Categorical Proportional Difference: A Feature Selection Method for Text Categorization

Mondelle Simeon **Robert Hilderman**

Department of Computer Science
University of Regina
Regina, Saskatchewan, Canada S4S 0A2
Email: {simeon2m, hilder}@cs.uregina.ca

Abstract

Supervised text categorization is a machine learning task where a predefined category label is automatically assigned to a previously unlabelled document based upon characteristics of the words contained in the document. Since the number of unique words in a learning task (i.e., the number of features) can be very large, the efficiency and accuracy of the learning task can be increased by using feature selection methods to extract from a document a subset of the features that are considered most relevant. In this paper, we introduce a new feature selection method called categorical proportional difference (CPD), a measure of the degree to which a word contributes to differentiating a particular category from other categories. The CPD for a word in a particular category in a text corpus is a ratio that considers the number of documents of a category in which the word occurs and the number of documents from other categories in which the word also occurs. We conducted a series of experiments to evaluate CPD when used in conjunction with SVM and Naive Bayes text classifiers on the OHSUMED, 20 Newsgroups, and Reuters-21578 text corpora. Recall, precision, and the F-measure were used as the measures of performance. The results obtained using CPD were compared to those obtained using six common feature selection methods found in the literature: χ^2, information gain, document frequency, mutual information, odds ratio, and simplified χ^2. Empirical results showed that, in general, according to the F-measure, CPD outperforms the other feature selection methods in four out of six text categorization tasks.

Keywords: Text categorization, feature selection, supervised learning, categorical proportional difference.

1 Introduction

Due to the consistent and rapid growth of unstructured textual data that is available online, text categorization, the machine learning task of automatically assigning a predefined category label to a previously unlabelled document, is essential for handling and organizing this data. Widely used and well studied text categorization methods include Naive Bayes (Kim et al., 2006), support vector machines (SVM) (Joachims, 1998), and k-nearest neighbor (k-NN) (Han et al., 2001) methods. All of these methods use a collection of pre-labelled examples to build a predictive model for each distinct category contained in the examples. For a survey of these and other automated text categorization methods and applications, see (Sebastiani, 2002).

If the number of unique words (i.e., the number of features) encountered by a text categorization task is large, the efficiency and accuracy of the method may be adversely affected. For example, accuracy can be reduced if a method is used where features with low predictive value are included in the model, and efficiency can be improved if a method is used where less computation and/or memory is required to categorize a given text corpus (Forman, 2008). As a result, feature selection methods are used to address efficiency and accuracy by extracting from a document a subset of the features that are considered most relevant. When using a feature selection method, each word is scored using some predefined measure and the most relevant words are selected based upon this measure.

1.1 Related Work

Many feature selection methods have been proposed and studied by various authors. In (Yang and Pedersen, 1997), document frequency, two information gain methods, mutual information, term strength, and three chi-square methods are evaluated on the Reuters and OHSUMED text corpora using k-nearest neighbor and linear least squares fitting classifiers. Here it is suggested that document frequency, one of the information gain methods, and one of the chi-square methods are the most effective feature selection methods, with strong correlations found between the results obtained using these methods.

The odds ratio was proposed as a feature selection method and compared to a variety of other feature selection methods in (Mladenic and Grobelnik, 1999). Here the evaluation was somewhat limited as only a multinomial Naive Bayes text classifier was used and only on the Reuters text corpus. They did find that their proposed method performed best. However, their results disagreed with the conclusions in (Yang and Pedersen, 1997) in regard to the strength of IG as a feature selection method. They attributed this discrepancy to differences in domain definitions and the classifiers used.

In (Galavotti et al., 2000), a simplified chi-square feature selection method was proposed and compared to the original chi-square method. Again, the evaluation was somewhat limited as only different variants of a k-NN classifier were used and only on the Reuters text corpus. They did find, though, that their proposed method outperformed two chi-square feature selection methods under conditions that would be characterized as extremely aggressive feature se-

lection.

In (Ng et al., 1997), a correlation coefficient based variant of the chi-square feature selection method, was compared to a chi-square feature selection method. In this study, a perception learning classifier and the Reuters text corpus were used. They found that their method outperformed other methods.

A group of scoring measures for feature selection were proposed in (Montanes et al., 2005). The measures were evaluated using an SVM classifier on the Reuters and OHSUMED text corpora. Here it was found that results were mixed with their scoring measures outperforming both information gain and TF*IDF in some situations.

In (Forman, 2003), a study of the effects of various feature selection methods on an SVM text classifier is described. Here, an evaluation methodology is proposed for determining the feature selection method or methods that are most likely to provide the best results. In addition, a new feature selection method, called bi-normal separation, is shown to outperform other commonly known methods in some circumstances.

The collected results from various feature selection studies are described in (Sebastiani, 2002). Here, some general recommendations are made regarding the relative performance of numerous feature selection methods. However, it is suggested that in order to make more conclusive statements on the relative performance of the feature selection methods studied, that comparative experiments under controlled conditions using a variety of text corpora and classifiers are required.

1.2 Our Contribution

In this paper, we introduce a new feature selection method called categorical proportional difference (CPD), a measure of the degree to which a word contributes to differentiating a particular category from other categories in a text corpus. The CPD for a word in a particular category is a ratio that considers the number of documents of the category in which the word occurs and the number of documents from other categories in which the word also occurs. We conducted a series of experiments to evaluate CPD when used in conjunction with SVM and Naive Bayes classifiers on the OHSUMED, 20 Newsgroups, and Reuters-21578 text corpora. The F-measure, a measure that combines both recall and precision, was used as the measure of performance. The results obtained using CPD were compared to those obtained using six feature selection methods commonly found in the literature: χ^2 (Yang and Pedersen, 1997), information gain (Yang and Pedersen, 1997), document frequency (Yang and Pedersen, 1997), mutual information (Yang and Pedersen, 1997), odds ratio (Mladenic and Grobelnik, 1999), and simplified-χ^2 (Galavotti et al., 2000). Empirical results showed that, in general, according to the F-measure, CPD is an effective feature selection method, outperforming the other feature selection methods in four out of six text categorization tasks.

The remainder of this paper is organized as follows. In Section 2, we introduce the CPD feature selection measure. In Section 3, we provide an overview of relevant details regarding our methodological approach to evaluating CPD. In Section 4, we present the experimental results from a comprehensive series of text categorization tasks. We conclude in Section 5 with a summary of our results and suggestions for future work.

2 A New Method for Feature Selection

In this section, we introduce the CPD feature selection measure and provide a brief example to demonstrate its use on a sample text corpus.

2.1 Categorical Proportional Difference

We define CPD (and the other feature selection methods described in Section 3.2) in reference to a typical 2×2 contingency table. A example contingency table is shown in Table 1, where A is the number of times word w and category c occur together, B is the number of times word w occurs without category c, C is the number of times category c occurs without word w, D is the number of times neither word w nor category c occur, and $N = A + B + C + D$.

Table 1: An example contingency table

	c	$\neg c$	Σ *Row*
w	A	B	$A+B$
$\neg w$	C	D	$C+D$
Σ *Column*	$A+C$	$B+D$	N

CPD measures the degree to which a word contributes to differentiating a particular category from other categories in a text corpus. The possible values for CPD are restricted to the interval $(-1, 1]$, where values near –1 indicate that a word occurs in approximately an equal number of documents in all categories (it approaches –1 as the number of categories increases) and a 1 indicates that a word occurs in the documents of only one category. More formally, the categorical proportional difference for word w in category c is defined as

$$\text{CPD}(w, c) \quad = \quad \frac{A - B}{A + B}.$$

That is, the calculation is simply the ratio of the difference between the number of documents of a category in which a word occurs and the number of documents of other categories in which the word also occurs, divided by the total number of documents in which the word occurs. The CPD for a word is the ratio associated with the category c_i for which the value is greatest. That is,

$$\text{CPD}(w) = \max_i\{\text{CPD}(w, c_i)\}.$$

2.2 Example

Words and their frequency of occurrence in labelled documents from a sample text corpus are shown in Table 2. In Table 2, the *Word* column contains a subset of the words occurring in the documents, the *No. in Grain*, *No. in Trade*, *No. in Interest*, and *No. in Agriculture* columns contain the number of documents in which a word occurs that have been assigned the corresponding category label, the *No. of Documents* column contains the total number of documents in which a word occurs, and the *CPD* column contains the value calculated for a word using the CPD feature selection measure.

For example, consider the word "wheat", where all occurrences of the word are in documents of the *Grain* category. Here, CPD(wheat, *Grain*) = (25 – 0) / (25 + 0) = 1, and CPD(wheat, {*Trade*, *Interest*, *Agriculture*}) = (0 – 25) / (0 + 25) = –1. Thus, CPD(wheat) = max {1, –1, –1, –1} = 1. Now consider the word "economy", where the word occurs in the same number of documents in each category. Here we have

Table 2: Word distribution and CPD in a sample text corpus

Word	*No. in Grain*	*No. in Trade*	*No. in Interest*	*No. in Agriculture*	*No. of Documents*	*CPD*
wheat	25	0	0	0	25	1.00
economy	15	15	15	15	60	–0.50
surplus	18	5	0	2	25	0.44
quotas	1	50	1	1	53	0.89
feed	7	9	4	11	31	–0.29

CPD(economy, {*Grain*, *Trade*, *Interest*, *Agriculture*}) = (15 – 45) / (15 + 45) = –0.5 and CPD(economy) = –0.5. Finally, consider the word "surplus". Here CPD(surplus, *Grain*) = 11 / 25 = 0.44, CPD(surplus, *Trade*) = –15 / 25 = –0.6, CPD(surplus, *Interest*) = –25 / 25 = –1.0, CPD(surplus, *Agriculture*) = –21 / 25 = –0.84 and CPD(surplus) = 0.44.

3 Methodological Overview

In this section, we describe relevant details and issues related to the underlying methodological approach used to obtain the experimental results.

3.1 Text Classifiers

Text categorization in this, and other work, is essentially a two-step process. In the first step, a category model for each category in a text corpus of labelled training documents is built. In the second step, a text categorization algorithm compares an unlabelled document to the learned category models to determine the "best" category label to assign to the unlabelled document. In this work, we used the SVM and Naive Bayes classifiers provided in the Weka collection of machine learning algorithms (Witten and Frank, 2005) to generate all the experimental results.

3.2 Feature Selection Methods

Seven feature selection methods were used in conjunction with the SVM and Naive Bayes classifiers: CPD, χ^2, information gain (IG), document frequency (DF), mutual information (MI), odds ratio (OR), and simplified-χ^2 (S-χ^2). CPD was previously defined in Section 2.1, and the variables A, B, C, D, and N used below are the same as those described in that section.

χ^2 measures the lack of independence between a word w and a category c if it is assumed that the occurrence of a word is actually independent from the category label. For a word w and a category c,

$$\chi^2(w,c) = \frac{N(AD-BC)^2}{(A+C)(B+D)(A+B)(C+D)}.$$

After χ^2 for every combination of the word w and category c is determined, we take the maximum χ^2 value over all the categories c_i as the χ^2 value for the word. That is,

$$\chi^2(w) = \max_i\{\chi^2(w,c_i)\}.$$

IG measures the decrease in entropy when a selected feature is present versus when it is absent. For a word w and a category c,

$$\begin{aligned}\mathrm{IG}(w,c) = \\ & e(POS,NEG) - [p(w)e(TP,FP) + \\ & p(\neg w)e(FN,TN)],\end{aligned}$$

where

$$e(x,y) = -\frac{x}{x+y}\log_2\frac{x}{x+y} - \frac{y}{x+y}\log_2\frac{y}{x+y},$$

$$POS = A+C,$$

$$NEG = B+D,$$

$$p(w) = \frac{A+B}{N},$$

$$p(\neg w) = 1-p(w),$$

and TP, FP, TN, and FN are the number of true positives, false positives, true negatives, and false negatives, respectively. We take the sum of the IG values for a word over all the categories c_i as the IG value for the word. That is,

$$\mathrm{IG}(w) = \sum_i \mathrm{IG}(w,c_i).$$

DF simply measures the number of documents in which a word occurs and is determined without reference to category labels. So,

$$\mathrm{DF} = A+B.$$

MI measures the mutual dependence of a word w and a category c. For a word w and a category c,

$$\mathrm{MI}(w,c) = \log\frac{AN}{(A+B)(A+C)}.$$

We take the maximum of the MI values for a word over all the categories c_i as the MI value for the word. That is,

$$\mathrm{MI}(w) = \max_i\{\mathrm{MI}(w,c_i)\}.$$

OR measures the odds of a word occurring in the positive class normalized by that of the negative class. To avoid a situation where division by zero may occur, one is added to any zero in the denominator. For a word w and a category c,

$$\mathrm{OR}(w,c) = \frac{AD}{BC}.$$

We take the sum of the OR values over all categories c_i as the OR value for the word. That is,

$$\mathrm{OR}(w) = \sum_i \mathrm{OR}(w,c_i).$$

S-χ^2 is a variant of χ^2. Positive values are indicative of membership of a word w in a category c, and negative values are indicative of non-membership. For a word w and a category c,

$$\text{S-}\chi^2(w,c) = \frac{AD-BC}{N^2}.$$

We take the maximum S-χ^2 value over all categories c_i as the S-χ^2 value for the word. That is,

$$\text{S-}\chi^2(w) = \max_i\{\text{S-}\chi^2(w,c_i)\}.$$

Table 3: Summary statistics for the randomly generated subset datasets

Description	*#/%*	*Statistic*	*OHSUMED*	*20 Newsgroups*	*Reuters-21578*
Possible Categories	#		15	16	14
Categories/Dataset	#	max	3	3	5
	#	min	2	2	4
	#	mean	2.4	2.3	4.6
Documents/Dataset	#	max	1200	1200	1120
	#	min	800	800	921
	#	mean	960	920	1069
Words/Document	#	max	60	102	65
	#	min	53	52	58
	#	mean	57	76	61
Features/Dataset	#	max	8327	16923	6737
	#	min	6403	8005	4919
	#	mean	7239	11601	5959
Words in Single Category	%	max	66	77	54
	%	min	53	69	45
	%	mean	59	73	50
Categories/Word	#	max	3	3	5
	#	min	1	1	1
	#	mean	1.5	1.3	2.1

3.3 Performance Measures

To evaluate the utility of the various feature selection methods used, we use the F-measure, a measure that combines precision and recall, two commonly used measures of text categorization performance. Precision (P) measures the percentage of documents assigned to category c that are correctly assigned to category c, and recall (R) measures the percentage of documents that should have been assigned to category c that actually were assigned to category c. More formally,

$$P_i = \frac{TP_i}{TP_i + FP_i}$$

and

$$R_i = \frac{TP_i}{TP_i + FN_i},$$

where TP_i (i.e., true positives) is the number of documents assigned correctly to category c_i, FP_i (i.e., false positives) is the number of documents assigned to category c_i that should have been assigned to other categories, and FN_i (i.e., false negatives) is the number of documents assigned to other categories that should have been assigned to category c_i. The F-measure (F) is the harmonic average of precision and recall, and is defined as

$$F_i = \frac{2P_iR_i}{P_i + R_i},$$

where P_i and R_i are the precision and recall, respectively, for category c_i. After the F-measure is determined for each category c_i, the macro-average (i.e., the traditional arithmetic mean) of these values is determined, and this value becomes the overall F-measure. That is,

$$\textit{average maximum } F = \frac{\sum_i F_i}{n},$$

where n is the number of categories.

3.4 Text Corpora

In previous studies on text categorization and feature selection, it is common for a new method or measure to be evaluated against frequently used text corpora such as OHSUMED (OHSUMED, 2005), 20 Newsgroups (Newsgroups, 1999), and Reuters-21578 (Reuters-21578, 1997). The OHSUMED text corpus is a subset of the MEDLINE database containing 348,588 abstracts from 270 medical journals for the five years from 1987 to 1991. The 20 Newsgroups text corpus is a set of 20,000 Usenet articles. The Reuters-21578 text corpus is a set of economic news stories published in 1987.

In this work, we also use the OHSUMED, 20 Newsgroups, and Reuters-21578 text corpora, but we use them only as repositories from which to randomly generate many subset datasets having different characteristics. For example, from OHSUMED, we use only the first 20,000 abstracts from 1991, where each document is labelled with one of 23 cardiovascular disease categories. From these documents, ten unique subset datasets were randomly generated, each containing from 800 to 1200 documents. From 20 Newsgroups, where each article is labelled with one of 20 categories, ten unique subset datasets were randomly generated in the same way as for OHSUMED. And from Reuters-21578, where each document is labelled with one of 90 categories, we use the ModAptè split which contains 12,902 documents (9,603 in the training set and 3,299 in the test set). The documents in the training and test sets were combined into one document collection and those documents from categories that contained fewer than 100 documents were discarded. From the remaining documents, ten unique subset datasets were randomly generated, each containing from 921 to 1120 documents.

Summary statistics for the randomly generated subset datasets from the three text corpora are shown in Table 3. In Table 3, the *#/%* column describes either a count (i.e., *#*) or a percentage (i.e., *%*), the *Statistic* column describes the maximum, minimum, or arithmetic mean of the corresponding count or percentage, and the *OHSUMED*, *20 Newsgroups*, and *Reuters-21578* columns describe the values of the corresponding measures. For example, for the ten randomly generated Reuters-21578 datasets, the maximum, minimum, and arithmetic mean for the number of features per dataset (i.e., Features/Dataset), is 6737, 4919, and 5959, respectively.

3.5 Our Approach

Given some collection of text corpora and some collection of feature selection methods, the algorithm shown in Figure 1 describes the steps followed in our approach to generating the experimental results. In Figure 1, the algorithm consists of two phases.

In the first phase, the document pre-processing phase (lines 3 to 20), a word is compared against a list of common stop words, and if it is determined to be a stop word, it is discarded (lines 3 to 5). Punctuation

```
1: for each each text corpus do
2:    for each dataset randomly generated from the text corpus do
3:       for each document in the dataset do
4:          Remove stop-words, punctuation, and non-alphanumeric text
5:       end for
6:       for each remaining word in the dataset do
7:          Stem the word (a Porter stemmer was used)
8:          Store each unique stemmed word in the word list
9:       end for
10:      for each word in the word list do
11:         Determine TF (i.e., term frequency)
12:         Store TF in the corresponding element of the weight matrix
13:         Determine IDF (i.e., inverse document frequency)
14:         if IDF == zero then
15:            Remove the word from the word list
16:            Remove the corresponding TF from the weight matrix
17:         else
18:            Store normalized TF*IDF in the corresponding element of the weight matrix
19:         end if
20:      end for
21:      for  each classifier do
22:         Create a Weka model using the word list and weight matrix
23:         Perform ten-fold cross-validation
24:         Store the F-Measure
25:         for each feature selection method do
26:            for each word in the word list do
27:               Score each word according to the feature selection method
28:               Store each unique score in the score list
29:            end for
30:            Sort the score list in ascending order
31:            Set the maximum F-measure to zero
32:            for each score in the score list from smallest to largest do
33:               Use the current score as the cutoff score
34:               for each word in the word list whose score <= the cutoff score do
35:                  Remove the word from the word list
36:                  Remove the corresponding TF*IDF from the weight matrix
37:               end for
38:               Create a Weka model using the word list and weight matrix
39:               Perform ten-fold cross-validation
40:               if the F-measure > the maximum F-measure then
41:                  Set the maximum F-measure to F-measure
42:               end if
43:            end for
44:         end for
45:      end for
46:   end for
47:   Determine the average maximum F-measure over all the datasets
48: end for
```

Figure 1: The steps followed in our approach to generating the experimental results

and non-alphanumeric text is also discarded. The remaining words are stemmed and the unique words are stored in a word list (lines 6 to 9). Then an $m \times n$ weight matrix is built for the current dataset (lines 10 to 20), where m is the number of documents in the dataset and n is the number of words in the feature space. Each word is associated with a particular column in the matrix and the element at the intersection of a row and column is the normalized TF*IDF value for the word in the corresponding document. In addition to keeping track of the unique words encountered, the word list also specifies the column in the weight matrix that is associated with each word.

In the second phase, the exhaustive search phase (lines 21 to 45), Weka classification models are constructed for the current dataset to determine F-measure values using a five step process. In the first step, the base F-measure is determined by running the current classifier without any feature selection method (lines 22 to 24). That is, using the full feature space. In the second step, the current classifier is run on the the current dataset using each feature selection method, in turn, with the objective of finding the maximum possible F-measure (lines 25 to 44) for each feature selection method. To find the maximum possible F-measure, each word is scored using the current feature selection method and the unique scores are stored in a sorted score list (lines 26 to 30). In the third step, each score is used as a cutoff point to eliminate features from the feature space (lines 33 to 37). In the fourth step, the maximum possible F-measure is determined by running the current classifier on the reduced feature space (lines 38 to 42). Finally, in the fifth step, the average of the maximum F-measures for the datasets is determined.

It is important to note that the search is exhaustive and to understand why an exhaustive search is required. For example, if there are 1,000 scores in the score list, the classifier must be run 1,000 times. It is not merely sufficient to run the classifier using the scores associated with the endpoints of some pre-determined intervals (e.g., every 100-th score) because that would result in determining the F-measure for only 10 scores, and the maximum F-measure could occur at some score not associated with an endpoint. Further, for each score, different words are eliminated from the feature space and the F-measures generated by each run of the classifier do not follow some regular curve, where the intermediate points can be interpolated. That is, the F-measure may increase or decrease as words are eliminated from the feature space.

4 Experimental Results

In this section, we present the results of our experimental evaluation of the CPD feature selection method. All of the experiments were run under Windows XP on an Intel Core 2 1.83 GHz processor with 3072 MB of memory. Due to the exhaustive search phase required to find the average maximum F-measure, the actual calendar duration of the experiments required several weeks to run to completion. The results are shown in Tables 4 through 6.

The relative performance of the seven feature selection methods is shown in Tables 4 and 5.

Table 4: Relative performance of feature selection using an SVM classifier

Text Corpus	*Feature Selection*	*Rank*	*Avg. Max. F-Measure*	*Change*	*Standard Deviation*	*Avg. Feature Space %*
OHSUMED	None	6	0.778	—	0.115	100.0
	CPD	1	**0.900**	+0.122	0.083	61.2
	χ^2	5	**0.821**	+0.043	0.124	24.9
	IG	2	**0.867**	+0.089	0.086	28.7
	DF	7	**0.768**	-0.010	0.111	46.9
	MI	2	**0.867**	+0.089	0.109	59.6
	OR	3	**0.857**	+0.079	0.082	16.2
	S-χ^2	4	**0.823**	+0.045	0.100	31.9
20 Newsgroups	None	7	0.956	—	0.037	100.0
	CPD	1	**0.979**	+0.023	0.018	76.5
	χ^2	6	0.957	+0.001	0.035	60.9
	IG	2	**0.974**	+0.018	0.021	46.5
	DF	8	0.948	-0.008	0.047	44.7
	MI	3	0.967	+0.011	0.030	78.9
	OR	4	0.964	+0.008	0.031	35.8
	S-χ^2	5	**0.963**	+0.007	0.029	48.2
Reuters-21578	None	8	0.761	—	0.074	100.0
	CPD	2	**0.805**	+0.044	0.071	64.7
	χ^2	5	**0.778**	+0.017	0.074	26.0
	IG	3	**0.800**	+0.039	0.064	9.5
	DF	6	**0.773**	+0.012	0.070	15.0
	MI	7	0.763	+0.002	0.073	99.7
	OR	1	**0.823**	+0.062	0.058	11.0
	S-χ^2	4	**0.790**	+0.029	0.067	10.1

Table 5: Relative performance of feature selection using a Naive Bayes classifier

Text Corpus	*Feature Selection*	*Rank*	*Avg. Max. F-Measure*	*Change*	*Standard Deviation*	*Avg. Feature Space %*
OHSUMED	None	8	0.754	—	0.109	100.0
	CPD	1	**0.856**	+0.102	0.088	66.3
	χ^2	5	0.774	+0.020	0.130	11.6
	IG	2	**0.847**	+0.093	0.085	13.0
	DF	7	**0.760**	+0.006	0.109	17.9
	MI	6	0.761	+0.007	0.120	92.8
	OR	3	**0.846**	+0.092	0.088	16.1
	S-χ^2	4	**0.817**	+0.063	0.095	11.9
20 Newsgroups	None	6	0.918	—	0.056	100.0
	CPD	1	**0.947**	+0.029	0.039	77.3
	χ^2	7	0.915	-0.003	0.054	15.3
	IG	1	**0.947**	+0.029	0.037	20.1
	DF	4	**0.922**	+0.004	0.055	23.0
	MI	5	0.921	+0.003	0.053	90.3
	OR	2	**0.941**	+0.023	0.039	24.6
	S-χ^2	3	**0.938**	+0.020	0.043	12.9
Reuters-21578	None	7	0.727	—	0.068	100.0
	CPD	3	**0.773**	+0.046	0.066	62.0
	χ^2	5	**0.752**	+0.025	0.066	18.9
	IG	2	**0.775**	+0.048	0.068	12.8
	DF	6	**0.746**	+0.019	0.070	11.5
	MI	8	**0.725**	-0.002	0.068	100.0
	OR	1	**0.783**	+0.056	0.066	30.8
	S-χ^2	4	**0.763**	+0.036	0.058	14.9

In Tables 4 and 5, the *Feature Selection* column describes the feature selection method used on the corresponding text corpus (the term "None" in this column describes the base case where no feature selection was used), the *Rank* column describes the standing of the F-measure for the corresponding feature selection method in relation to the F-measures for the other methods, the *Avg. Max. F-Measure* column describes the arithmetic mean of the ten largest F-measure values obtained from the ten randomly generated datasets, the *Change* column describes the difference between the average maximum F-measure value obtained when no feature selection is used and the average maximum F-measure value obtained for the corresponding feature selection method, the *Standard Deviation* column describes the standard deviation of the ten largest F-measure values used to determine the average maximum F-measure, and the *Avg. Feature Space %* column describes the average percentage of the feature spaces used corresponding to the ten largest F-measure values. For example, in Table 4, the average maximum F-measure corresponding to CPD when using OHSUMED is 0.900, representing an increase of +0.122 over the F-measure obtained when no feature selection is used, and the average percentage of the feature space used is 61.2%. The highest ranking feature selection method according to the average maximum F-measure is indicated by bold font. Average maximum F-measure values shown in bold represent a statistically significant difference from the F-measure obtained when no feature selection is used. The paired Student's t-test was used to determine statistical significance using a 95% level of significance (i.e., $\alpha = 0.05$) and a null hypothesis that the mean difference between paired observations is zero.

In Table 4, when using OHSUMED, all of the feature selection methods showed a statistically significant increase in the F-measure from when no feature selection is used, except for DF, which showed a statistically significant decrease. When using 20 Newsgroups, only CPD, IG, and S-χ^2 showed a statistically significant increase. DF showed a decrease, but it was not statistically significant. And when using Reuters-21578, all of the feature selection methods showed a statistically significant increase, except for MI. When using OHSUMED and 20 Newsgroups, CPD showed the largest increase, while on Reuters-

21578, OR showed the largest increase followed by CPD, which ranked second.

In Table 5, when using OHSUMED and 20 Newsgroups, only CPD, IG, DF, OR, and S-χ^2 showed a statistically significant increase. When using 20 Newsgroups, χ^2 showed a decrease, but it was not statistically significant. When using Reuters-21578, all of the feature selection methods showed a statistically significant increase, except for MI, which showed a statistically significant decrease. CPD showed the largest increase when using OHSUMED, and tied with IG when using 20 Newsgroups. When using Reuters-21578, OR showed the largest increase followed closely by IG and CPD, which ranked second and a very close third, respectively.

In comparison to the other feature selection methods, CPD appears to perform competitively according to the F-measure value, consistently showing statistically significant increases and having the largest F-measure in four out of six text categorization tasks. However, to this point, the relative performance of the individual feature selection methods has only been statistically verified against the base case where no feature selection method is used.A comparison of the relative performance of CPD to that of the other feature selection methods is shown in Table 6.

Table 6: Comparison of CPD to the other feature selection methods

Text Corpus	*Feature Selection*	*SVM*	*Naive Bayes*
OHSUMED	χ^2	+	+
	IG	+	+
	DF	+	+
	MI	+	+
	OR	+	+
	S-χ^2	+	+
20 Newsgroups	χ^2	+	+
	IG	+	∘
	DF	+	+
	MI	+	+
	OR	+	+
	S-χ^2	+	+
Reuters-21578	χ^2	+	+
	IG	∘	∘
	DF	+	+
	MI	+	+
	OR	–	∘
	S-χ^2	+	+

In Table 6, the *Feature Selection* column describes the feature selection methods to which CPD is being compared, and the *SVM* and *Naive Bayes* columns describe whether the F-measure for CPD is statistically significantly different from that of the other feature selection methods when using the SVM and Naive Bayes classifiers, respectively. The plus sign (i.e., +), circle (i.e., ∘), and minus sign (i.e., –) in these columns represent a statistically significant increase, no statistically significant difference, and a statistically significant decrease, respectively, in the F-measure value for CPD from that of the corresponding feature selection method. For example, we saw previously in Table 4 that the F-measures for CPD and IG are 0.900 and 0.867, respectively. In Table 6, the F-measure for CPD is shown to be statistically significantly different from that for IG. Specifically, the F-measure value for CPD represents a statistically significant increase from that for IG. Again, the paired Student's t-test was used to determine statistical significance using a 95% level of significance (i.e., $\alpha = 0.05$).

In Table 6, when using the SVM classifier, the F-measure values for CPD represent a statistically significant increase from all the other feature selection methods, regardless of the text corpus used, except for IG and OR when using Reuters-21578, where there is no statistically significant difference and a statistically significant decrease, respectively. Similar results are shown when using the Naive Bayes classifier, except for IG when using 20 Newsgroups and IG and OR when using Reuters-21578, where there is no statistically significant difference.

5 Conclusion and Future Work

We introduced and evaluated a new feature section method for text categorization tasks called CPD (categorical proportional difference). Experimental results showed that CPD outperformed other frequently studied feature selection met-hods in four out of six text categorization tasks using the OHSUMED, 20 Newsgroups, and Reuters-21578 text corpora.

Future work will focus on expanding the scope of the experiments to include additional classifiers and to utilize larger datasets with a greater number of features and categories. In addition, we will also attempt to identify distinct statistical differences between corpora to better understand the performance of a particular combination of classifier and feature selection method. For example, CPD was not the best performer on the Reuters-21578 dataset. Summary statistics for this dataset showed that it had a much lower percentage of words occurring in a single category. This suggests that measuring when a word and category occur together, and when neither the word nor category occur, values not considered by CPD, may be necessary for maximizing classifier performance. Finally, we will investigate whether statistical properties of a dataset can be used to predict the affect of feature selection on the dataset. For example, preliminary results have suggested that ratios based upon the number of overlapping and non-overlapping words across categories, and the distribution of words across categories show a surprising correlation to the size of the increase in accuracy that can be expected.

References

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

Forman, G. (2008), Feature selection for text classification, *in* H. Liu and H. Motoda, eds, 'Computational Methods of Feature Selection', Chapman and Hall / CRC, pp. 257–276.

Galavotti, L., Sabastiani, F. and Simi, M. (2000), Experiments on the use of feature selection and negative evidence in automated text categorization, *in* 'Proceedings of the 4th European Conference on Research and Advanced Technology for Digital Libraries (ECDL'00)', Lisbon, Portugal, pp. 59–68.

Han, E.-H., Karypis, G. and Kumar, V. (2001), Text categorization using weight adjusted k-nearest neighbor classification, *in* 'Proceedings of the 5th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'01)', Hong Kong, China, pp. 53–65.

Joachims, T. (1998), Text categorization with support vector machines: Learning with many relevant features, *in* 'Proceedings of the 10th European Conference on Machine Learning (ECML'98)', Chemnitz, Germany, pp. 137–142.

Kim, S.-B., Han, K.-S., Rim, H.-C. and Myaeng, S. (2006), 'Some effective techniques for naive bayes

text classification', *IEEE Transactions on Knowledge and Data Engineering* **18**(11), 1457–1466.

Mladenic, D. and Grobelnik, M. (1999), Feature selection for unbalanced class distribution and naive bayes, *in* 'Proceedings of the 16th International Conference on Machine Learning (ICML'99)', Bled, Slovenia, pp. 258–267.

Montanes, E., Diaz, I., Ranilla, J., Combarro, E. and Fernandez, J. (2005), 'Scoring and selecting terms for text categorization', *IEEE Intelligent Systems* **20**(3), 40–47.

Newsgroups (1999), http://kdd.ics.uci.edu/databases/20newsgroups/20newsgroups.html.

Ng, H., Goh, W. and Low, K. (1997), Feature selection, perceptron learning, and a usability case study for text categorization, *in* 'Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'97)', Philadelphia, Pennsylvania, pp. 67–73.

OHSUMED (2005), http://davis.wpi.edu/~xmdv/datasets/ohsumed.html.

Reuters-21578 (1997), http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html.

Sebastiani, F. (2002), 'Machine learning in automated text categorization', *ACM Computing Surveys* **34**(1), 1–47.

Witten, I. and Frank, E. (2005), *Data Mining: practical machine learning tools and techniques (2nd Edition)*, Morgan Kaufmann.

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