An Assessment of Case Base Reasoning for Short Text Message Classification

Matt Healy[1], Sarah Jane Delany[1], and Anton Zamolotskikh[2]

[1] Dublin Institute of Technology,
Kevin Street, Dublin 8, Ireland
matt.healy@student.dit.ie, sarahjane.delany@comp.dit.ie
[2] University of Dublin, Trinity College,
Dublin 2, Ireland
zamolota@cs.tcd.ie

Abstract. Message classification is a text classification task that has provoked much interest in machine learning. One aspect of message classification that presents a particular challenge is the classification of short text messages. This paper presents an assessment of applying a case-based reasoning approach that was developed for long text messages (specifically spam filtering) to short text messages. The evaluation involves determining the most appropriate feature types and feature representation for short text messages and then comparing the performance of the case-based classifier with both a Naïve Bayes classifier and a Support Vector Machine. Our evaluation shows that short text messages require different features and even different classifiers than long text messages. A machine learner which is to classify text messages will require some level of configuration in these aspects.

1 Introduction

Message classification is a text classification task that has provoked much interest in machine learning [1–4]. One aspect of message classification that presents a particular challenge is the classification of short text messages where the *signature* of the concept being learned is weak. This is an important issue as there are a number of message classification application areas where short text is inherent. The classic example is the Short Message Service (SMS) where a text message has a limit of 160 characters. Other application areas include classification of text entered into comment boxes, online or otherwise.

This paper presents an assessment of using case-based reasoning for the classification of short text messages. Our earlier work on message classification has been in the area of spam filtering [3, 5]. We propose in this paper, to evaluate and extend this case-base approach to the classification of short text messages. We evaluate the case-based classifier on a number of datasets of two types, SMS messages and customer comments from guests of a large hotel chain. We assess how the configuration of the case-based classifier (with respect to the feature

representation, feature selection and case selection policies) differs from that used for email (long) text messages.

This paper begins with an overview of existing research into short text message classification in Section 2. Section 3 then describes the case-based approach that we use. Section 4 discusses the evaluation of the case-based approach on short test messages, describing the datasets used, the evaluation methodology and the most appropriate configuration of the classifier for short text. In Section 5 we compare the performance of the case-based classifier to two other machine learning approaches, Naïve Bayes (NB) and Support Vector Machines (SVM) both of which are popular in text classification. The paper concludes in Section 6 with directions for future work.

2 Review of Existing Short Text Classification Approaches

The machine learning approach to text classification has been studied and analysed for many years [6–9] but there has been little previous work in the short text classification domain. The techniques used for text classification work well for datasets with large documents such as scientific papers but suffer when the documents in the training corpus are short. The performance loss can be attributed to the weak signature of the concept being modeled due to the short length of the text.

Previous research into short text classification has focused on including additional information with the training data to aid the classification process. Zelikovitz's [10] approach to short text classification uses Latent Semantic Indexing (LSI). LSI is an unsupervised learner that creates a reduced vector space through singular value decomposition (SVD). Zelikovitz combines the training data with the unlabeled test examples when creating the reduced vector space. She concludes that this expanded feature space includes semantic associations that help to classify short text documents. Zelikovitz has also used unlabeled background information that is related in some way to the training data [11, 12]. For example if the classification task is to classify titles of scientific papers, the unlabeled background information used could be the abstracts of the papers in the training dataset.

3 A Case-based Approach to Text Classification

In this section we describe Email Classification Using Examples (ECUE) the case-based approach that was used to classify email into spam and legitimate email [3]. We outline the design decisions that were made for long text messages and our objectives in the assessment of applying ECUE to short text messages.

The ECUE system extracts three types of features, these are word features (i.e. a sequence of characters separated by white space), single characters features (i.e. letters) and statistical features (e.g. the proportion of uppercase or

punctuation characters). This combination of features gave the best generalisation accuracy for spam (long text) classification. For this evaluation we need to determine the combination that is most appropriate for short message classification.

There are two possible feature representations for text features; binary (i.e. true or false, indicating that a particular feature simply exists in the text or not) and numeric (i.e. a number representing the frequency of a particular feature in the text). The ECUE system uses binary feature representation as it was found to produce the best generalisation accuracy for spam filtering [3].

ECUE represents a case e_i in the case-base as a vector of features values, $e_i = (f_1, f_2...f_n, s)$, where f is a feature and s is the class. Binary feature representation for word features uses the existence rule i.e. if the feature exists in the case f_i =1 otherwise $f_i = 0$. For statistical and single character features we use the Information Gain (IG) [13] value, as calculated during the feature selection process, to determine if f_i is set to 1 or 0. This is determined by comparing the normalised frequency of the feature with the threshold value which returns the highest IG. If the normalised frequency is greater than the threshold value $f_i = 1$ otherwise $f_i = 0$. For numeric feature representation we simply use the normalised frequency of a feature. In this evaluation we will determine which feature representation is appropriate for short message classification.

ECUE uses a k-nearest neighbour classifier which returns the k-nearest neighbours (k-NN) that are most similar to the target case. A False Positive (FP) (a legitimate message classified incorrectly as spam) is significant for spam filtering as a legitimate email being misclassified as spam is unacceptable in most situations. To reduce the rate of FPs, ECUE uses the k-NN algorithm with unanimous voting to bias the classifier away from FPs. In unanimous voting all k-nearest neighbours have to be classified as spam before the test email (case) is classified as spam otherwise it is classified as a legitimate email. As we are looking at general message classification of short text messages, FPs are not significant so in this evaluation we use the k-NN algorithm with weighted distance voting [14].

The ECUE system uses a case-base editing technique called Competence Base Editing (CBE) [15] to manage the size of the case-base by removing noisy and redundant cases. It was found that editing a case-base using CBE yields the best generalisation accuracy in the spam filtering domain [15]. In this paper we will evaluate CBE to determine if it improves generalisation accuracy for short text case-bases.

4 Evaluation of CBR for Short Text Message Classification

In this evaluation there are two objectives. The first objective is to determine the most appropriate case representation for short text message classification. The second objective is to determine if the CBE editing technique is effective for short text messages.

4.1 Datasets Used

This assessment used two types of datasets that contain short text messages, a corpus of customer comments and a corpus of Short Message Service (SMS) messages.

The customer comments corpus consists of over 5000 comments from guests visiting hotels that are part of a large hotel chain. These comments are classified as *satisfactory* for guests that were happy with the service provided by the hotel and *unsatisfactory* for guests that were unhappy. The customer comments corpus was divided into four datasets consisting of 500 satisfactory and 500 non satisfactory comments. The comments themselves range from a few words e.g. "V good" to a detailed description of what a guest found good or bad e.g. "Enjoyed our stay in our family room immensely. Can't wait to come back. Kids loved it." The customer comments present a particular challenge for a text classifier as the difference between satisfactory and non satisfactory comments can be slight, for example "The room was good" and "The room was not good".

The SMS corpus consists of two datasets with 100 legitimate and 100 spam messages in both. The legitimate SMS messages consist of personal and business text messages and the spam SMS messages contain promotional SMS messages and text alerts. While legitimate SMS messages are normally from personal corespondents and are normally short messages such as "Where are you?", spam SMS are normally from companies who are trying to offer some service or product such as "1000 Download as many ring tones as u like no restrictions, 1000s 2 choose. U can even send 2 yr buddys. Txt Sir to 80082 EUR3".

4.2 Evaluation Metrics

The main evaluation metric that will be reported in this evaluation is the percentage error, i.e. the percentage of test instances incorrectly classified by the classifier. For the SMS message datasets the rate of FPs will also be reported, as similar to the situation with spam filtering, an FP is not acceptable in the SMS domain.

4.3 Evaluation Methods

The evaluation method used for each dataset was a 10 fold cross-validation, dividing the dataset into 10 stratified divisions or folds. In this method each fold in turn is used as a test dataset while the rest of the nine folds are considered to be the training dataset. A case-base was built from each training dataset using the top 500 features ranked using Information Gain [13]. For each fold and test set combination the performance measures were calculated for different case-base configurations (e.g. with different feature types and feature representation).

Confidence levels were calculated using McNemar's test [16] to determine whether significant differences exists between any two case-base configurations. McNemar's test has some advantages over other performance measures (e.g.

paired t-test), it has a lower Type I error (the probability of incorrectly detecting a difference when no difference exists) and has a better ability to detect a deference where one exists [16].

4.4 Evaluation to Determine Case Representation

The objective of this evaluation was to determine (i) the combination of either word, statistical and/or single character features and (ii) the feature representation (binary or numeric) that gives the best generalisation accuracy.

We performed a number of experiments varying k from $k = 1$ to $k = 9$ and varying the type of features used. The combination of the types of feature we evaluated are word features only, word and statistical features, word and letter features and word, statistical and letter features. Our results indicated that for both types of datasets the types of features that performed best are word and statistical features with no letter features. This combination of feature types is different from the combination that ECUE uses for email messages. ECUE uses all three types of features to filter spam emails. It is not surprising that letter features are not predictive for the datasets we have used here. Email spammers use obfuscation to confuse email filters by including punctuation in the middle of words, e.g. V.1:a.g.r:a or by replacing certain letters, e.g. 'i' with 1's or 'l's. Spammers also tend to use a lot of uppercase characters e.g. I.N.V.E.S.T.M.E.N.T O.P.T.I.O.N.S. This would explain why letters are so predictive. The text message and customer comments datasets use normal structured English where letters would not necessarily be as predictive. Also SMS spam is in its infancy and SMS spammers have not had to obfuscate their text message to bypass filters yet.

Our experiments also found that a k-NN classifier with k=7 gave the best performance for the customer comments datasets and a k-NN classifier with k=3 gave the best performance for SMS messages datasets. This indicated that the signature of SMS spam is stronger and more easily differentiated from legitimate SMS messages, even with short text, than that of satisfactory and unsatisfactory customers comments. This is to be expected as customer comment messages with different classifications can differ in as little as just one single word.

Our next objective was to determine the best feature representation, either binary or numeric. We ran our experiments using the combination of feature types which gave the best generalisation accuracy (i.e. word and statistical features) and compared the results for a binary and numeric feature representation. Figure 1 shows the percentage error for binary and numeric feature representation for each of the four customer comments and the overall result across all four datasets.

The results for the customer comments dataset shows that a numeric representation of features gives better performance and reduces error across all four datasets. The difference between numeric and binary representation for dataset 1, 2 and 3 are significant at the 99.9% confidence level whereas the difference for dataset 4 and the overall result are significant at the 90% confidence level.

Figure 2 shows the percentage error and FP rate for each of the SMS datasets and the overall result across both datasets. The graphs in Figure 2 show that a binary representation of features gives better performance and reduces the error and the FP rate across all datasets. The difference in the percentage error between numeric and binary representation is only significant for dataset 1 and the overall result at the 95% confidence level whereas the difference in the FP rate is only significant at the 95% confidence level for dataset 1.

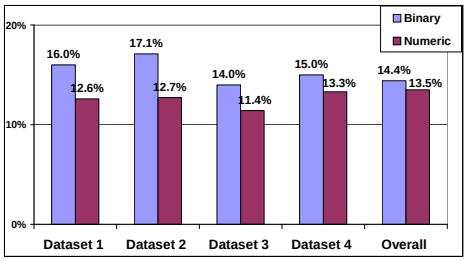

Fig. 1. Results of comparing binary and numeric feature representation for the customer comments datasets

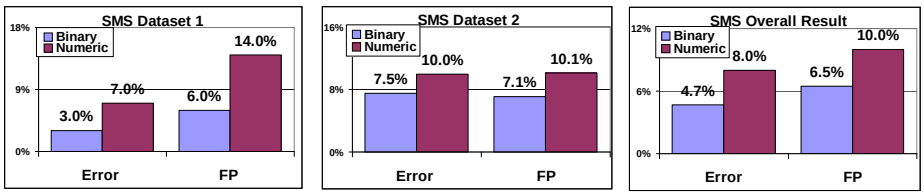

Fig. 2. Results of comparing binary and numeric feature representation for the SMS datasets

4.5 Evaluation of Case-Base Editing Technique

The objective of this evaluation is to determine whether applying the case-base editing technique (CBE) will improve the generalisation accuracy of a case-base of short text messages. Figures 3 and 4 show the results for the customer comments datasets and the SMS datasets respectively. The reduced size of each dataset is included in the figures as a percentage of the original size.

The differences for the customer comments datasets are all significant at the 95% level or higher except for dataset 2 where the difference is not significant. None of the differences on the SMS datasets are significant in any case.

The results show that the editing technique CBE is not appropriate for short text message classification. One of the objectives of CBE was to conservatively reduce the size of the case-base [15]. ECUE reduces an email case-base by approximately 30%, but applying CBE to the customer comments and SMS datasets results in an overall average reduction of 56% and 65% respectively. This suggests that the sparsity of the cases due to the short text content of the messages is not appropriate for the editing technique resulting in too many cases being removed.

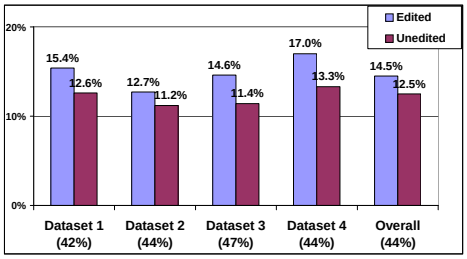

Fig. 3. Results of applying CBE to the customer comments datasets

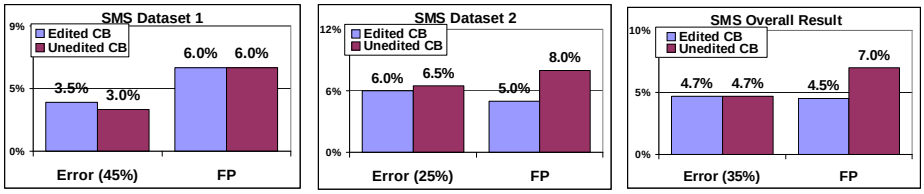

Fig. 4. Results of applying CBE to the SMS datasets

5 Comparison with Naïve Bayes and Support Vector Machines

Naïve Bayes (NB) [14] and Support Vector Machines (SVMs) [17] are both popular classifiers used in text categorisation. Our comparisons of the ECUE system with NB for spam filtering concluded that neither classifier outperformed the other consistently but that the k-NN classifier was in fact better at handling the concept drift in the email [5]. For the purposes of general message classification, it is important to compare the performance of the k-NN classifier on short text messages with that of NB and an SVM.

We evaluated using a NB classifier and an SVM on each of the six datasets. The NB implementation used is that described by Delany et al. [3] while the SVM implementation used is a 2-norm soft-margin SVM as described in [17] with a normalised dot product kernel function. The results are displayed in Figures 5 and 6 respectively.

It is evident from Figure 5 that NB and the SVM consistently outperform the k-NN classifier for both datasets. The differences between NB and k-NN are significant at the 99.9% level in all cases, whereas the differences between the SVM and k-NN are significant at the 99% level or higher in all cases except dataset 3 where there is no significant difference. This is contrary to what was found for email messages. Email, both spam and legitimate, is a diverse concept; spam offering cheap prescription drugs has little in common with spam covering investment opportunities and personal email messages will be quite different from business email.This suggests that the lack of diversity in the customer comments datasets is not appropriate for a local learner like k-NN but is more appropriate for a classifier that uses a global concept like NB or SVM.

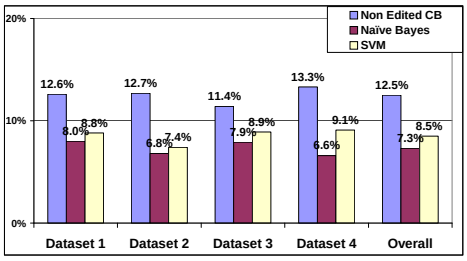

Fig. 5. Results of comparing the k-NN classifier with Naïve Bayes and SVM on the customer comments datasets

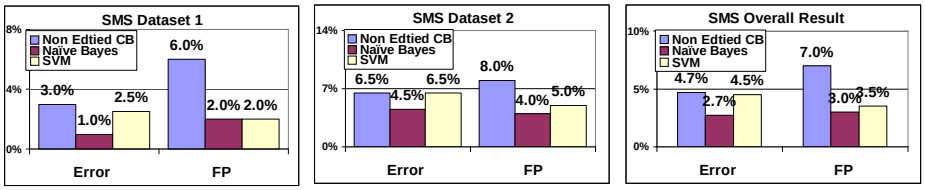

Fig. 6. Results of comparing the k-NN classifier with Naïve Bayes and SVM on the SMS datasets

The SMS figures reported in Figure 6 show similar results in that both NB and SVM outperform k-NN in all cases with equal or lower overall error rates and lower FP rates. In spite of some of the differences being large, (e.g. on dataset

1 the k-NN FP rate of 6% compared with a NB or SVM FP rate of 2%), none of the differences are significant using McNemar's test. This is most probably due to the small numbers of messages in the SMS datasets under consideration. It will be important to source a significant number of SMS text messages, both spam and legitimate, to confirm these results.

6 Conclusions and Future Work

In this paper we have identified that the most appropriate types of feature to use for short text message classification are words and statistical features. Unlike longer email message classification, letter features do not improve performance. We have also shown that the feature representation used is dependent on the domain and types of data that are being classified. Email and text messages require a binary representation but the classification of the customer comment text messages requires a numeric representation which includes feature frequency information.

The results of our evaluations presented here have shown that short text messages require different feature representation, different feature types and even different classifiers than longer email messages to achieve best performance. This suggests that any machine learning system which is to classify text messages needs to be configurable in all these respects. The configuration could be automatically performed using the data on which the system will be trained.

Our future work in this area is to extend the classifier to cater for multiple classifications to facilitate such applications such as message routing or general email filtering.

References

1. Busemann, S., Schmeier, S., Arens, R.G.: Message classification in the call center. In: Procs of the 6th conference on Applied natural language processing, Morgan Kaufmann (2000)
2. Neumann, G., Schmeier, S.: Combining shallow text processing and machine learning in real world applications. In: Proc of the 16th International Joint Conference on Artificial Intelligence (IJCAI '99). Workshop on Machine Learning for Information Filtering, Stockholm, Sweden (1999)
3. Delany, S., Cunningham, P., Coyle, L.: An assessment of case-based reasoning for spam filtering. In McGinty, L., Crean, B., eds.: Procs. of 15th Irish Conference on Artificial Intelligence and Cognitive Science. (2004) 9–18
4. Androutsopoulos, I., Koutsias, J., Chandrinos, K.V., Spyropoulos, C.D.: An experimental comparison of naive bayesian and keyword-based anti-spam filtering with personal e-mail messages. In: SIGIR '00: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval, ACM Press (2000) 160–167
5. Delany, S.J., Cunningham, P., Tsymbal, A., Coyle, L.: A case-based technique for tracking concept drift in spam filtering. In Macintosh, A., Ellis, R., Allen, T., eds.: Applications and Innovations in Intelligent Systems XII, Procs. of AI 2004, Springer (2004) 3–16

6. Sebastiani, F.: Machine learning in automated text categorization. In: CM Computing Surveys. (2002) 1–47
7. Joachims, T.: Text categorization with support vector machines: Learning with many relevant features. In: Proc of the 10th ECML, Springer (1999)
8. Cohen, W., Singer, Y.: Context-sensitive learning methods for text categorization. In: Proc of SIGIR-96. (1996)
9. Lewis, D.: Feature selection and feature extraction for text categorization. In: Proceedings of Speech and Natural Language Workshop. (1992) 212–217
10. Zelikovitz, S.: Transductive LSI for short text classification problems. In: Proceedings of the 17th International FLAIRS Conference. (2004)
11. Zelikovitz, S., Hirsh, H.: Improving short-text classification using unlabeled background knowledge to assess document similarity. In: Proceedings of the Seventeenth International Conference on Machine Learning (ICML). (2000)
12. Zelikovitz, S., Hirsh, H.: Using LSI for text classification in the presence of background text. In: Proceedings for the Conference on Information and Knowledge Management (CIKM). (2001)
13. Quinlan, J.R.: C4.5 Programs for Machine Learning. Morgan Kaufmann Publishers Inc., San Mateo, CA. (1997)
14. Mitchell, T.: Machine Learning. McGraw Hill, New York (1997)
15. Delany, S.J., Cunningham, P.: An analysis of case-based editing in a spam filtering system. In Funk, P., P.González-Calero, eds.: 7th European Conference on Case-Based Reasoning (ECCBR 2004). Volume 3155 of LNAI., Springer (2004) 128–141
16. Dietterich, D.T.: Approximate statistical tests for comparing supervised classification learning algorithms. Neural Computing **10** (1998) 1895–1923
17. Christianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines: And Other Kernel-based Learning Methods. Cambridge University Press (2000)