Classification in e-procurement

Paul J. Roberts

@UK plc,
5 Jupiter House, Calleva Park, Aldermaston, RG7 8NN
E-mail: paul.roberts@ukplc.net

Richard J. Mitchell* and Virginie F. Ruiz

School of Systems Engineering,
University of Reading,
Whiteknghts, Reading, Berks RG6 6AY, UK
E-mail: r.j.mitchell@reading.ac.uk
E-mail: v.f.ruiz@reading.ac.uk
*Corresponding author

J. Mark Bishop

Department of Computing,
Goldsmiths, University of London,
New Cross, London, SE14 6NW, UK
E-mail: m.bishop@gold.ac.uk

Abstract: Three coupled knowledge transfer partnerships used pattern recognition techniques to produce an e-procurement system which, the National Audit Office reports, could save the National Health Service £500 m per annum. An extension to the system, GreenInsight, allows the environmental impact of procurements to be assessed and savings made. Both systems require suitable products to be discovered and equivalent products recognised, for which classification is a key component. This paper describes the innovative work done for product classification, feature selection and reducing the impact of mislabelled data.

Keywords: classification; feature selection; noise reduction; e-procurement.

Reference to this paper should be made as follows: Roberts, P.J., Mitchell, R.J., Ruiz, V.F. and Bishop, J.M. (2014) 'Classification in e-procurement', *Int. J. Applied Pattern Recognition*, Vol. 1, No. 3, pp.298–314.

Biographical notes: Paul J. Roberts graduated with a first class degree in Computer Science from the University of Reading in 2006. He was then employed as a Knowledge Transfer Associate in the Classification Project, one of three integrated projects between @UK plc, the University of Reading and Goldsmith College, London. His work for the project contributed to his PhD in Automatic Project Classification which was awarded in 2011. He is still working at @UK.

Richard J. Mitchell received his BSc (hons.) in Cybernetics and Control Engineering and PhD in Cybernetics from the University of Reading, Reading, UK, in 1980 and 1987, respectively. He was appointed Lecturer in Cybernetics in 1983 and is now Senior Lecturer in Cybernetics and also Senior Tutor in the School of Systems Engineering, University of Reading. He won a University Teaching Fellowship in 2011. He has published four textbooks, edited two custom books on cybernetics, and has over 100 research papers in control engineering, robotics, and learning systems. He was Lead Academic on the Classification KTP project.

Virginie F. Ruiz received her BSc, MSc and PhD in Signal Processing from the University of Rouen, France. She was a recipient of the French Foreign Office, Lavoisier programme. Her research focuses on the theory and application of non-linear filtering for estimation, detection, prediction, analysis, and recognition. She joined the Department of Cybernetics at University of Reading in 1998 and is now Professor of Signal Processing. She was Academic Supervisor for the Classification Project. She is a member of many technical programme committees for international conferences and serves as reviewer for a number of international journals.

J. Mark Bishop is a Professor of Cognitive Computing at Goldsmiths, University of London and Chair (elect) of the AISB (the UK Society for the Study of Artificial Intelligence and the Simulation of Behaviour). His research is in the field of cognitive computing: its theory – where his interests centre on the foundations of stochastic diffusion processes, for which he was awarded his PhD from the University of Reading in 1989; its application; and its philosophical foundations. Together with John Preston, he has co-edited a critique of John Searle's arguments against machine intelligence, *Views into the Chinese Room* (OUP, 2002).

1 Introduction

Three coupled knowledge transfer partnerships (KTP) between the University of Reading, Goldsmiths College and @UK plc, a leading electronic marketplace provider, have generated the SpendInsight system. This uses artificial intelligence techniques to enable e-procurers to analyse their purchases and identify potentially significant savings. As an added benefit, the carbon footprint of products can be found and an environmentally friendly procurement policy developed.

The overall aim of the three KTP projects was to develop a product- and location-aware search engine which would be a key component in the @UK plc e-procurement and e-marketplace platform, adding value by improving the user experience. One project concentrated on spidering the web to determine available products. A second ensured that user queries were ranked suitably. The third developed systems to classify the products found. Classification is the focus of this paper.

The products are categorised into popular product classification systems such as eClass used by the UK's National Health Service (NHS), National Supplier Vocabulary (NSV) and United Nations Standard Products and Services Code (UNSPSC). Such classifications are important to large purchasing organisations for spend analysis, but manually classifying supplier catalogues is a time consuming and expensive process.

Therefore, *automatic* classification would provide an important unique selling point for the marketplace platform.

The projects contributed to an integrated system named *SpendInsight* (2012). Key to the system is the ability to classify numerous different products from a variety of suppliers, determine equivalent products from different suppliers, assess the economic cost of each and hence choose the cheapest. Significant savings could then be made in various domains. This was confirmed in the report from the National Audit Office (2011), which concludes that across the 165 hospital trusts in England, annual savings of £500 m pounds could be made, over 10% of the £4.6 bn spending on consumables.

In addition, the 'environmental' cost of each classified product can also be allocated, and this is available in the associated system *GreenInsight* (2012). This allows e-procurers to assess the environmental cost of their purchases.

Classification is the central component of these systems, for which there is considerable research even in the domain of text classification (product information is textual). However, the text being processed here is very different from that used in most text classification problems. The purpose of this paper is to consider the approaches taken in the three stages needed to classify product data: fuller details are in Roberts (2011).

2 Product classification

The systems need to classify textual descriptions of products, that is they must assign one of a number of predefined classes to a document based upon the natural language therein. This is almost always achieved with supervised learning: the process of using a set of pre-classified example documents to predict the class of unseen documents.

Aggarwal and Zhai (2012) provide a comprehensive survey of text classification methods. Although text classification has been used for many practical purposes, the focus of the work here is product classification, in particular from purchase order (PO) lines, in a real-world setting. A labelled dataset is formed using 2,179,122 PO lines, taken from 87 NHS trusts. It has 909 distinct labels, each PO line is described only by a short description and there are many mislabelled documents. The large numbers of training items and classes mean that implementing classification algorithms so that they complete in an acceptable amount of time and use an acceptable amount of resources is difficult and many standard implementations are unusable.

The purpose of classifying NHS PO lines is to allow the expenditure to be analysed. An NHS trust typically buys around 15,000 distinct products in a given year. If these products are classified to a formal schema, analysis of spending patterns is possible, and areas can be found where negotiating contracts with suppliers would be particularly beneficial. If the average carbon footprint per pound spent on products of a class is known, then classified PO lines can be used to estimate the carbon footprint of an organisation, and to track how it changes over time.

Product classification is, in general, more difficult than standard document classification. This is because there are more numerous classes, the textural descriptions of products are generally very short (typically four or five words); they do not necessarily employ correct English spelling or grammar; and they contain irrelevant or subsidiary information such as trademarks or codes. For instance

> Pinnacle sector, acetabular cup/duofix HA sz 52 mm.

Here 'Pinnacle sector' is the brand name of an acetabular cup, which is used in hip replacements, 'duofix HA' is its name, and its size is 52 mm. Very few if any of such words could be found in a standard dictionary with the meaning intended here.

In general there are far more categories in product schemata: over 20,000 in UNSPSC (Hepp et al., 2005) than in typical document classification tasks [in the oft-used Reuters corpus (Yang, 1999) there are between 93 and 113 classes]. The class distribution is very skewed: few classes are very common, before a long tail of infrequent classes. Product classification also differs from many text classification tasks due to the presence of a formal schema into which products are classified.

Three stages are needed for this to be achieved: automatic product classification; feature selection and noise reduction. These are briefly introduced below, with more detail given in later sections.

For NHS trusts that wish to gain access to information on products, manual classification is usually used. This requires domain expertise (Fensel et al., 2001), is expensive (as much as £1 per item), and tends to have a large error rate. Therefore, suitable algorithms are needed to classify the data automatically.

Feature selection is the process of identifying which of the inputs are crucial in the process: when done correctly this reduces the amount of data which needs to be processed by the classifier and thus its complexity. Yang and Pedersen (1997) show that it also reduces the risk of overfitting and making the classifier more accurate. Liu and Yu (2005) give an overview of relevant feature selection methods.

Noise reduction is needed to address the two forms of noise in training data: textual noise, such as spelling mistakes, unconventional abbreviations and irrelevant text; and label noise, when the data have been incorrectly labelled. In the application here, data have been labelled by different people at different times, which lead to inconsistencies. Of these problems, textural noise is less of an issue (Agarwal et al., 2007), particularly when feature selection is used. Occasional label noise can be detected easily by looking at outliers. The main issue addressed here is systematic label noise often caused by overlapping and ambiguous classes.

In three sections below classification, feature selection and noise reduction are discussed and the associated experiments are described which were performed on two forms of data: the PO data, and, as a control, the Reuters corpus (Yang, 1999). First, however, an overview is given of previous research in these areas.

3 Overview of previous research

3.1 Automatic product classification

The two main published product classifiers are GoldenBullet and AutoCat. The GoldenBullet (Ding et al., 2002) product classifier uses vector space models, k-nearest neighbour and Naïve Bayes to classify products into UNSPSC. 41,913 products were manually classified into 421 UNSPSC categories as training and test sets. The best method was Naïve Bayes, where an accuracy of 78% was reported. They also attempted to take advantage of the hierarchical nature of the UNSPSC by building a classifier at each of the four levels of the UNSPSC which, surprisingly reduced the accuracy to 38%.

The AutoCat classifier (Wolin, 2002) uses a vector space model for product classification. It was tested with 206,301 products belonging to 272 commodities related

to computing, reporting an accuracy of 79.5%. They attempted to use numeric attributes such as product cost to alter class confidences, but this did not work well as category descriptions were too broad, and because of the bundling of multiple products. The standard deviation of product cost exceeded the mean in 82.5% of categories.

Neither classifier repeated their tests on the full UNSPSC tree; both used around 5% of it.

3.2 Feature selection

Yang and Pederson (1997) compared five feature selection methods in the domain of text classification, with a focus on very aggressive feature selection. They found that information gain, document frequency thresholding and χ^2 can remove over 90% of terms without losing accuracy, whereas term strength and mutual information performed poorly.

Forman (2003) compared eleven feature selection methods with a support vector machine, and analysed the effects of skew. He found that only his algorithm, bi-normal separation (BNS), and information gain improved performance, the latter being better when skew was low or when a high percentage of features was removed.

Li et al. (2001) identified two basic measurements upon which many feature selection methods are based. They show theoretically and experimentally that a spectrum can be defined, with a preference for higher class ratio and lower document frequency at one end, and the reverse at the other. Mutual information and BNS are at the former end of this spectrum, document frequency thresholding is at the other, and information gain and χ^2 are in between. They present a new method that uses cross-validation to find the appropriate place for the given domain to be on this spectrum. They tested their hypotheses on four corpora, all with class skew removed. It would be interesting to know how their method performed with class skew, and whether the best place on the spectrum is calculable.

Zheng et al. (2004) compared one-sided feature selection measures (such as the odds ratio) with two-sided measures (such as χ^2 and information gain). They found the implicit two-sided measures performed better, and so adapted the methods to be explicitly two-sided. They found that these measures were best, especially on imbalanced data.

3.3 Label noise reduction

Brodley and Friedl (1996) proposed a method of cross-validation, where training items are identified as noise if they are classified incorrectly by a validation classifier. Their tests of detecting artificial noise in satellite data were encouraging. When 20% of items had their classifications changed, 7.3% of good data was removed, and 35.5% of erroneous data was kept. Other studies have found that the method suffers from a high false positive rate (Gamberger et al., 1999; Verbaeten and van Assche, 2003).

Daza and Acuna (2007) created QcleanNoise, which uses the distance between training items and class centroids, and the training items' nearest neighbours to identify noise. They tested the method by switching the labels of a proportion of the two largest classes in the dataset. They reported high levels of precision without much cost in recall.

Verbaeten and van Assche (2003) compared cross-validation techniques with some based upon bootstrap aggregating (bagging) for detecting artificial noise in a training set with Boolean classes. They found that bagging was more precise but conservative, whereas majority-voting cross-validation was less conservative but less precise.

Ramakrishnan et al. (2005) used expectation maximisation to estimate probabilities of documents being mislabelled. They do not state how much noise they were able to identify, but report significant improvements in the performance of the classifier.

Gamberger et al. (1999) presented a method of finding noise and other outliers by iteratively reducing the complexity of the least complex correct hypothesis. This method performed very well on their test set of 327 items, but would be too expensive for large training sets.

Dave (1991) introduced the concept of a 'noise cluster', which is designed to be more distant than correct items are to their clusters of their class, but less distant than noisy items are to clusters of their assigned class.

4 Experiments and performance measures

The rest of the paper considers the research performed by Roberts (2011) in these three areas. For automatic product classification, different classification algorithms were tried, and their performance compared both on PO data and the Reuters corpus: see Roberts et al. (2012) for more details. Eight methods for feature selection were also tested, and again tested on PO and Reuters data. For noise reduction, a novel hill climbing technique has been developed, processing the PO data: more details are in Roberts et al. (2010).

The methods are assessed using precision p, recall r and the F measure which combines them (Yang, 1999). These are given for one class or for multiple classes where they are amalgamated either as micro averaging, where these formulae are applied to the sum of all classes, or by averaging each class's measures which is known as macro-averaging. More details are in Roberts et al. (2010).

5 Automatic product classification

Of the various classification algorithms available, the ones used in this research are arguably the five most popular methods in the literature: k-nearest neighbour (Yang, 1994), Rocchio (Ittner et al., 1995), Naïve Bayes (Lewis, 1998), support vector machines (Joachims, 2001) and decision trees (Quinlan, 1993). Two trivial classifiers were also used, 'null hypothesis 1' where classes are assigned at random and 'null hypothesis 2' where the most frequently occurring class in the dataset is assigned. The detailed equations for the methods used are in Roberts (2011). Preliminary studies were carried out on subsets of the data in order to determine the best parameters for each classifier.

For Naïve Bayes, the Bernoullii event model was used as the presence or absence of a word in a document is treated as a Boolean event across all experiments. This is because repetitions of a word are rare because PO data documents are so short. As Naïve Bayes' performance improved as the prior was reduced, while it exceeded zero, prior was set to 10–11.

k-nearest neighbour worked best when k = 5. When k exceeded 5, the algorithm was poor at predicting rare classes. When it was less than 5, the classifier was poorer at generalisation.

It was found that Rocchio performed better with an equal balance between the negative and positive prototype.

Linear kernals were used for SVM, as text classification tends to be linear and the SVM took far too long to train with non-linear kernals on PO data. It performed best with squared slack variables and a trade-off between training error and margin of C = 1:0.

5.1 Main results

The five classification algorithms and two null hypotheses were compared on the PO and the Reuters data: each algorithm using the same data to make the comparisons valid. Figure 1 shows the results of this experiment, with more details in Table 1 where precision, p, recall, r, and F measures are given for the PO and Reuters datasets.

Figure 1 Performance of classifiers for the two datasets – using both the macro and micro measures (see online version for colours)

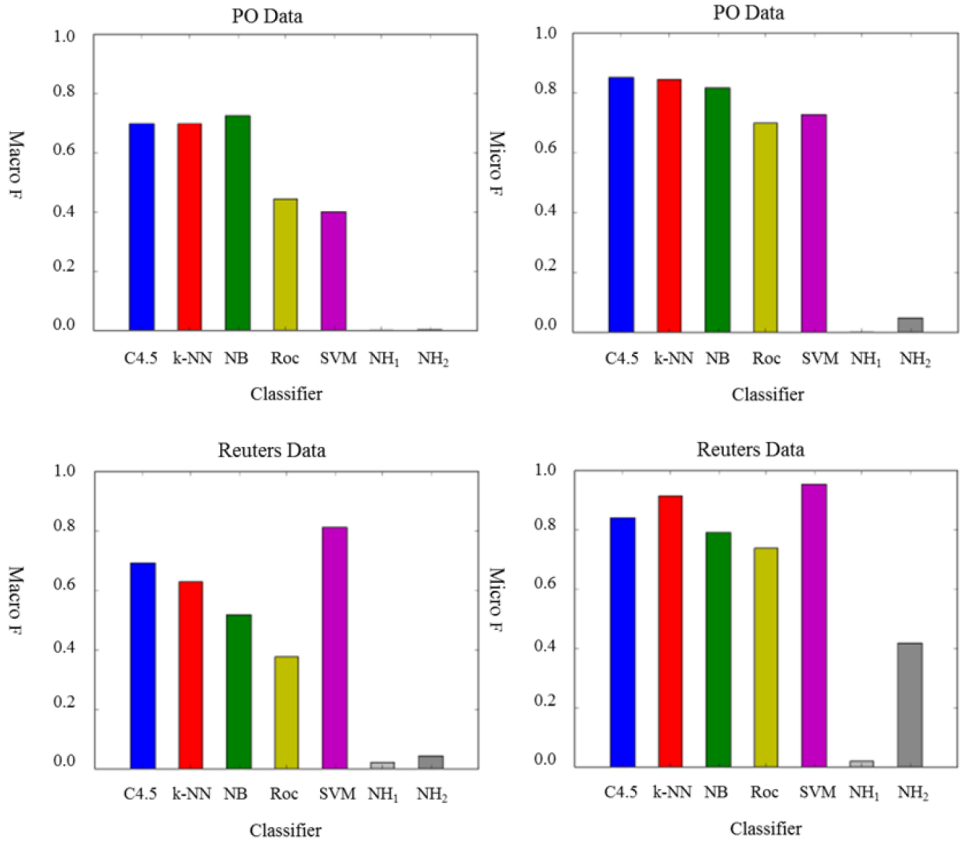


Table 1 The performance of the classifiers on the datasets

Dataset	*Classifier*	*Macro*			*Micro*			*C*
		p	*r*	*F*	*p*	*r*	*F*	
PO	C4.5	0.738	0.662	0.698	0.853	0.853	0.853	679
	k-NN	0.753	0.649	0.697	0.845	0.845	0.845	663
	Naïve Bayes	0.719	0.734	0.726	0.817	0.817	0.817	691
	Rocchio	0.604	0.350	0.443	0.712	0.689	0.700	614
	SVM	0.723	0.277	0.400	0.728	0.728	0.728	388
	Null Hyp 1	$2.8 * 10^{-3}$	$1.5 * 10^{-5}$	$3.0 * 10^{-5}$	$5.0 * 10^{-3}$	$4.6 * 10^{-6}$	$9.2 * 10^{-6}$	177
	Null Hyp 2	$2.8 * 10^{-3}$	$1.5 * 10^{-5}$	$3.0 * 10^{-5}$	$5.0 * 10^{-3}$	$4.6 * 10^{-6}$	$9.2 * 10^{-6}$	177
Reuters	C4.5	0.733	0.653	0.691	0.846	0.837	0.841	40
	k-NN	0.736	0.550	0.630	0.914	0.913	0.914	35
	Naïve Bayes	0.734	0.400	0.518	0.794	0.789	0.792	34
	Rocchio	0.852	0.243	0.378	0.737	0.737	0.737	23
	SVM	0.899	0.741	0.812	0.954	0.953	0.954	39
	Null Hyp 1	0.023	0.018	0.021	0.023	0.016	0.019	43
	Null Hyp 2	0.418	0.023	0.043	0.418	0.418	0.418	1

5.2 Discussion

Although the SVM significantly outperformed all other methods in the Reuters data, as expected from the literature (Joachims, 1998; Dumais et al., 1998), it performed poorly with the PO data. Rocchio performed badly in all experiments, which agrees with the literature (Joachims, 1997). C4.5, k-NN, and Naïve Bayes performed similarly well in the PO data, but in the Reuters data Naïve Bayes performed worse than the other two.

All classifiers performed much better than the null hypotheses. Unsurprisingly, null hypothesis 2 performed better than null hypothesis 1, especially in the micro-averaged results. Due to the large number of classes, the performance of the null hypotheses in the PO data was much lower than in the Reuters.

Skew in the datasets results in large numbers of rare classes which dominate the measures, and hence the macro measures are lower. For the PO data, the Naïve Bayes method performs best, the Rocchio and SVM the worse (apart from the null hypotheses). However, SVM is the best for the Reuters data. Except for Naïve Bayes in the PO set, all classifiers have much higher macro-averaged precisions than recalls, whereas their micro-averaged values are closer. This suggests that rarer classes have lower false positive and higher false negative results.

Significantly, although SVMs are normally recommended for text classification (and they work best for the Reuters data), they perform less well on the PO data. Roberts et al. (2012) hypothesise this may be due to class distribution or noise, and recommend further work in this area. However, the performance of classifiers can be influenced by pre-processing. The next section considers the use of feature selection for this purpose.

6 Feature selection

One recurring issue with text classification algorithms is the high dimensionality of the data. Having a dimension for every potential word in the vocabulary is expensive both for storage and for processing. Feature selection is a technique for reducing the number of words used by the classifier, and thus its complexity. Yang and Pedersen (1997) showed that due to the reduction in the risk of overfitting, rigorous feature selection can make algorithms more accurate.

In this section, eight methods of feature selection and weighting are presented. They are compared in both the PO and the Reuters datasets. The feature selection methods are also compared with a method that selects features at random. The methods used are briefly introduced below.

In document frequency thresholding (Apte et al., 1994), where frequency is the number of documents in which a word occurs, inclusion of a term is determined by comparing the frequency with some threshold. Opinions are divided as to whether classification algorithms perform better when common terms or rare terms are removed. Mendonca et al. (2001) found in the domain of medical documents that data that occur often are less important than data that occur rarely. Yang and Pedersen (1997), however, found that common words are more informative. Mendonca et al. (2001) used standardised attribute pairs rather than Yang and Pedersen's (1997), 'bag of words', so the latter's findings are more likely to be relevant to feature selection in free text.

Information gain has been used successfully as a feature selection technique for a variety of algorithms (Quinlan, 1986; Jirapech-Umpai and Aitken, 2005). It uses the conditional probability of classes given the word, to calculate the information gain the presence of the word brings. Words that have an information gain above a threshold are selected. It has been found to be very accurate, but is computationally expensive.

Mutual information measures the connection between a word and a class (Ding et al., 1997). It compares the joint probability of observing the class and word together with the probabilities of observing the class and word independently.

The χ^2 statistic (Huang, 2003) has been a successful feature selection method for a number of classification tasks (Cohen et al., 2004; Cantu-Paz et al., 2004). It measures the lack of independence between a word and a class. It is known to perform badly for infrequent terms (Dunning, 1994), which make up a large portion of most text documents.

Term strength (Yang and Pedersen, 1997) uses the assumption that the words in common between two closely related documents of the same class are more important for classification than the words that are not. It uses clustering techniques to find pairs of documents whose similarity is above a threshold, and the term strength over these documents is the estimated conditional probability of a term appearing in the second of these documents, given that it has appeared in the first.

Term frequency * inverse document frequency, TFIDF, uses the intuition that important words appear frequently in a document, but only appear in a small number of documents. This intuition is supported in Mendonca et al. (2001). It is thus the product of the term frequency within the document, and the inverse document frequency.

ConfWeight (Soucy and Mineau, 2005) replaces the inverse document frequency term of TFIDF with a term that favours the features that are proportionally more frequent in documents of one class than in documents of all other classes. It outperforms TFIDF with both k-nearest neighbour and support vector machine classifiers.

BNS (Forman, 2003) uses the idea of modelling the occurrence of a feature in a document as the event of a random normal variable exceeding a threshold. The area under the normal curve past this threshold is the prevalence of the feature. BNS measures the difference in area between when measured over documents of a class, compared to with documents not of a class. BNS has been found to perform well compared to other methods when the performance is measured by recall.

The normalised probability of word given class (Roberts, 2011), denoted here NP(ω | c), is a measure based upon the idea that a feature is useful if there exists a class in whose documents it is often found, but the total number of classes in whose documents it exists is small.

These methods were first tested to see the effect of feature selection upon the performance of a classifier – in this case the Naïve Bayes. In each case, unique features are given weights according to the method listed. A threshold, t, describes the ratio (chosen by weight) of unique features that are made available to the classifiers, and the remainder are removed. If no features remain in a document, it is assigned the 'unclassified' class and marked wrong.

6.1 Results

The macro and micro measures for the PO and Reuters data are shown in Figure 2, as the threshold t is varied.

Information gain and document frequency thresholding performed best on the PO data, followed by BNS and χ^2. χ^2 performed best in the Reuters data, followed by NP(ω | c), information gain and document frequency thresholding.

Mutual information and cTFIDF both performed worse than random, as measured by micro F, and not much better than random by macro F. ConfWeight performed better on PO than Reuters, but under-performed other methods. BNS performed well with the PO data, but less so with the Reuters. Forman (2003) found that his BNS outperformed most feature selection methods, especially in cases with large class skew.

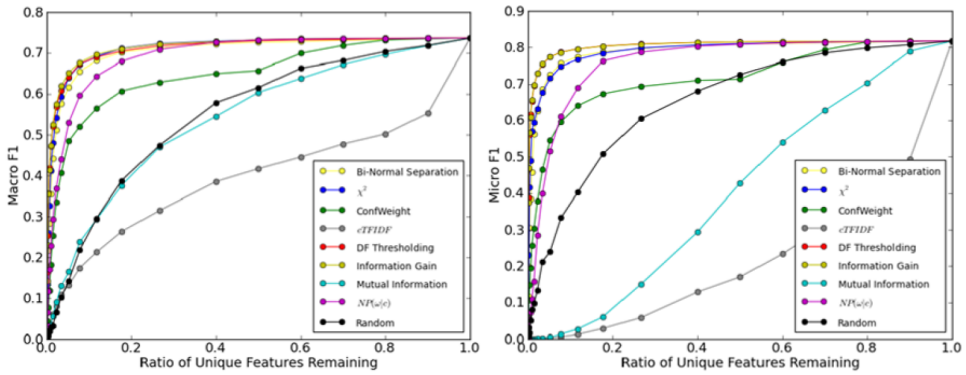
The micro results are very similar between the PO and Reuters data, but feature selection can raise the macro F measure in the Reuters, while it can only be maintained in the PO. The PO results were much smoother than the Reuters results, which is likely to be due to the larger size of the dataset.

The second tests were on feature weighting. Here every feature is made available but with weights that describe how much impact they have upon the classifier. As Naïve Bayes is probabilistic, feature weighting was tried on the Rocchio classifier. As with the feature selection experiment, where a weight calculated for a (feature, class) pair, these are aggregated into a single weight for the feature by taking a maximum.

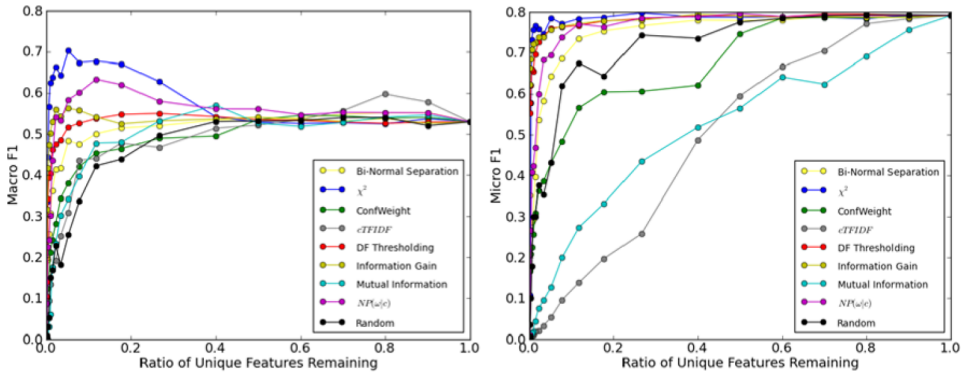
Table 2 gives the macro and micro F measures for the Rocchio classifier with different feature weighting methods for both the Reuters and PO data. The last entry in the tables can be used as a baseline. This is the performance of the classifier when all features are weighted equally. The numbers in brackets give the ratio between the result and the baseline.

In both data, document frequency and information gain both performed badly. In the PO test, the best method was cTFIDF, which raised the macro F measure by 18.3% and the micro by 5.9%. In the Reuters data, χ^2 and NP(ω | c) performed the best, raising the macro F by 2.4% and 0.6%, and the micro by 63.3% and 61.0% respectively.

Figure 2 Effect of varying threshold on macro and micro measures for (a) PO data and (b) Reuters data (see online version for colours)



(a)



(b)

Table 2 Comparison of feature weighting methods on the data

Method	*PO data*		*Reuters data*	
	Macro F	*Micro F*	*Macro F*	*Micro F*
$\chi 2$	0.4768 (1.029)	0.6593 (0.955)	0.7545 (1.024)	0.6287 (1.633)
cTFIDF	0.5479 (1.183)	0.7310 (1.059)	0.6243 (0.848)	0.491 (1.276)
Document frequency	0.3362 (0.726)	0.5581 (0.809)	0.4402 (0.598)	0.0717 (0.186)
Information gain	0.3539 (0.764)	0.5841 (0.846)	0.5915 (0.803)	0.2618 (0.680)
Mutual information	0.5199 (1.122)	0.6924 (1.003)	0.7079 (0.961)	0.5492 (1.427)
NP(ω | c)	0.5054 (1.091)	0.6839 (0.991)	0.781 (1.060)	0.6199 (1.610)
TFIDF	0.5295 (1.143)	0.6949 (1.007)	0.6402 (0.869)	0.4674 (1.214)
Uniform	0.4633 (1.000)	0.6902 (1.000)	0.7365 (1.000)	0.3849 (1.000)

6.2 Discussion

While in the Reuters control set performance can be increased with feature selection, the results show that it can merely be maintained in the PO data. This makes intuitive sense, as there is an order of magnitude more features in the average document in the Reuters data than in the PO. Therefore, features in the PO data are much more likely to be pertinent.

In PO data, therefore, there is little advantage to feature selection where the dimensionality of the data is not limiting. Where the dimensionality is a problem, some 80% of unique features can be removed without much loss to performance.

This work agrees with Yang and Pederson (1997) that mutual information does not compare well with other methods. The similarity between the results for information gain and document frequency thresholding in the PO data supports their findings that they are closely correlated. As document frequency thresholding is much simpler to calculate than information gain or χ^2, this could well be the best method of feature selection in this domain.

Notably, methods that are good at feature selection with Naïve Bayes are not necessarily good at feature weighting with Rocchio. Both information gain and document frequency feature weighting caused Rocchio to perform badly. However, care should be taken when comparing the results of feature selection with those of feature weighting. They are not directly comparable as they are being used by different classification algorithms.

The best method for weighting features on the PO data is cTFIDF, which supports common wisdom that TFIDF-related methods are best for weighting features.

7 Noise reduction

The issue considered finally is that of the reduction of label noise in the data used to train classifiers. This occurs when documents in the training or testing set are erroneously or inconsistently labelled into potentially ambiguous and overlapping classes. Occasional errors can be detected as outliers. The method described here addresses the cases where there are systematic errors, which has happened with the PO data which have been hand classified by different people.

The approach taken [described in full in Roberts (2011)] aims to identify pairs of classes in the training documents where there is insufficient information to distinguish them in the classification stage. Sometimes further labelled data can be sought or training data for the classes can be corrected to resolve the classes. If not, then the two classes can be merged together.

In many cases, these merged class pairs will be more useful than they were individually as there is more confidence in the labels, and the schema will contain fewer arbitrary choices for future manual classifiers. As an example, a number of classes of different types of medical stent were generated which subsequently were able to be merged into one class.

Decisions on whether to merge classes are based on information measures. For two variables X and Y, the entropies H(X), H(Y) and H(X, Y) are found together with the information content between X and Y as measured by their mutual information, I(X, Y). The performance of a classifier is then I(X, Y) normalised over H(X, Y) to give a value

J(X,Y). The relevant equations and approach is described in detail in Roberts et al. (2010).

The noise reduction process is then achieved by splitting the training data into a training set and a validation set. A classifier is trained on the former and predicts classes in the latter. For each pair of classes, X, Y, the change in J(X, Y) which would occur if X and Y were merged is found. The pairs for which this change is the greatest are then found using hill climbing and, assuming the change is positive. The information measures are updated and the process continues until there is no benefit to merging the classes.

The hill climbing process is made more robust by splitting the training data into n equal subsets (similar to n-fold cross validation). The n classifiers are each trained on the other n – 1 subsets. Roberts shows that this method is robust. Here n = 10.

7.1 Results

A control classifier was trained and tested first. Then hill climbing was run on five occasions, each time splitting the training set randomly into ten equal subsets, but using different combinations. For each the change in H(X, Y), I(X, Y) and J(X, Y) were measured. Finally, single hill-climbing was performed on the complete dataset, known as the 'complete' run. Here the complete training set is used to train the classifier, and the X and Y distributions are taken from the actual and predicted labels in the test set. In normal use, of course, this would be impossible as the actual labels of the test set are unknown. The lists of merges produced by each experiment were compared to assess the robustness of the method. Finally, the results of manual inspection of the actual merges are given.

The combined hill climbing algorithm produces a ranked list of suggested merges. There were 104, 101, 108, 105 and 104 merges suggested on the five runs of combined hill climbing, respectively, with a high level of consistency between the five runs. 82.8% of unique class pairs were suggested in all five runs. 96.1% of the distinct suggested merges ranked 1–50 were suggested in all five runs. There were 116 unique merges suggested across all five runs. The suggested merges ranked 1–10 were identical in each run.

For comparison, hill climbing was performed separately on each of the ten splits for one run. The results were far less consistent than in the combined run, giving 165, 156, 165, 171, 173, 156, 162, 161, 150 and 162 class pairs, respectively. All ten put the same class pair at the top of the list, but they deviated on the second. One class pair was put in ninth or tenth place in three of the splits, and not included in the other seven's lists at all. There were a total of 712 unique class pairs over all lists.

By comparing this with the consistency between the five combined hill climbing runs, it can be concluded that the combined hill climbing is very robust. Table 3 shows the values of H(X, Y), I(X, Y), and J(X, Y) for the control, for each of the five hill climbing runs, and for the complete run.

The different hill climbing runs give very similar results, being 96.7% of the way between the control and the complete run. The combined hill climbing using a validation set gets very close to the information levels that can be obtained from the entire dataset.

The precision, recall and F measures for macro- and micro averaging are shown in Table 4. The results are given for the control, and then for the hill climbing and then one can see what they would be if the identified classes were merged.

Table 3 The entropy and information, before and after hill climbing

	H(X, Y)	*I(X, Y)*	*J(X, Y)*
Control	5.70199	4.40894	0.77322
Hill climbing 1	5.52451	4.33484	0.78466
Hill climbing 2	5.52444	4.33479	0.78466
Hill climbing 3	5.52432	4.33463	0.78465
Hill climbing 4	5.52439	4.33477	0.78466
Hill climbing 5	5.52441	4.33480	0.78466
Complete	5.51681	4.33099	0.78505

Table 4 Macro and micro averaged precision, recall and F values

	Macro averaging			*Micro averaging*		
	Precision	*Recall*	*F*	*Precision*	*Recall*	*F*
Control	0.7190	0.7338	0.7263	0.8172	0.8170	0.8171
Hill climbing 1	0.7779	0.7690	0.7735	0.8828	0.8826	0.8827
Hill climbing 2	0.7771	0.7661	0.7716	0.8828	0.8826	0.8827
Hill climbing 3	0.7797	0.7690	0.7743	0.8828	0.8826	0.8827
Hill climbing 4	0.7779	0.7680	0.7729	0.8828	0.8826	0.8827
Hill climbing 5	0.7767	0.7681	0.7724	0.8828	0.8826	0.8827
Complete	0.8058	0.7812	0.7933	0.8830	0.8829	0.8829

It is clear from the information and performance measures in Tables 3 and 4 that the best result is achieved by the 'complete' process.

Table 5 Suggested classes to merge

WAP (computer consumables)	WAT (laser printer consumables and cartridges)
FDC (bag and mask)	FXU (respiratory items Laryngeal mask)
MKC (cleaning cloths dishcloths and dusters)	VEP (tissues and medical wipes)
KBB (laboratory and pathology apparatus)	KBD (laboratory and pathology – other consumables)
PMP (general electrical equipment dryers heaters radiators amplifiers)	PNB (batteries and battery chargers)
EHQ (elastic adhesive)	EIH (plaster elastic fabric)
BCZ (workwear protective and chemical protection)	BYC (coveralls – laboratory wear)
ERF (shoes – podiatry products)	GOX (footwear adaptation maintenance and repair)
HAC (immunoassay)	HBB (analytical kits)
GFE (audiometers tympanometers test boxes REM)	GFS (hearing aids spares)
AFD (chilled pork)	AFM (processed pork ham bacon and sausages)
BWD (dresses – theatre wear)	WCN (audiovisual equipment)
TUK (pedestals)	TKS (cupboards, stationery storage)

Some of the suggested merges from the ranked lists are shown in Table 5. The first five are examples of the 71 classes to merge which were at the top of each ranked list. The next three were suggested by all runs, but it varied as to where they were in the lists. The next three were merges suggested by only some of the runs. All of these seem to make sense. The last two, however, look wrong, but can be explained. For the first, the 30 training items labelled as BWD are in fact mislabelled electronic headsets. As regards the pedestals, while there are 261 items classified (mostly accurately) as pedestals, five out of the ten items classified to TKS are actually pedestals. These are examples of systematic noise affecting an entire class that hill climbing can discover.

8 Further work

The methods outlined above are used successfully in the SpendInsight and GreenInsight systems. However there is scope for further work on the algorithms. Cross-validation is often used in classifier design – it may be of interest to see if its inclusion could improve performance. Further work on data processing and normalisation could be investigated as these can be significant. A comparison of classifiers as applied to the different feature selection methods may also be of benefit.

9 Conclusions

Successful automatic classification has been demonstrated on purchase order data, where the best methods to use are k-nearest neighbour and Naïve Bayes, not a support vector machine, usually used for text classification. For the PO data, there are relatively few features, so feature selection does not improve performance significantly here. The PO data suffers from systematic label noise, but the novel hill climbing method described here is shown to robustly improve the classification system. The research has contributed significantly to an e-procurement system which has allowed for significant savings for NHS trusts.

References

Agarwal, S., Godbole, S., Punjani, D. and Roy, S. (2007) 'How much noise is too much: a study in automatic text classification', *Proc ICDM-07, the 7th IEEE International Conference on Data Mining*, pp.3–12.

Aggarwal, C.C. and Zhai, C. (2012) 'A survey of text clustering algorithms', *Mining Text Data*, pp.163–222, doi 10.1007/978-1-4614-3223-4_6, Springer-Verlag.

Apte, C., Damerau, F. and Weiss, S.M. (1994) 'Automated learning of decision rules for text categorization', *Information Systems*, Vol. 12, No. 3, pp.233–251.

Brodley, C.E. and Friedl, M.A. (1996) 'Identifying and eliminating mislabelled training instances', *Proceedings of AAAI-96, the 13th National Conference on Artificial Intelligence*, pp.799–805.

Cantu-Paz, E., Newsam, S. and Kamath, C. (2004) 'Feature selection in scientific applications', *Proceedings of KDDD-04, the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.788–793, New York, USA.

Cohen, A.M., Bhupatiraju, R.T. and Hersh, W.R. (2004) 'Feature generation, feature selection, classifiers, and conceptual drift for biomedical document triage', *Proceedings of TREC-04, the 13th Text Retrieval Conference*.

Dave, R.N. (1991) 'Characterization and detection of noise in clustering', *Pattern Recognition Letters*, Vol. 12, No. 11, pp.657–664.

Daza, L. and Acuna, E. (2007) 'An algorithm for detecting noise on supervised classification', *Proceedings of WCECS-07, the 1st World Conference on Engineering and Computer Science*, pp.701–706.

Ding, Y., Korotkiy, M., Omelayenko, B., Kartseva, V., Zykov, V., Klein, M., Schulten, E. and Fensel, D. (2002) 'Goldenbullet. Automated classification of product data in e-commerce', *Proc. BIS-02*, Poznan, pp.1–9.

Dumais, S.T., Platt, J., Heckerman, D. and Sahami, M. (1998) 'Inductive learning algorithms and representations for text categorization', *Proc. CIKM-98*, pp.148–155.

Dunning, T. (1994) 'Accurate methods for the statistics of surprise and coincidence', *Computational Linguistics*, Vol. 19, No. 1, pp.61–74.

Fensel, D., Ding, Y., Omelayenko, B., Schulten, E., Botquin, G., Brown, M. and Flett, A. (2001) 'Product data integration in b2b e-commerce', *IEEE Intelligent Systems*, Vol. 16, No. 4, pp.54–59.

Forman, G. (2003) 'An extensive empirical study of feature selection metrics for text classification', *Journal of Machine Learning Research*, March, Vol. 3, pp.1289–1305.

Gamberger, D., Lavrac, N. and Groselj, C. (1999) 'Experiments with noise filtering in a medical domain', *Proceedings of ICML-99, the 16th International Conference on Machine Learning*, pp.143–151.

GreenInsight (2012) [online] http://www.green-insight.com (accessed 8 August 2012).

Hepp, M., Leukel, J. and Schmitz, V. (2005) 'A quantitative analysis of eCl@ss, UNSPSC, eOTD, and RNTD content, coverage and maintenance', *Proc. ICEBE-05*, pp.572–581.

Huang, S.H. (2003) 'Dimensionality reduction in automatic knowledge acquisition: a simple greedy search approach', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 15, No. 6, pp.1364–1373.

Ittner, D.J., Lewis, D.D. and Ahn, D.D. (1995) 'Text categorization of low quality images', *Proc. SDAIR-95*, pp.301–315.

Jirapech-Umpai, T. and Aitken, S. (2005) 'Feature selection and classification for microarray data analysis – evolutionary methods for identifying predictive genes', *BMC Bioinformatics*, Vol. 6, No. 148, 11p, DOIs: http://dx.doi.org/10.1186/1471-2105-6-148.

Joachims, T. (1997) 'A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization', *Proc. ICML-97*, pp.143–151.

Joachims, T. (1998) 'Text categorization with support vector machines: learning with many relevant features', *Proc. ECML-98*, pp.137–142.

Joachims, T. (2001) *Learning to Classify Text Using Support Vector Machines*, Kluwer Academic Publishers, Norwell, MA.

Lewis, D.D. (1998) 'Naïve (Bayes) at forty: the independence assumption in information retrieval', *Proc. ECML-98*, pp.4–15.

Li, S., Xia, R., Zong, C. and Huang, C. (2001) 'A framework for feature selection methods for text categorization', *Proceedings of ACL-09, the 47th Annual Meeting of the Association for Computational Linguistics*, pp.692–700.

Liu, H. and Yu, L. (2005) 'Toward integrating feature selection algorithms for classification and clustering', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 17, No. 4, pp.491–502.

Mendonca, E.A., Cimino, J.J. and Johnson, S.B. (2001) 'Using narrative reports to support a digital library', *Journal of the American Medical Informatics Association*, Vol. 8.

National Audit Office (2011) *The Procurement of Consumables by NHS Hospital Trusts* [online] http://www.nao.org.uk/publications/1011/nhs_procurement.aspx (accessed 16 July 2012).

Quinlan, J.R. (1986) 'Induction of decision trees', *Machine Learning*, Vol. 1, No. 1, pp.81–106.

Quinlan, J.R. (1993) *C4.5: Programs for Machine Learning*, Morgan Kaufmann, San Francisco, CA, USA.

Ramakrishnan, G., Chitrapura, K.P., Krishnapuram, R. and Bhattacharyya, P. (2005) 'A model for handling approximate, noisy or incomplete labeling in text classification', *Proceedings of ICML-05, the 22nd International Conference on Machine Learning*, pp.681–688.

Roberts, P.J. (2011) *Automatic Product Classification*, PhD thesis, University of Reading, UK.

Roberts, P.J., Howroyd, J., Mitchell, R.J. and Ruiz. V.F. (2010) 'Identifying problematic classes in text classification', *Proc. CIS2010*, pp.136–141.

Roberts, P.J., Mitchell, R.J., Ruiz, V.F. and Bishop, J.M. (2012) 'Classification in e-procurement', *Proc CIS2012*, Limerick, pp.1–6.

Soucy, P. and Mineau, G.W. (2005) 'Beyond TFIDF weighting for text categorization in the vector space model', *Proc of IJCAI-05, the 19th International Joint Conference on Artificial Intelligence*, pp.1130–1135.

SpendInsight (2012) [online] http://www.spendinsight.com (accessed 8 August 2012).

Verbaeten, S. and van Assche, A. (2003) 'Ensemble methods for noise elimination in classification problems', in Windeatt, T. and Roli, F. (Eds.): *Multiple Classifier Systems*, Vol. 2709, Lecture Notes in Computer Science, pp.317–325.

Wolin, B. (2002) 'Automatic classification in product catalogs', *Proc. SIGIR-02*, pp.351–352.

Yang, Y. (1994) 'Expert network: effective and efficient learning from human decisions in text categorization and retrieval', *Proc. SIGIR-94*, pp.13–22.

Yang, Y. (1999) 'An evaluation of statistical approaches to text categorization', *Information Retrieval*, Vol. 1, No. 1, pp.69–90.

Yang, Y. and Pedersen, J.O. (1997) 'A comparative study on feature selection in text categorization', *Proc ICML-97, the 14th International Conference on Machine Learning*, pp.412–420, Nashville, USA.

Zheng, Z., Wu, X. and Srihari, R. (2004) 'Feature selection for text categorization on imbalanced data', *ACM SIGKDD Explorations Newsletter*, Vol. 6, No. 1, pp.80–89.