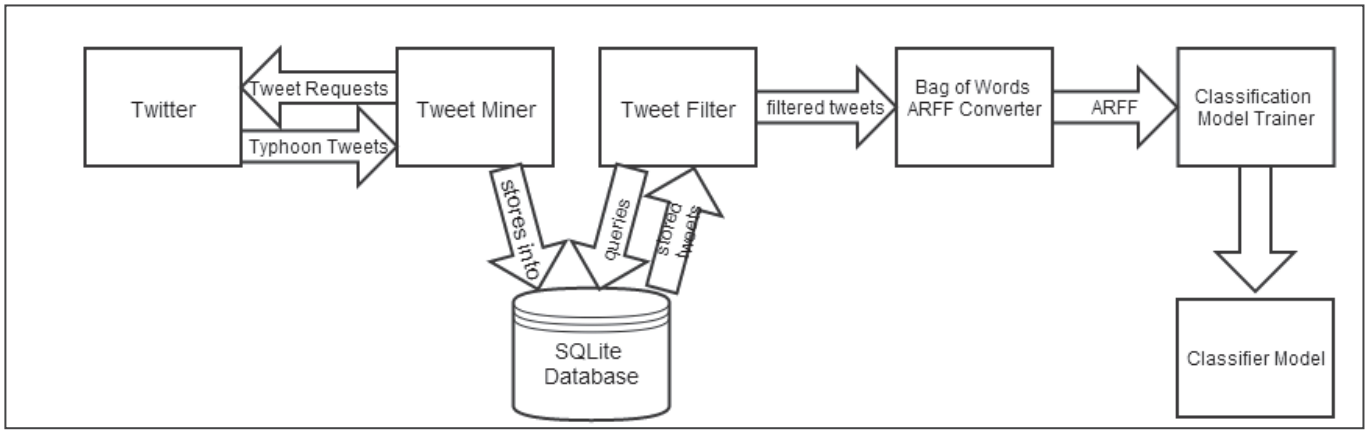

Figure 1. The system architecture

SVMs are among the most effective algorithms in solving many classification problems [1]. This is why they it was used in the development of a text classification system for Haiti. The feature selection approaches used for this system are the Bag of Words approach, the Relief Feature Selection Method, Feature Abstraction, and the Latent Dirichlet Allocation method [1]. The results of the study, shown in Table 2, reveal that extracting features through the Bag of Words (BoW) and Feature Abstraction (FA) approaches yield the highest F-scores.

Table 2. F-scores for the Caragea et al. SVM system

Class	BoW (%)	FA (%)
Medical emergency	29 ± 6	27 ± 8
People trapped	68 ± 11	74 ± 9
Food shortage	71 ± 2	73 ± 3
Water shortage	66 ± 3	67 ± 2
Water sanitation	91 ± 1	94 ± 1
Shelter needed	52 ± 2	52 ± 5
Collapsed structure	42 ± 8	33 ± 15
Food distribution	27 ± 5	27 ± 3
Hospital / clinic services	56 ± 4	59 ± 6
Person news	55 ± 6	59 ± 4

The use of the Term Frequency – Inverse Document Frequency (TF-IDF) weighting scheme can improve text retrieval and data mining systems [8]. In this case, TF-IDF can potentially improve classifier performance by giving appropriate weights to the terms in the Bag of Words.

3. CLASSIFICATION SYSTEM

Naïve Bayes classifiers were used for classifying disaster-related tweets [4, 10]. Naïve Bayes classifiers performed better than Support Vector Machines (SVM) [4], and have been known to perform well for the task of text classification [10]. Table 1 shows the performance of the system developed by Imran et al. [4]. shows the system architecture. The Tweet Miner queries Twitter for relevant tweets which are then stored in a local database for later manipulation. The Tweet Filter cleans up and manages stored Tweets to be used by the converter. The converter then manipulates the filtered Tweets into an ARFF file, a format accessible by Weka[3]. The ARFF files are then fed to Weka for training the classifiers. Once the classifiers are produced, their models are then saved for future use in either Weka or as a part of a bigger system.

3.1 Tweet Miner

The Tweet Miner is responsible for requesting tweets from Twitter. It was implemented using the Twitter4J 3.0.5 library[4]. It requests tweets that had the six official Philippine typhoon hashtags[5], representing the six categories used in the system:

- #RescuePH for *resource coordination*
- #ReliefPH for *urgent rescue needed*
- #SafeNow for *urgent rescue resolution*
- #FloodPH for *damage reporting*
- #TracingPH for *missing people,* and
- #YolandaPH for *media storm coverage*.

Hashtags are words prefixed with a hash symbol and are used as metadata in searching tweets. These six hashtags represent the categories to be used, and are the official hashtags set by the Philippine government during the relief efforts for Typhoon Haiyan. The tweets were all gathered within three weeks after typhoon Haiyan struck the Philippines on November 8, 2013. Obtained tweets were then inserted into an SQLite database. The Tweet Miner was implemented using Java.

The final dataset includes a total of 2,356 filtered tweets. However, the dataset was very imbalanced, with tweets for media storm coverage and urgent rescue needed already accounting for 2,019 of the tweets. The next highest was for missing people with 215 instances, then resource coordination with 105 instances. Lastly, urgent rescue resolution and damage reporting had the least instances, with ten and six tweets respectively. Despite repeated attempts to retrieve more tweets in these categories, the

[3] Weka is a suite of machine learning algorithms written in Java. http://www.cs.waikato.ac.nz/ml/weka/

[4] Twitter4J is a Java library for the Twitter API. http://twitter4j.org/en/index.html

[5] From "#ReliefPH | Twitter consolidates hashtags for typhoon yolanda updates, efforts," by J. Garcia, 2013, *InterAksyon*.

numbers remained low due to people not using these hashtags frequently.

3.2 SQLite Database

The SQLite database is responsible for storing the tweets and ensuring data integrity. The database was authored using the SQLite 3 format via SQLite Studio 2.1.4[6]. Values stored are Twitter ID, username, text, date, longitude, and latitude. It is strongly assumed that each tweet has a unique ID which is thus used to check for duplicates. The database automatically rejects the insertion of duplicate tweets based on the Twitter ID. Tweet duplicates need to be accounted for because Twitter4J usually returns the same set of tweets if there are no new tweets having the desired hashtag. This usually happens if the miner program is ran in succession with short time intervals in between.

3.3 Tweet Filter

The Tweet Filter is responsible for managing access to the tweets stored in the database. It provides functions for accessing, cleaning, and checking the category of the tweets. The Tweet Filter was implemented using Java.

Tweets containing more than one official hashtag were likely general announcements about how to use these hashtags. An example would be "Please use: #ReliefPH for Resource Coordination, #SafeNow Resolves #RescuePH, #FloodPH Damage Reporting, #TracingPH Report missing people". While such tweets are important for the general public, this will not be of help to government agencies. Alternatively, such hashtags may have been used inappropriately, or the tweet belonged to multiple categories. One real example of inappropriate use is, "#PrayForThePhilippines #RescuePH #PilipinoYanEh #Philippines #PrayForThePhilippines #RescuePH #PilipinoYanEh #Philippines". Since it would be best to train classifiers with tweets that represent their respective categories clearly, the group decided not to include tweets with multiple official hashtags.

Presence of one of the six hashtags is searched in the tweets' texts in order to properly label them. Some tweets, especially retweets, are truncated by Twitter in order to fit the 140 character limit. Due to this, several tweets had partial hashtags in them. The group decided to filter out these tweets that do not contain a full, official hashtag. Also, all tweets were converted to lower case, accomplishing two goals. First is to normalize the text so as to eliminate duplicate words that arise from inconsistent casing. Second is to easily remove official hashtags in the tweets prior to ARFF conversion, the purpose of which was to train the classifiers based on the tweet text and not on the official hashtags.

3.4 Bag of Words ARFF Converter

The converter was used to transform the filtered tweets into a BoW representation in ARFF format, which is required to be able to use Weka. This was implemented using Java and the Weka external library.

[6] SQLiteStudio is a SQLite database manager for Windows. http://sqlitestudio.pl/

3.5 Classification Model Trainer

Weka was used to train the SVM and Naive Bayes classifiers using the produced ARFF files. Weka also provided the functionality for applying the TF-IDF weighting scheme to the Bag of Words ARFF, and for saving the classification models, which can be used for application or some other future work.

4. TESTING AND DISCUSSION

Both the SVM and Naïve Bayes classifiers were tested using Weka's built in ten-fold cross validation. The whole dataset was partitioned by Weka into ten subsets. Nine out of the ten subsets were used as a training set to develop the classifier; the unused subset was used as the testing set. This is referred to as a fold. This process was then repeated ten times with each subset being used as the testing set. The metrics used are precision, recall, the F-score, and the kappa statistic.

Table 3 shows the precision and recall for the SVM classifier. Precision and recall are broken down into the six categories, and their average is taken. The same is shown for the Naïve Bayes classifier in Table 4. Table 5 shows a comparison of the two classifiers in terms of these two metrics along with the F-score and kappa statistic.

Table 3. Performance metrics of the SVM classifier

Category	Precision	Recall
Resource coordination	0.866	0.870
Urgent rescue needed	0.860	0.755
Urgent rescue resolution	0.000	0.000
Damage reporting	1.000	0.333
Missing people	0.971	0.926
Media storm coverage	0.895	0.919
Average	*0.887*	*0.889*

Table 3 shows that the SVM classifier performed relatively well for missing people, media storm coverage, urgent rescue needed, and resource coordination as compared to damage reporting and urgent rescue resolution in both metrics, with the lowest value being 0.755 for recall of resource coordination. This means that the classifier was able to categorize tweets belonging to these categories quite well. On the other hand, the classifier's performance for urgent rescue resolution and damage reporting was very poor. This was expected due to the dataset containing very few instances of tweets belonging to both categories. Only ten tweets were labeled under urgent rescue resolution in the dataset, and only six tweets were for damage reporting.

Table 4. Performance metrics of the Naïve Bayes classifier

Category	Precision	Recall
Resource coordination	0.921	0.533
Urgent rescue needed	0.901	0.689
Urgent rescue resolution	0.077	0.100
Damage reporting	0.158	0.500
Missing people	0.964	0.884

Media storm coverage	0.724	0.952
Average	*0.819*	*0.782*

It can be observed in Table 4 that the Naïve Bayes classifier performed well for missing people and media storm coverage, although the precision for media storm coverage was not very high. The same can be said for the precision of media storm coverage in the SVM classifier. This was likely due to the vast amount of tweets belonging to this category in the dataset, leading the classifiers to incorrectly classify negatives as positives. The Naïve Bayes classifier's recall for resource coordination, urgent rescue needed, and damage reporting was moderately good. Despite this, the precision for urgent rescue needed and resource coordination was very high. However, the same is not true for damage reporting, but it is again probably due to the very small number of tweets belonging to this category in the dataset. Lastly, the classifier performed poorly for urgent rescue resolution, which can most likely be attributed to the small number of tweets under this category.

Table 5. Comparison of the performance of the SVM and Naïve Bayes classifiers

Metric Mean	SVM	Naïve Bayes
Precision	0.887	0.819
Recall	0.889	0.782
F-score	0.887	0.773
Kappa statistic	0.817	0.626

As can be seen in Table 5, the SVM classifier outperformed the Naïve Bayes classifier in both metrics. The difference in recall reached around 10.7%, while the difference in precision was around 6.8%. Overall, the SVM classifier scored a bigger F-score than Naïve Bayes by 11.4%. Furthermore, SVM had a kappa statistic of 0.817 while Naive Bayes had 0.626. This shows that there is stronger agreement between the classifications and the true classes for SVM than for Naive Bayes.

5. CONCLUSION AND FUTURE WORK

In conclusion, we have presented an SVM classifier and a Naive Bayes classifier that categorized tweets related to Philippine typhoons into resource coordination, urgent rescue needed, urgent rescue resolution, damage reporting, missing people, and media storm coverage. The results show that the SVM classifier performed better than the Naïve Bayes classifier. The former scored a precision of 81.9%, a recall of 88.9%, an overall F-score of 88.7%, and a kappa statistic of 81.7%. On the other hand, Naïve Bayes had 81.9% precision, 78.2% recall, an overall F-score of 77.3%, and a kappa statistic of 62.6%. These metrics show that the SVM classifier can be utilized for classifying Philippine typhoon related tweets with better results. Naïve Bayes probably performed poorly because it is known to require a large amount of training data to provide reliable results. This weakness is inherent in its evaluation because it forms very simplistic assumptions about the dataset, as reported in the literature [9].

To further improve the classifiers, there has to be more tweets belonging to the urgent rescue resolution and damage reporting categories in the dataset. The developed classifiers performed poorly in general for these two categories mainly due to the lack of tweets under them. That is, the classifiers performed poorly for categories which had very little training data. In addition, other better performing feature selection methods like Feature Abstraction [1] might improve the performance of the classifiers.

Lastly, the system could also be improved by adding a category for tweets that do not belong to any of the six categories. This is because practical use of the classifiers would definitely require the need for filtering out irrelevant tweets. This can be done by using binary classifiers [1] for each of the category instead of using one classifier so that tweets which do not belong will hopefully be categorized as negative by all the binary classifiers. Additionally, having binary classifiers would allow the classification of tweets into multiple categories, which may be beneficial for tweets that contain multiple types of information.

6. REFERENCES

[1] Caragea, C., McNeese, N., Jaiswal A., Traylor, G. Kim, H., Mitra, P., …, and Yen, J. 2011. *Classifying text messages for the Haiti earthquake*. Paper presented at the 8th International ISCRAM Conference. Lisbon, Portugal.

[2] Collins, K. (2013). *How AI, Twitter and digital volunteers are transforming humanitarian disaster response*. Retrieved from http://www.wired.co.uk/news/archive/2013-09/30/digital-humanitarianism

[3] Domingos, P. and Pazzani, M. 1997. On the optimality of the simple Bayesian classifier under zero-one loss. *Machine Learning*, *29*(2-3), 103-130. doi: 10.1023/A:1007413511361

[4] Imran, M., Elbassuoni, S., Castillo, C., Diaz, F., and Meier P. 2013. Proceedings of the 10th international ISCRAM Conference: *Extracting Information Nuggets from Disaster-Related Messages in Social Media*. Baden-Baden, Germany.

[5] Maron, D. F. 2013. *How social media is changing disaster response.* Retrieved from http://www.scientificamerican.com/article.cfm?id=how-social-media-is-changing- disaster-response

[6] Meier, P. 2013. Update: twitter dashboard for disaster Response. Retrieved from http://irevolution.net/2013/02/11/update-twitter-dashboard/

[7] Rennie, J. D. M., Shih, L., Teevan, J., and Karger D. R. 2003. Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003): *Tackling the Poor Assumptions of Naïve Bayes Text Classifiers*. Washington DC, the United states of America.

[8] Robertson, S. 2004. Understanding inverse document frequency: on theoretical arguments for IDF. *Journal of Documentation, 60*, 503-520.

[9] Shimueli G, Patel, N. R., Bruce, P.C. 2010, October 26. D*ata mining for business intelligence .(2nd ed.)* (2010, October 26). John Wiley & Sons.

[9] Verma, S., Vieweg, S., Corvey, W. J., Palen, L., Martin, J. H., Palmer, M., …, and Anderson, K. 2011. Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media: *Natural Language Processing to the Rescue?: Extracting "Situational Awareness" Tweets During Mass Emergency.* Barcelona, Spain.