2009 International Conference on Machine Learning and Applications

Text Classification Methodologies Applied to Micro-text in Military Chat

Kevin Dela Rosa
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA, USA
kdelaros@cs.cmu.edu

Jeffrey Ellen
SPAWAR Systems Center Pacific
San Diego, CA, USA
jeffrey.ellen@navy.mil

***Abstract*—We propose methods to classify lines of military chat, or posts, which contain items of interest. We evaluated several current text categorization and feature selection methodologies on chat posts. Our chat posts are examples of 'micro-text', or text that is generally very short in length, semi-structured, and characterized by unstructured or informal grammar and language. Although this study focused specifically on tactical updates via chat, we believe the findings are applicable to content of a similar linguistic structure. Completion of this milestone is a significant first step in allowing for more complex categorization and information extraction.**

Text Classification; Natural Language Processing; Micro-text; Chat; Support Vector Machines; k-Nearest Neighbor; Rocchio; Naïve Bayes;

I. INTRODUCTION

We experimented with applying text classification methodologies to a new domain, that of 'micro-text' chat entries. We consider 'micro-text' to have three main characteristics that separate it from the traditional documents used in text categorization:

- Individual author contributions are very brief, consisting of as little as a single word, and almost always less than a paragraph. Frequently the contribution is a single sentence or less.
- The grammar used by the authors is generally informal and unstructured, relative to the pertinent domain. The tone is conversational, and frequently unedited therefore errors and abbreviations are more common.
- The text is 'semi-structured' by traditional NLP definitions since it contains some meta-data (in the case of chat it contains a timestamp, a room/channel, and an author) in proportion to some free-text

While military chat rooms are our specific focus, we believe the results of our research and lessons learned are not specific to the medium of chat, but are applicable to any situation involving extremely brief 'micro-text'. This would include point to point instant messaging via any protocol (such as XMPP), SMS (Short Message Service) common on mobile phones, transcriptions of voice conversations, and micro-blogging which has been popularized by Twitter and similar services.

The remainder of this paper is structured as follows: Section II elaborates on the domain of military chat analysis and the motivation behind applying text categorization methods on chat posts. Section III describes the corpus used in this study and the preprocessing that was performed on it. Section IV outlines the algorithms used for feature selection and text classification, as well as the metrics used to measure each classifiers performance. Section V details the experimental setup and results. Lastly, Section VI provides a discussion of the experimental results and describes some of the future research work planned.

II. MOTIVATION

Chat rooms have become a vital medium for war fighters and analysts to communicate information updates [1]. Unlike voice communications, which can quickly become a cacophony with more than a handful of participants, chat rooms support near simultaneous exchange of information amongst a large number of war fighters and analysts. Manually monitoring chat rooms is tedious, manual, and expensive in terms of manpower. Implementing algorithms that can help draw the watchstanders' attention to items of interest can result in significant cost savings as well as increased accuracy. The ultimate goal is to increase 'Situational Awareness', which has a particular connotation for military use, but has a corollary for most any industry, business, or community of interest.

Assuming we can classify the 'micro-text' in chat rooms consistently, we can then focus on more complex information extraction, which is planned as a follow up study to this work.

Information retrieval and extraction on free text (e.g. long form prose, newswire releases, emails, etc) is a relatively vibrant and burgeoning research area, but we found a lack of studies and experiments on shorter texts, especially where grammar is less formal and abbreviations are more common. As electronic communications become more prevalent, we expect 'micro-text' sources to become more common, and more important in day-to-day operations within every industry.

III. CORPUS

For this study we used a synthetic chat data set which we generated. The semantic content and grammatical structure are relatively similar to the actual classified-secret level chat

978-0-7695-3926-3 2009

DOI 10.1109/ICMLA.2009.49

data, so the NLP challenges are replicated even though the information is quite different. The examples in Fig. 1 characterize four rudimentary categories of 'updates', which itself is a category within the general 'item of interest' category. For the purposes of this unclassified study, all items of interest were of the 'update' type.

A. Chat Data Description

The corpus consists of thousands of individual lines of chat. Each chat line contained a post (the main message in the form of text), a generated timestamp, a randomly selected author, and a channel from our list of possibilities. Each post may contain a simulated naval ship update, an update plus some noise, or consist entirely of tactically insignificant chat data. Each update within a post consists of a ship name, or key, and an update value. These posts fit within our definition of 'micro-text'. Fig. 1 shows some example posts that appear in data set. Note that the author, timestamp, and other meta-data are not shown in the figure.

We generated five categories of synthetic posts: *filler* denotes a post that does not contain an update, *binary* denotes a post whose update has one of two possible values, *numeric* denotes a post whose update has some numeric value, *class-value* denotes a post whose update value comes from a finite set of possible values, and *text* denotes a post whose update value is some string of free text. Randomly selected posts from the NPS chat corpus [2] were used for noise, which make up the *filler* posts, and are randomly appended and/or pre-pended to posts with naval ship updates. Additionally, many of the *filler* posts contain random naval ship key and value terms, to simulate posts that are not tactically significant, but contain terms that appear in naval ship update lexicon but lack the structure required to be considered an item of interest. For the remainder of this paper *interest* posts refers to posts that contain updates, or in other words posts whose categories are *binary*, *numeric*, *class-value*, or *text*.

Sample Updates

Binary update (possible values are up or down):
`USS Olympia is up`

Numeric update:
`USS Antietam is at (22.299289, 58.086360)`

Class update (possible values are empty/standby/hot):
`USS New Hampshire is standby`

Text update:
`USS Enterprise is at Pearl Harbor HI`

Filler (no update):
`tired ... I did n't sleep much last night`

Sample chat lines (updates are in bold)

(1) wow !!! did n't realize that
(2) no shes not **New Hampshire is empty**
(3) oh .. never knew that standby
(4) hello U20 (16.381914, 140.673367)
(5) poor thing **Kidd is at (26.026418, 164.611238)** PART
(6) whisling dixie up
(7) up 27 male maryland **Ross is at Kings Bay GA** i dont have to know how to read to drive a nice car ..
(8) Yeah , he 's back

Figure 1. Example naval ship updates and chat lines

There are 19,898 posts in the data set used for this experiment, approximately half of which belonging to the *filler* category, with the other 4 categories taking the remainder near evenly. This distribution described here was used to ensure that enough examples of each category were present during the training and testing phases. The data was segmented for training and testing, with 9995 posts used for training and 9903 posts used for testing. The mean number of terms per post was 39, with a median of 33.

B. Normalization and Document Representation

The training set's vocabulary was normalized by lowering the case of each term, removing stop words, and lemmatizing, which was done with the aid of the NLTK toolkit [3]. It should be noted that lemmatizing was used instead of stemming because we feared a stemmer would incorrectly modify abbreviations and/or irregular terms which appear frequently in chat, particularly military chat. Additionally, a mapping was done for numbers and coordinates to special terms for generality.

Normalizing resulted in a vocabulary set of 3406 unique terms. Next, we performed feature selection on the normalized vocabulary, as described in Section IV.

Since we are interested in spotting posts that contain naval ship updates, we decided to consider each individual post a document. Each document was represented with commonly used document vector models. Specifically, in SMART notation [4], *ltc* vectors were most commonly used for our various text classifiers. Each document also had a corresponding category vector, consisting of a series of Boolean values signifying category membership for the document.

IV. METHODS AND PERFORMANCE MEASURES

Prior to classification, various feature selection strategies were used to reduce the dimensionality of the document vectors. In this study, we focused on two classification problems: a 5-class categorization problem (filler, binary, numeric, class-value, or text) and the more general binary classification problem of filler or interest. For the experiments, we implemented 4 different classifiers: Support Vector Machine (SVM), k-Nearest Neighbor (k-NN), Rocchio, and Naïve Bayes (NB).

A. Feature Selection

We experimented with a number of feature selection methods to reduce the dimensionality of the document vector space. The methods used include term selection based on document frequency (DF), χ^2-test, (CHI), information gain (IG), and mutual information (MI), all of which are detailed in [5].

Furthermore, during feature selection rare terms (DF ≤ 5) were eliminated, inspired by a feature selection method variation in [6]. This document frequency cut, or *df-cut*, resulted in a vocabulary of 762 unique terms. During our experiments we varied the number of terms selected by each method to measure classifier performance.

B. Classifiers

In this study, we selected 4 classifiers that are commonly known for their high performance in text categorization:

- Support Vector Machine (SVM)
- k-Nearest Neighbor (k-NN)
- Rocchio
- Naïve Bayes (NB)

A local implementation was used for the k-NN, Rocchio, and Naïve Bayes classifiers, all of which are detailed in [7]. For SVM, we utilized the freely available SVM^{light} package [8]. For k-NN, we set k = 45, which was the value that was empirically observed to perform the best on our synthetic chat data set. Other values that were experimented with include k = 5, 15, 30 and 60, but since k = 45 performed the best, only those results are shown in this paper.

C. Performance Metrics

A number of standard measures were used to evaluate the text classifiers' performance on the chat data, including precision, recall, and F_1 measure. Precision is the proportion of returned documents that are targets, recall is the proportion of target documents returned, and the F_1 measure is an evaluation metric that combines recall and precision [9].

For the multi-class categorization experiments, we provide both the *micro-averaged* and *macro-averaged* values of these measures. The micro-averaged calculations give equal weight to each document, while the macro-averaged values give equal weight to each category [9].

V. Experiments

A. Experimental Setup

We ran two experiments on the simulated chat data. The first experiment focused on determining how well each of the classifiers performed on the binary classification problem of determining whether a post was of tactical interest or not. The second experiment focused on determining how well each classifier was able to further determine what kind of naval ship update was contained in the post.

Within each experiment we tried each of the feature selection techniques listed in Section IV, but we have only included the results for the best performing ones. We varied the number of features used to evaluate the best number of features for each classifier. In general, MI and IG feature selection showed very poor results on the chat data set so their results are not listed here.

The numbers of features tested for each experiment were 762, 400, 200, and 80, which corresponds roughly to the following percentage of total features available: 100%, 50%, 25%, and 10%. Note that for all feature selection methods, terms with document frequency of 5 or less were eliminated.

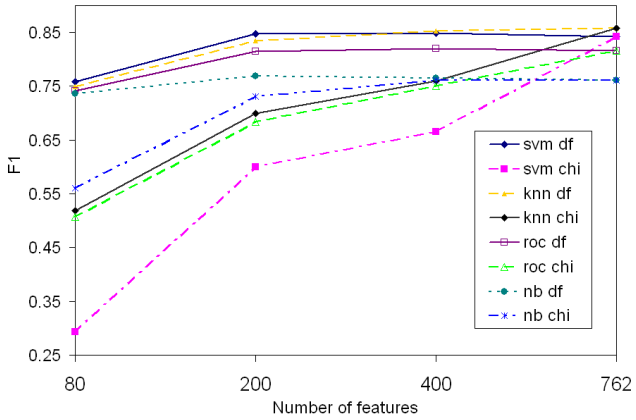

Figure 2. Performance of binary classifiers

TABLE I. Performance Summary of Best Binary Classifiers

method	recall	prec.	F1	# features	features
SVM	0.8034	0.8995	0.8487	400	DF
k-NN	0.8460	0.8713	0.8585	762	DF or CHI
Roc.	0.8359	0.8051	0.8202	400	DF
NB	0.7402	0.7219	0.7309	200	CHI

B. Binary Classification Experiment

For this experiment, we trained each classifier to identify *interest* posts. Table I shows the experimental results using the best performing feature selection method for each classifier. Fig. 2 shows how each classifier performed when we varied the number of features that were selected.

The k-NN classifier using 762 features performed the best in this experiment, with respect to the F_1 measure. The SVM classifier using 400 document frequency based features performed comparably well, with a slightly lower F_1 measure score.

C. Multi-class Classification Experiment

For this experiment, we trained each classifier to categorize a post into any of the following categories: *filler*, *binary*, *numeric*, *class-value*, or *text*. Table II shows the results for every configuration of each classifier, where *miR* is micro averaged recall, *miP* is micro averaged precision, *miF1* is micro averaged F_1 measure, and *maF1* is macro averaged F_1 measure. The values when the feature count was set to 80 were omitted due to the overall poor performance at that number of features.

Fig. 3 shows how the macro averaged F_1 measure varied with the number of features used for each classifier. Similarly, Fig. 4 shows how the micro averaged F_1 measure varied with the number of features used for each classifier.

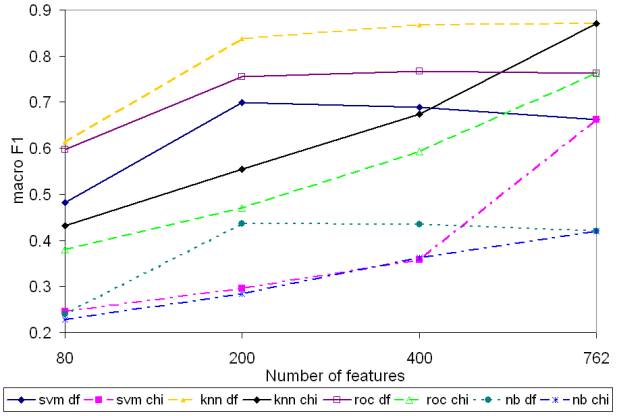

Figure 3. Performance of multi-class classifiers: macro averaged F1 measure

The k-NN classifier using 762 features performed the best in this experiment, with respect to the F_1 measure. The SVM classifier using 762 features performed comparably well, with a slightly lower F_1 measure score. Furthermore, the k-NN classifier appeared to have much better recall than the SVM, while the SVM had much better precision.

VI. Conclusions and Future Work

The empirical results of this study suggest that k-NN and SVM are well suited for categorizing our synthetic military chat data. This is significant because our literature search returned few examples of text classification applied to short documents and/or documents containing a large amount of informal language. Our study shows that these two standard text classifiers can be used in these domains.

Our results also suggest that for this domain feature selection based on document frequency tends to perform fairly well, chi-squared feature selection performs slightly worse, while information gain and mutual information tend to perform poorly. This is not unexpected given the concise nature of the source data.

Although results would be expected to vary with individual domain usage, we expect these results to provide a starting point for classification of micro-text.

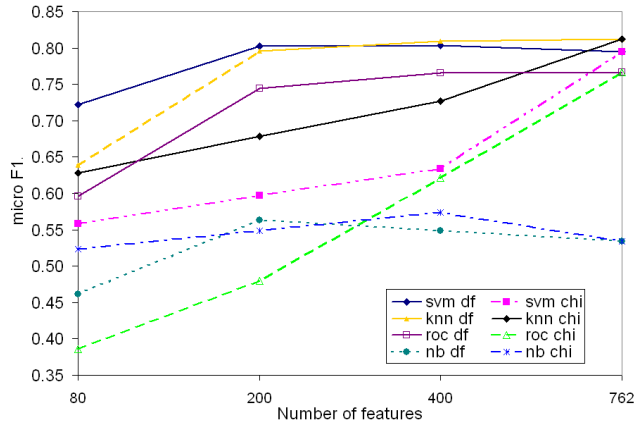

Figure 4. Performance of multi-class classifiers: micro averaged F1 measure

TABLE II. Performance Summary of Multi-class Classifiers

method	# feat.	miR	miP	miF1	maF1
SVM	762	0.7470	0.8480	0.7943	0.6621
k-NN	762	0.9586	0.7043	0.8120	0.8698
Roc.	762	0.8434	0.7020	0.7662	0.7625
NB	762	0.3698	0.9581	0.5336	0.4206
SVM CHI	400	0.6160	0.6519	0.6334	0.3586
k-NN CHI	400	0.8100	0.6598	0.7272	0.6737
Roc. CHI	400	0.6230	0.6203	0.6217	0.5933
NB CHI	400	0.4477	0.7978	0.5736	0.3632
SVM CHI	200	0.5871	0.6079	0.5973	0.2954
k-NN CHI	200	0.7401	0.6263	0.6784	0.5539
Roc. CHI	200	0.4633	0.4968	0.4795	0.4704
NB CHI	200	0.4479	0.7089	0.5490	0.2835
SVM DF	400	0.7596	0.8538	0.8039	0.6883
k-NN DF	400	0.9541	0.7022	0.8090	0.8682
Roc. DF	400	0.8353	0.7074	0.7661	0.7675
NB,DF	400	0.3847	0.9549	0.5485	0.4342
SVM DF	200	0.7616	0.8483	0.8026	0.6995
k-NN DF	200	0.9236	0.6995	0.7961	0.8379
Roc. DF	200	0.7967	0.6992	0.7448	0.7549
NB DF	200	0.4031	0.9358	0.5635	0.4370

Further studies must be done to find the optimal number of features used for each classifier, since the F_1 measures seemed to vary greatly for each classifier as the number of features was reduced. Additionally, we plan to make more extensive use of the available meta-data during feature set construction.

This preliminary study will help guide the further development of micro-text classification on actual military chat.

Acknowledgment

The authors thank the Office of Naval Research and the Space and Naval Warfare Systems Center, Pacific, Science and Technology Initiative for their support of this work. We would also like to thank Emily W. Medina for her guidance. This paper is the work of U.S. Government employees performed in the course of employment and no copyright subsists therein. This paper is approved for public release with an unlimited distribution.

References

[1] E. Medina, "Military Textual Analysis and Chat Research", IEEE-ICSC 2008 Proceedings, Santa Clara CA, August 4-7 2008.

[2] Eric N. Forsyth and Craig H. Martell, "Lexical and Discourse Analysis of Online Chat Dialog", Proceedings of the First IEEE International Conference on *Semantic Computing* (ICSC 2007), pp. 19-26, September 2007.

[3] E. Loper and S. Bird, "NLTK: The natural language toolkit", 2002. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.4.585

[4] G. Salton, Automatic text processing: the transformation, analysis, and retrieval of information by computer. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1989

[5] Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in *ICML '97: Proceedings of the Fourteenth International Conference on Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1997, pp. 412-420.

[6] M. Rogati and Y. Yang, "High-performing feature selection for text classification," in *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management*. New York, NY, USA: ACM Press, 2002, pp. 659-661.

[7] Manning, C.D., P. Raghavan, and H. Schutze, *Introduction to Information Retrieval*, Cambridge University Press, July 2008.

[8] T. Joachims, "Making large-scale support vector machine learning practical", *Advances in kernel methods: support vector learning*, pp 169-184.

[9] "Evaluation of Text Categorization Systems," April 2007. [Online]. Available: http://cs.ubbcluj.ro/~zbodo/docs/Writings/eval.pdf