Exploiting Extremely Rare Features in Text Categorization[⋆]

Péter Schönhofen and András A. Benczúr

Informatics Laboratory
Computer and Automation Research Institute
Hungarian Academy of Sciences
Lagymanyosi u 11, H-1111 Budapest
schonhofen@gmail.com, benczur@sztaki.hu

Abstract. One of the first steps of document classification, clustering and many other information retrieval tasks is to discard words occurring only a few times in the corpus, based on the assumption that they have little contribution to the bag of words representation. However, as we will show, rare n-grams and other similar features are able to indicate surprisingly well if two documents belong to the same category, and thus can aid classification. In our experiments over four corpora, we found that while keeping the size of the training set constant, 5-25% of the test set can be classified essentially for free based on rare features without any loss of accuracy, even experiencing an improvement of 0.6-1.6%.

1 Introduction

Document categorization and clustering is a well studied area, several papers survey the available methods and their performance [21, 20, 4, 3]. In most results both frequent and rare words are discarded as part of pre-processing. The only measurement which takes rarity into account is the inverse document frequency in the tf-idf weighting scheme. In their classical paper Yang and Pedersen [21] disprove the widely held belief that common terms are non-informative for text categorization. In this paper we observe the same about *rare terms*; more precisely we show how rare words and n-grams ($n \leq 6$) [2] can be exploited to improve quality of classification. A feature instance w has **frequency** f if it is present in exactly f documents; the feature is **rare** if $f \leq 10$. For experiments with more features including skipping n-grams and contextual bigrams see the full version of the paper.

Our results indicate that topical similarity between two documents sharing the same extremely rare n-gram can be much stronger than those between their bag of words vectors [18] exploited by traditional classifiers. A possible explanation of this phenomenon can be given based on the assumption that a rare feature usually has some bias towards a certain topic and is not spread completely

[⋆] Support from the NKFP 2005 project MOLINGV. Full version available at www.ilab.sztaki.hu/websearch-data/Publications/rarewords-techrep.pdf

J. Fürnkranz, T. Scheffer, and M. Spiliopoulou (Eds.): ECML 2006, LNAI 4212, pp. 759–766, 2006.

uniformly across the documents. If the probability that the feature is present in a document is only by a small margin above the threshold required for the appearance in the corpus, then it is likely to appear in some of the documents about the characteristic topic of the feature but not but not in others.

In order to exploit rare words and n-grams in categorization we have to resolve the computational burden of handling a very large number of features. As the majority of features are rare and a single one of them is known to have little effect on the output, we may neither give them all nor a selected part of them as input to classifiers or clustering algorithms. Instead we preprocess the corpus by forcing documents with a sufficient number of rare features in common into the same category. This can be realized in several ways: we may pre-classify documents that share rare features with training set documents as the simplest use. We may either merge the content of documents that share rare features to mutually enrich their vocabulary or represent one with the text of the other. In Section 2 we give various methods for prioritizing among different features in common with different topics and documents, filtering out less reliable pairs, and resolving conflicts when features exist in common with more than one category.

Although the usefulness of rare features for classification has been already pointed out by Price and Thelwall [12], the approach proposed in this paper is essentially different from theirs. We emphasize that our method does not carry out feature extraction in the conventional sense, since we do not use rare features as document representatives. Instead, after exploiting them to discover rare feature instances, they are removed from documents before passing them to the classification algorithm.

To prove that rare words and features indeed reflect general topical similarity between documents and their usability does not depend on any peculiar characteristics of corpora, we tested our method on four text collections, namely Reuters-21578, Reuters Corpus Volume 1 (RCV1), Ken Lang's 20 Newsgroups, and the abstracts of patents contained in the World International Property Organization's (WIPO) corpus. Results are discussed in Section 3. The effect of our method on clustering accuracy is shown in the full paper.

1.1 Related Results

The idea to consider high and low frequency words separately originates in Luhn's [7] intuition (see also [18]) that middle-ranking words are the most indicative of the content. For example, [15] shows that words with the highest average discriminatory power tend to occur in 1%-90% of documents. Therefore infrequent words, usually thought to be typos or obscure phrases [18, 19], are often ignored in IR systems. When measuring the effect of removing rare and frequent words prior to clustering, Rigouste et al. [13] found that while the former hurts, rare words can be safely discarded. Similarly Yang and Pedersen [21] acknowledge that rare words have no significant influence on classification.

Several authors only partly accept that rare words are completely useless for classification. Price and Thelwall [12] show that words of frequency even as low as 2 are useful for academic domain clustering, suggesting that they

Algorithm 1. Connecting documents via rare features.

```
Discard words with frequency above threshold $cutoff$; select rare features with
  frequency $f \leq rarity$ based on remaining words
Weight each document pair by the number of common selected features.
Form the graph over documents with edges for pairs with weight at least $w_{\min}$.
for all pairs $d$, $d'$ in order of decreasing weight do
    if $d$ and $d'$ belongs to no pair then
        form component $(d, d')$
for all components do
    represent the component by either a random document of the component or the
      union of all text in the component
```

Table 1. Parameters of Algorithm 1

stemming	on or off
rarity	frequency f threshold to consider a feature rare
cutoff	maximum frequency of a word allowed to appear in a rare feature
$w_{\min}$	minimum no. rare features needed in common to connect two documents
$\mathrm{dist}_{\max}$	maximum distance of indirection to form composite documents
merging	choice of passing merged text or a sample document to the classifier

contain subject-related information. However they describe no efficient method to train a classifier based on rare terms; for future work they envision an artificial intelligence or natural language processing approach which would discard useless ones [1]. Similarly [21] mentions that discarding rare words too aggressively can be counterproductive but gives no solution to the computational issues.

A problem of rare words is that due to their large number they cannot be fed to computationally hard methods that would be able to separate useful words from useless ones; algorithms such as vocabulary spectral analysis [17] are infeasible over rare words. In addition, rare words often cause noise that confuse term weighing and feature selection such as χ^2 [14] or mutual information [21,11]. The only exception is the inverse document frequency or idf [16] that is commonly used in summarization, feature extraction and dimensionality reduction.

2 Algorithm for Connecting Documents Via Rare Features

Next we describe our algorithm that, prior to classification, preprocesses documents by connecting them via rare features in common. The algorithm can be tuned by several parameters to maximize the gain in classification quality. We may exclude very frequent words from rare n-grams (*cutoff*), limit the maximum feature frequency (*rarity*) to consider a feature rare; require several features in common to connect two documents ($w_{\min}$) and limit the distance ($\mathrm{dist}_{\max}$) between connected documents. The parameters are described in detail next and summarized in Table 1.

Algorithm 2. Kruskal's algorithm in the special case of connecting test set documents to train set ones (supervised case).

```
for distance = 1, ..., $\text{dist}_{\max}$ do
  for all remaining test set documents $d$ do
    if $d$ has edge to at least one $d'$ in the train set then
      merge $d$ with all of its edges into $d'$ where the weight of $(d, d')$ maximum
for all documents $d'$ in the train set do
  for all $d$ merged with $d'$ do
    Classify $d$ into the category of $d'$
    Expand $d'$ with the text of $d$ /*optional*/
    Remove $d$ from test set
```

The main idea of the algorithm is to greedily pair documents in the order of decreasing number of rare features in common. The algorithm hence uses the simplest form of single linkage clustering [10, and many others]; we tested a few more complicated versions but achieved no improvements. The general method is described in Algorithm 1 while a slightly stronger version of steps 4–8 is given in detail for the simpler special case of connecting test set documents to train set ones in Algorithm 2. Later in Fig. 1-c we will justify the choice of this simple clustering algorithm by showing that pairs connected by rare features are in isolation and larger components appear only sporadically.

The main computational effort in our Algorithm 1 is devoted to identifying rare features (line 1), a very large fraction of all features. It is easy to implement the selection by external memory sorting; in our experiments we choose the simpler and faster internal memory solution that poses limits on the corpus size.

Given the collection of rare features together with the documents containing them, we build an undirected graph over documents as nodes (lines 2–3). We iterate over the features and add a new edge candidate whenever we discover a pair of documents sharing a rare feature. We weight edges by the number of rare features in common and discard edge candidates below weight $w_{\min}$.

We connect documents by iterating through edges in the order of decreasing weight. If the next highest weight edge (d, d') is such that neither d nor d' belongs to a component formed earlier in line 6, then we add this pair as a new component. More complex algorithms may form components of chains of length up to some $\text{dist}_{\max}$, replace the greedy choice of the loop in line 4 for example by a maximum weight matching algorithm, or form larger components by iteratively merging them into new ones.

Finally we pass the corpus to the classifier; here for the components we have the choice to pass the merged text or just a randomly selected document to the classifier. Optionally we may enrich the content of each document d of the train set with the text of all or some documents d' merged with d during the algorithm. If we train the classifier with the extended documents, we observe that their richer content characterize categories better.

While our preprocessing algorithm is also suited for unsupervised clustering, if exists, we may prefer train–test document connections. Documents connected

Table 2. Characteristics of the four corpora used in our experiments

Corpus	Reuters-21578	RCV1	20 Newsgroups	WIPO abstr.
No. of docs	10,944	199,835	18,828	75,250
No. of categories	36	91	20	114
Avg doc length	69 words	122 words	125 words	62 words

to the train set can then be classified "for free". This special case is described in Algorithm 2 where we iterate through all test set documents d; whenever d is connected to another in the train set, we merge d with d' such that they are connected by the largest number of features in common.

We may also consider indirect connections to the train set. If document d is connected to another in the test set that is in turn connected to d' of the train set, we may also pre-classify d into the category of d'. We set a distance threshold $\text{dist}_{\max}$; this turns into $\text{dist}_{\max}$ iterations of the first **for** loop of Algorithm 2.

3 Experiments

We tested our algorithm on four corpora of different domain and nature. Table 3 shows their most important properties. For all corpora, we removed stop words and performed stemming with the Morph component of WordNet [9].

Reuters-21578 [5] and Reuters Corpus Volume 1 (RCV1) [6] contains news about politics, economics and trade. In RCV1 we characterized documents solely by their topic codes, industry and country classifications were ignored. If a document was assigned to multiple topics, it was re-assigned to the smallest, provided that it covered at least 50 documents; documents without topic indication were discarded. Due to performance limitations, only the first 200,000 documents were considered from RCV1. However, to demonstrate our method's scalability, in the full version of the paper we also processed RCV1 in a partitioned way.

Ken Lang's 20 Newsgroups, or more precisely its slightly modified and cleaned variant made available by Jason Rennie (`http://people.csail.mit.edu/jrennie/20Newsgroups/20news-18828.tar.gz`), consists of short Usenet postings evenly distributed among 20 domains. The World Internet Property Organization (WIPO) corpus is a collection of patents organized in a strict multilevel classification system. Because the full text of documents is unnecessarily long for our purposes, we only utilize abstracts. Furthermore due to the very large number of categories, we only keep the top two levels of the hierarchy.

First let us explore the quality of rare words and n-grams for $2 \leq n \leq 6$ with frequency $2 \leq f \leq 10$. Let $q(w)$ be the probability that two random documents containing w belong to the same category (that may be apriori assumed in an unsupervised experiment). For a fixed feature type, **feature quality** is the average of $q(w)$ over all feature instances w of the given type with frequency $f \geq 2$. Quality is inversely proportional to the number of pairs formed by the preprocessing algorithm; the expected performance of our preprocessing algorithm based on these two values is explored in the full version of the paper.

As Fig. 1-d shows, quality quickly decays with increasing frequency f for short features (words, bigrams), while for 5- and 6-grams, quality remains high even at frequencies close to 10. We also observed a slight increase in quality when lowering the *cutoff* limit to exclude frequent words from rare features (Fig. 1-a); however, this affect classification and clustering accuracy only marginally, since coverage becomes very low.

Fig. 1-b shows that out of the document pairs generated by our algorithm, what percentage of pairs connected by selected features have Jaccard similarity less than or equal to the threshold specified over the horizontal axis. The figure supports our claim that documents share rare features due to a general topical similarity and not just because of some side effect of (near) replication or quoting from other documents. If rare features all arose from duplicates, the curve would proceed close to 0, jumping to 1 only when the similarity threshold is increased to its maximum. If, on the other hand, rare features appeared in very dissimilar documents in common, then the curve would jump already at a fairly low similarity level. In fact, in 20 Newsgroups our method pairs the least similar documents, while in RCV1 we observe just the contrary, pairings are between the most similar ones.

As seen in Fig. 1-c, the largest fraction of selected rare features connect pairs of documents that remain in isolation. arising in line 3 of Algorithm 1, the number of connected components of various sizes decays exponentially and components of more than four documents occur only sporadically.

For classification we used the naive Bayes component of the Bow toolkit [8] with default parameters, as this enhanced naive Bayes implementation often outperformed the SVM classifier. Improvements achieved by our method are shown on Fig 1-e–f and h–i for the four corpora. In the figures, each measurement point represents the average accuracy of five random choices of training documents. The values for the free parameters of Table 1 are $rarity = 2$, $cutoff = 1000$, $w_{\min} = 3$ for all except WIPO where $w_{\min} = 2$, and merging set on. In Fig. 1-g we see that our algorithm significantly reduces the number of documents passed to the classifier.

In Fig. 1-e–f and h–i we see that except from 20 Newsgroups, the usefulness of the various feature types relative to each other reflect the ranking as expected by their feature quality: words are the least efficient, with n-grams providing better results; bigrams and 3-grams however perform unexpectedly well.

Note that improvements for 20 Newsgroups roughly stabilize beyond 10% training set ratio, possibly because for larger training sets the accuracy of the classification algorithm approaches the quality of features, diminishing their power.

4 Conclusion and Future Work

This paper presented a novel approach in which extremely rare n-grams, mostly neglected by previous research, are exploited to aid classification and clustering. In our experiments on four different corpora we found that even simple features

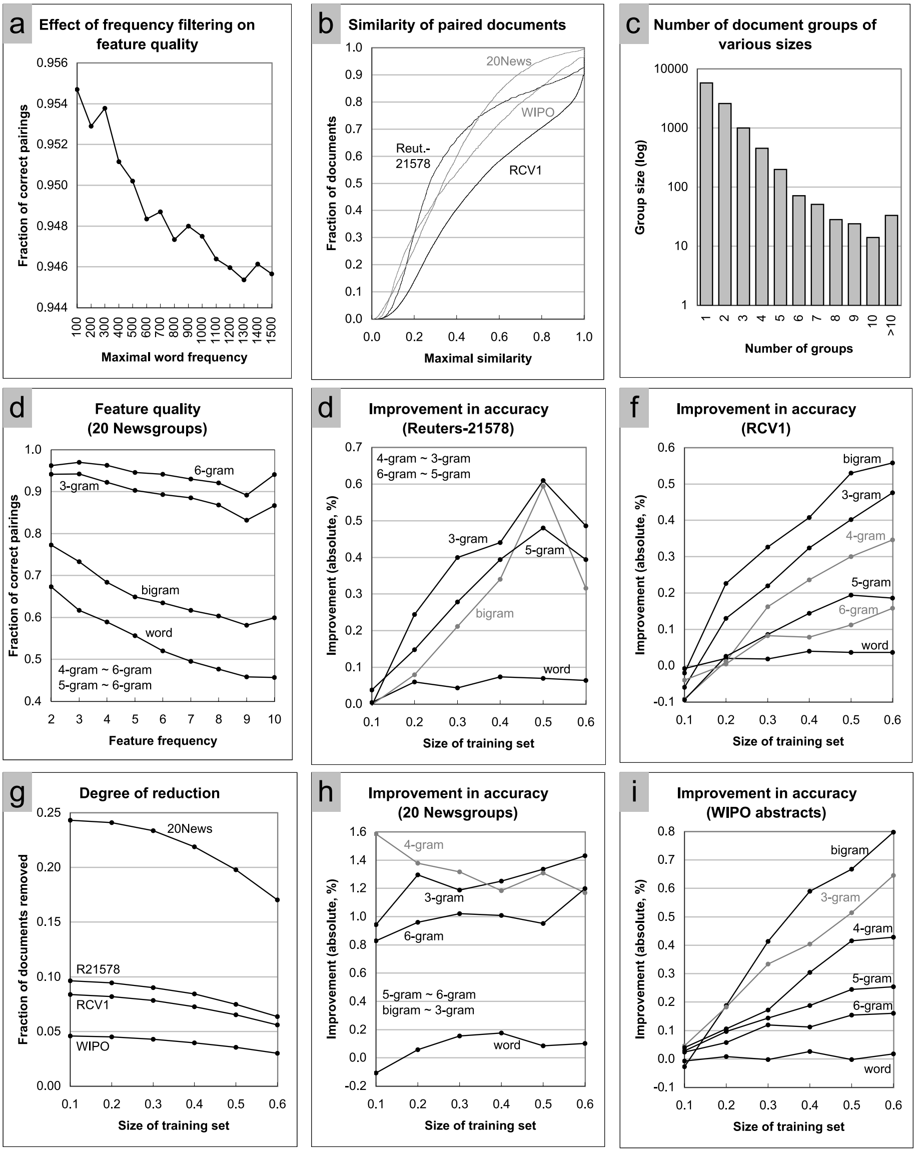

Fig. 1. a: Feature quality of selected features in 20 Newsgroups. **b:** Histogram of the Jaccard similarity of document pairs connected by the co-occurrence of a frequency 2 trigram. **c:** The number of components of various sizes connected by the frequency 2 trigrams of 20 Newsgroups. **d:** Feature quality as the function of *cutoff* in frequency 2 trigrams over 20 Newsgroups.**e–f** and **h–i:** Improvement over the baseline classification accuracy. For the sake of clarity we merged very close lines into one. **g:** The fraction of documents paired by our algorithm. Features used are 4-grams for 20 Newsgroups, trigrams for Reuters-21578 and bigrams for RCV1 and WIPO.

like rare bigrams and 3-grams are able to improve accuracy. Future directions may include new features and more sophisticated merging algorithms.

References

1. D. C. Comeau and W. J. Wilbur. Non-word identification or spell checking without a dictionary. *J. Am. Soc. Inf. Sci. Technol.*, 55(2):169–177, 2004.
2. J. Goodman. A bit of progress in language modeling. *CoRR*, cs.CL/0108005, 2001.
3. M. Iwayama and T. Tokunaga. Cluster-based text categorization: a comparison of category search strategies. In *SIGIR '95*, pages 273–280, 1995.
4. T. Joachims. Text categorization with suport vector machines: Learning with many relevant features. In *Proc. European Conference on Machine Learning*, pages 137–142, 1998.
5. D. D. Lewis. Reuters-21578 text categorization test collection, distribution 1.0, available at `http://www.daviddlewis.com/resources/ testcollections/ reuters21578` , 1997.
6. D. D. Lewis, Y. Yang, T. G. Rose, and F. Li. RCV1: A new benchmark collection for text categorization research. *J. Mach. Learn. Res.*, 5:361–397, 2004.
7. H. P. Luhn. A statistical approach to mechanized encoding and searching of literary information. *IBM J. Research and Development*, 1(4):309–317, 1957.
8. A. K. McCallum. Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. http://www.cs.cmu.edu/˜mccallum/bow, 1996.
9. G. A. Miller. Wordnet: A lexical database for English. *Commun. ACM*, 38(11):39–41, 1995.
10. P. Pantel and D. Lin. Discovering word senses from text. In *Porc. SigKDD*, 2002.
11. V. Pekar and M. Krkoska. Weighting distributional features for automatic semantic classification of words. In *International Conference on Recent Advances In Natural Language Processing*, pages 369–373, 2003.
12. L. Price and M. Thelwall. The clustering power of low frequency words in academic webs: Brief communication. *J. Am. Soc. Inf. Sci. Technol.*, 56(8):883–888, 2005.
13. L. Rigouste, O. Cappe, and F. Yvon. Evaluation of a probabilistic method for unsupervised text clustering. In *International Symposium on Applied Stochastic Models and Data Analysis (ASMDA)*, 2005.
14. M. Rogati and Y. Yang. High-performing feature selection for text classification. In *Proc. International Conference on Information and Knowledge Management*, pages 659–661, 2002.
15. G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. Technical report, Ithaca, NY, USA, 1974.
16. K. Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21, 1972.
17. M. Thelwall. Vocabulary spectral analysis as an exploratory tool for scientific web intelligence. In *Proc. Information Visualisation*, pages 501–506, 2004.
18. C. J. Van Rijsbergen. *Information Retrieval, 2nd edition.* Dept. of Computer Science, University of Glasgow, 1979.
19. M. Weeber, R. Vos, and R. H. Baayen. Extracting the lowest frequency words: Pitfalls and possibilities. *Computational Linguistics*, 26(3):301–317, 2000.
20. P. Willett. Recent trends in hierarchic document clustering: A critical review. *Inf. Process. Manage.*, 24(5):577–597, 1988.
21. Y. Yang and J. Pedersen. A comparative study on feature selection in text categorization. In *Proc. ICML-97*, pages 412–420, 1997.