Fast Supervised Feature Extraction by Term Discrimination Information Pooling

Amara Tariq
Dept. of Computer Science
LUMS School of Science and Engineering
Lahore, Pakistan
09060012@lums.edu.pk

Asim Karim
Dept. of Computer Science
LUMS School of Science and Engineering
Lahore, Pakistan
akarim@lums.edu.pk

ABSTRACT

Dimensionality reduction (DR) through feature extraction (FE) is desirable for efficient and effective processing of text documents. Many of the techniques for text FE produce features that are not readily interpretable and require superlinear computation time. In this paper, we present a fast supervised DR/FE technique, named FEDIP, that is motivated by the notion of relatedness of terms to topics or contexts. This relatedness is quantified by using the discrimination information provided by a term for a topic in a labeled document collection. Features are constructed by pooling the discrimination information of highly related terms for each topic. FEDIP's time complexity is linear in the size of the vocabulary and document collection. FEDIP is evaluated for document classification with SVM and naive Bayes classifiers on six text data sets. The results show that FEDIP produces low-dimension feature spaces that yield higher classification accuracy when compared with LDA and LSI. FEDIP is also found to be significantly faster than the other techniques on our evaluation data sets.

Categories and Subject Descriptors

I.7.m [**Document and Text Processing**]: Miscellaneous; H.2.8 [**Database Management**]: Database Applications—*Data Mining*

General Terms

Algorithms, Theory

1. INTRODUCTION

Dimensionality reduction (DR) via feature extraction (FE) is an important step in many textual information retrieval and text mining applications. It often produces more efficient and effective solutions, and in certain applications, can ensure the practical feasibility of a solution because of its efficiency. A popular text mining application is that of document classification where labeled documents are used to learn a classifier for labeling new documents. Feature extraction for document classification can yield classifiers that are more robust to noise and shifts in distributions. Another popular application is document understanding and visualization. A fast FE technique can enable interactive visualizations of documents and their features.

In this paper, we propose a fast and effective supervised feature extraction technique for textual information. Our technique, named FEDIP, is motivated by the notion of relatedness of terms to topics or contexts in which they occur. This notion of relatedness has been shown to play a significant role in human comprehension of text [3, 5]. For instance, humans can easily associate terms with a particular topic after reading a few documents discussing that topic. We use discriminative term weights, computed via the relative risk of a term in a topic-labeled document collection, to quantify the relatedness of terms to topics. Features are constructed by pooling the discrimination information of highly related terms for each topic. We evaluate FEDIP for document classification, comparing it with Latent Semantic Indexing (LSI) [2] and Linear discriminant Analysis (LDA) [8] on six data sets. The results show that FEDIP produces low-dimension feature spaces in which document classification is more accurate and efficient. FEDIP represents a new semantically motivated term pooling approach for text FE that supports efficiency and effectiveness in text mining.

The remainder of this paper is organized as follows. In Section 2, we outline the motivation for our feature extraction technique. Section 3 presents our fast supervised technique for feature extraction. We discuss the experimental evaluation of our technique in Section 4. Finally, we present our concluding remarks in Section 5.

2. MOTIVATION: EXPLOITING THE RELATEDNESS OF TERMS TO TOPICS

Typically in text mining and information retrieval, the notion of relatedness refers to the semantic relatedness between pairs of terms (e.g. words). Accordingly, two terms are said to be related when they are linked by semantic relationships. Usually these are the so-called classical lexical relationships like synonymy, antonymy, and hypernymy. Numerous semantic relatedness measures of this type have been proposed and evaluated in the literature. However, these measures have been developed for and are used primarily for natural language processing tasks like word sense disambiguation [6, 7].

On the other hand, the notion of relatedness of terms to contexts or topics is less known. Nonetheless, this notion

CIKM'11, October 24–28, 2011, Glasgow, Scotland, UK.
Copyright 2011 ACM 978-1-4503-0717-8/11/10 ...$10.00.

of relatedness is more powerful for some applications where contexts are pre-defined or known. Furthermore, this notion of relatedness has been linked to humans' comprehension of text by association of terms with their contexts. In a study by [5], it has been argued that readers develop non-classical relationships between terms by grouping terms associated to common contexts. These groups of terms may then serve as units of understanding of the respective contexts [3].

These observations indicate that groups of terms that are related to a particular context convey meaning about that context or topic. Thus, if we can find such groups of terms then we can pool their information content to construct features. These features will better capture the meaning of the text and aid in document understanding and document classification. Our FE technique, FEDIP, uses term discrimination information as a measure of its relatedness to a context in a labeled document collection, and constructs features by pooling the discrimination information of terms related to each context.

3. FEDIP – OUR FEATURE EXTRACTION TECHNIQUE

We describe the key steps of our feature extraction technique, FEDIP (Feature Extraction by [Term] Discrimination Information Pooling), in the following subsections. Here, we define the problem setting and notations.

Let $\mathcal{D} = \{d_1, d_2, d_3, \ldots, d_N\}$ be the set of documents and $\mathcal{V} = \{t_1, t_2, t_3, \ldots, t_M\}$ be the vocabulary of terms in these documents. The context or topic of a document d is identified by the context label $cont(d) \in \{c_1, c_2, c_3, \ldots, c_K\}$, where K is the total number of topics in $\mathcal{D}$ (usually $K \ll N$). The weight of term t_j in document d_i is identified by $x_{ij} = tw(t_j, d_i)$, which is always greater than or equal to zero. In practice, this term weight can be a 0/1 value (e.g. term occurrence), an integer value (e.g. term count), or a positive real number (e.g. term frequency inverse document frequency or TFIDF).

Given the above setting, a document d_i is represented by the vector $\mathbf{x}_i = \langle x_{i1}, x_{i2}, \ldots, x_{iM}\rangle$ in the input space $\mathcal{X} \subset \Re^M$. After dimensionality reduction, document d_i is represented by the vector $\mathbf{y}_i = \langle y_{i1}, y_{i2}, \ldots, y_{iR}\rangle$ in the feature space $\mathcal{Y} \subset \Re^R$. For our technique, the dimension R can range from K, the number of context labels, to M, the number of dimensions in the input space (i.e. $K \leq R \leq M$).

3.1 Discriminative Term Weights

Measures of discrimination information can be used to quantify the discriminative power provided by a term for a given context over all other contexts. They can also be viewed as the strength of the evidence or opinion that a term provides for a given context. Measures of discrimination and association have been studied for different purposes in the literature including feature selection, association rule mining, and text classification. The idea of discriminative term weights is introduced in [4] for quantifying the discrimination information provided by terms for classification purposes. Three measures (Kullbach-Leibler divergence, relative risk, and log relative risk) are evaluated and it is reported that the relative risk of a term provides robust classification performance.

In this work, we use the relative risk of a term to quantify its discrimination information for a given context over the others. Accordingly, the discriminative term weight of term t_j for context c_k is defined as

$$dtw(t_j, c_k) = \frac{p(t_j|c_k)}{p(t_j|\bar{c}_k)} \tag{1}$$

where $p(t_j|c_k)$ denotes the probability of term t_j in documents belonging to context c_k and $\bar{c}_k$ refers to documents in all contexts but c_k. If $dtw(t_j, c_k) > 1$ then term t_j provides positive discrimination information for context c_k, with larger values signifying higher discriminative power.

The probabilities in Equation 1 are computed from the set of labeled document $\mathcal{D}$ via maximum likelihood estimation (MLE). We evaluate two document models in this work: (a) documents' terms follow a Multinomial distribution and (b) each term follows a Bernoulli distribution.

Once the discriminative term weights are estimated, we can identify all the terms that provide significant positive discrimination information for a given context. Specifically, the set of terms $\mathcal{V}_k$ providing significant positive discrimination information for context c_k is defined as

$$\mathcal{V}_k = \{t_j \| dtw(t_j, c_k) > T \; \forall j\} \tag{2}$$

where $T \geq 1$ is a term selection parameter controlling the exclusion of insignificant terms. It is important to note that, in general, $\mathcal{V}_k \cap \mathcal{V}_l \neq \emptyset$ for all k and l. Also, depending on the value of T, $\cup_k \mathcal{V}_k \neq \mathcal{V}$ as some terms may not provide significant discrimination information for any context.

3.2 Relatedness of Terms to Contexts

Cai and van Rijsbergen [1] define the relatedness of a term to a context/topic as the product of a weight of the term in the context and a discrimination information measure of the term for the context. Following their framework, we define the relatedness of term t_j to context c_k as follows:

$$rel(t_j, c_k) = p(t_j|c_k) \times dtw(t_j, c_k) \tag{3}$$

Thus, the weight of a term in the context c_k is taken to be the probability of the term in documents belonging to context label c_k. Intuitively, a term is more related to a context if its discriminative term weight is larger and if it occurs frequently in documents belonging to the context.

This choice of the weight of a term in a context leads to the interpretation that the relatedness of a term to a context (Equation 3) is the term's contribution towards the expected discrimination information provided by all significant terms for that context. The expected discrimination information provided by all terms $t \in \mathcal{V}_k$ for context c_k over the other contexts can be written as

$$E_k = \sum_{t_j \in \mathcal{V}_k} p(t_j|c_k) \times dtw(t_j, c_k) = \sum_{t_j \in \mathcal{V}_k} rel(t_j, c_k). \tag{4}$$

In words, E_k is the expectation of the discrimination information for context c_k based on terms in the labeled document collection. And, the relatedness of a term to this context represents its contribution to the expected value E_k; the stronger the relatedness, the larger the contribution to E_k.

3.3 Feature Construction

In the previous subsections, we have defined the relatedness of a term to a context (Equation 3) and identified all terms that are related to a given context (Equation 2). Given this setup, we can construct features by pooling the discrimination information provided by terms in $\mathcal{V}_k$ ($k = 1, K$). Consider a document d_i represented in the input space by the vector $\mathbf{x}_i$. The representation of this document in the feature space $\mathbf{y}_i$ is defined as follows:

$$y_{ik} = \frac{\sum_{t_j \in \mathcal{V}_k} tw(t_j, d_i) \times dtw(t_j, c_k)}{\sum_{t_j \in \mathcal{V}_k} tw(t_j, d_i)} \quad k = 1, K \qquad (5)$$

This expression will create $R = K$ features, where K is the total number of contexts in the document collection. Each feature represents the linear opinion pool or ensemble average of the discrimination information provided by terms in the document. If a term does not occur in a document, then the discrimination information or opinion of that term is not included. The larger the value of y_{ik} the greater is the chance that document d_i belongs to context c_k. The extension for $R > K$ features is omitted from this shortened paper.

3.4 Algorithm and its Computational Complexity

Algorithm 1 FEDIP

```
1: Input: $\{d_i\}_{i=1}^N$ (documents), $\mathcal{V} = \{t_j\}_{j=1}^M$ (term vocabulary), $cont(d_i) \forall i$ (contexts of documents), $K$ (no. of contexts), $tw(t_j, d_i) \forall j, i$ (document-term weights), $R = K$ (no. of features)
2: Output: $\{\mathbf{y}_i\}_{i=1}^N$ (feature vectors of length $R$)
3:
4: for $k = 1 \to K$ do
5:     for $j = 1 \to M$ do
6:         $dtw(t_j, c_k) \leftarrow$ discriminative term weight of term $t_j$ for context $c_k$ (Eq. 1)
7:     end for
8:     $\mathcal{V}_k \leftarrow$ significant terms for context $c_k$ (Eq. 2)
9: end for
10:
11: for $i = 1 \to N$ do
12:     for $r = 1 \to R$ do
13:         $y_{ir} \leftarrow$ linear opinion pool of terms in $\mathcal{V}_r$ (Eq. 5)
14:     end for
15: end for
```

The steps in FEDIP are given in Algorithm 1. There are two main processing blocks in FEDIP. The first code block (lines 4– 9) finds the discriminative term weights and constructs the term pool for each context. The estimation of the discriminative term weight (line 6) for a given term and context requires one pass over the set of labeled documents. Hence, the time complexity of this block of processing is $O(KMN)$.

The second block of processing (lines 11 – 15) computes the desired features by outputting a feature value for each context. The time complexity of this processing is $O(RMN)$. The overall worst-time complexity of FEDIP is $O(KMN) + O(RMN)$. This expression shows that FEDIP's time complexity depends linearly upon the data parameters N and M, and since $K = R$ is usually much smaller than both M and N, FEDIP is computationally very efficient in practice.

4. EMPIRICAL EVALUATION

We evaluate FEDIP for document classification using six text data sets. Its performance is compared with Latent Semantic Indexing (LSI) and Linear Discriminant Analysis (LDA) under Support Vector Machine (SVM) and Naive Bayes (NB) classifiers. Our experiments are conducted in the MATLAB environment (LSI, NB). For SVM we use the LIBSVM[1] implementation, and for LDA we use the LDA/QR[2] implementation [8]. In all our experiments, the term selection parameter T is set equal to one.

Table 1: Key characteristics of the evaluation data sets

Name	Docs (N)	Terms (M)	Contexts (K)
MReview	2000	31296	2
Hitech	2301	13170	6
TR31	927	10128	7
TR41	878	7454	10
RE0	1504	2886	13
WAP	1560	8460	20

The names and key characteristics of the selected data sets are given in Table 1. The first data set is taken from a Cornell University page[3], while the remaining five data sets are available from Karypis Lab at University of Minnesota[4]. Brief descriptions of these data sets can be found on the respective Web sites.

FEDIP preserves the topic or context information in the reduced feature space. This characteristic is highly desirable for visualization of document collections and document classification. We illustrate this in Figure 1. In this figure, the left plot shows the Movie Review data set in the 2-dimensional feature space produced by FEDIP, while the right plot shows the same for LSI. The separation of documents belonging to the two contexts is evident in the feature space produced by FEDIP. Furthermore, the feature values produced by FEDIP are readily interpretable as relevance of documents to the respective contexts. In general, the scatter plots of documents in the 2-dimensional feature space produced by FEDIP have documents belonging to one context spread along one axis and documents belonging to the other context spread along the other axis with some overlap along the diagonal. This representation suggests that a linear discriminant passing through the origin can produce accurate classification.

Table 2 gives the classification performance (average percent accuracy and standard deviation obtained from five random train/test runs in which the size of the test set is 33% of the total data size) of SVM and NB classifiers in the feature space. The number of features is equal to the number of contexts ($R = K$) of the data sets, except for LDA/QR, for which $R = K - 1$. FEDIP-a is the variant based on the Multinomial model and FEDIP-b is the variant based on the Bernoulli model.

The following observations and interpretations can be made from these results. (1) FEDIP's feature spaces yield higher classification accuracy for the majority of the data sets. FEDIP's performance is much better for data sets with fewer contexts when compared to LDA/QR. (2) The performance gap between FEDIP and LDA/QR decreases as the number

[1] http://www.csie.ntu.edu.tw/ cjlin/libsvm/

[2] http://sites.google.com/site/mydemossite/

[3] http://www.cs.cornell.edu/people/pabo/movie-review-data/

[4] http://glaros.dtc.umn.edu/gkhome/cluto/cluto/download

Table 2: Classification performance of different feature extraction techniques when paired with SVM and NB classifiers

Data	NB Classifier				SVM Classifier			
	FEDIP-a	FEDIP-b	LSI	LDA/QR	FEDIP-a	FEDIP-b	LSI	LDA/QR
MReview	79.19 ± 0.63	**82.16 ± 0.73**	56.01 ± 3.76	65.98 ± 3.61	78.92 ± 0.67	81.89 ± 0.92	48.59 ± 0.57	66.13 ± 4.21
Hitech	70.3 ± 1.21	67.81 ± 0.66	53.91 ± 1.82	68.36 ± 0.95	**71.58 ± 1.19**	67.66 ± 0.92	31.37 ± 0.78	69.93 ± 1.32
TR31	95.5 ± 0.85	96.89 ± 0.995	77.09 ± 1.15	71.59 ± 6.5	97.73 ± 0.46	**98.38 ± 0.46**	55.02 ± 2.48	91.34 ± 2.39
TR41	93.33 ± 0.71	93.26 ± 1.10	84.12 ± 2.01	82.27 ± 1.91	95.26 ± 0.77	**95.81 ± 0.96**	44.19 ± 3.63	94.02 ± 1.33
RE0	78.84 ± 2.00	76.92 ± 2.30	60.88 ± 2.23	73.49 ± 2.50	**85.15 ± 1.49**	81.16 ± 1.75	42.28 ± 0.91	81.92 ± 2.02
WAP	73.95 ± 1.57	77.46 ± 1.98	67.94 ± 1.98	75.92 ± 1.30	78.65 ± 1.61	78.57 ± 0.89	29.59 ± 0.91	**81.5 ± 1.24**

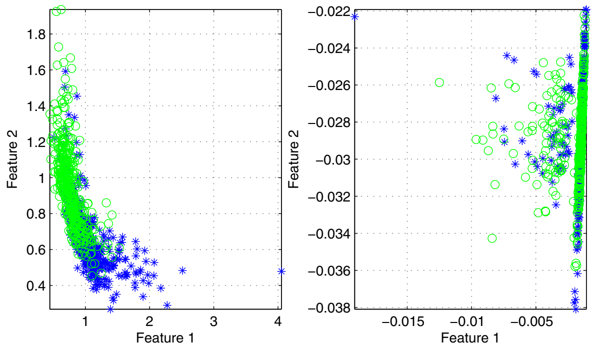

Figure 1: Scatter plot of Movie Review data set in 2-D feature space. Left plot is for FEDIP and right plot is for LSI. The two categories are identified by different color markers.

of contexts increases. (3) LSI's feature spaces produce poor classification accuracy, especially when paired with SVM. Since LSI is unsupervised, it is not able to preserve the category separation in the low-dimension feature space. Moreover, linear SVM, which is discriminative in nature, does poorly when compared to NB, which is generative in nature, in LSI feature spaces. (4) FEDIP-a and FEDIP-b can produce slightly different classification accuracies, and no clear winner is observable from our evaluation.

We now compare the computation times required by FEDIP, LSI, and LDA/QR on our evaluation data sets. Instead of providing absolute run-time values, which depend heavily upon the computing environment, we provide the relative run-times on our computing setup. We observed the run-times for all data sets and found that LDA/QR is at least 4 times slower than FEDIP and LSI is at least 15 times slower than FEDIP. Even though LDA/QR has a theoretical worst-case time complexity identical to that of FEDIP [8], in practice we found it to be significantly slower than FEDIP.

5. CONCLUDING REMARKS

Motivated by the notion of relatedness of terms to topics or contexts, we develop and evaluate a supervised feature extraction technique for text, named FEDIP, based on discriminative term weighting and term pooling. FEDIP produces features that are readily interpretable as the strength of the discrimination information provided by the term pool for a given context. The computational complexity of FEDIP is linear in M (number of terms) and N (number of documents). These characteristics of FEDIP are evaluated on six text data sets and its performance compared with LSI and LDA/QR using naive Bayes and SVM text classification. FEDIP outperforms LSI and LDA/QR especially when the dimensions have been reduced significantly.

This work demonstrates that term pooling is a practically effective technique for text dimensionality reduction. It also highlights the use of relatedness of terms to contexts. There is much potential for future research in this direction. Specifically, it is worth investigating the use of knowledge-based measures of relatedness (in addition to the corpus-based measure used in this work) for term pooling. We also feel that term selection, via the term selection parameter T, can improve the quality of the feature spaces produced by FEDIP.

6. REFERENCES

[1] D. Cai and C. J. van Rijsbergen. Learning semantic relatedness from term discrimination information. *Expert Systems with Applications*, 36:1860–1875, 2009.

[2] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by Latent Semantic Analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

[3] M. A. K. Haliday and R. Hassan. *Cohesion in English*. Longman, London, UK, 1976.

[4] K. Junejo and A. Karim. A robust discriminative term weighting based linear discriminant method for text classification. In *Proceedings of Eighth IEEE International Conference on Data Mining (ICDM '08)*, pages 323 –332, 2008.

[5] J. Morris and G. Hirst. Non-classical lexical semantic relations. In *Proceedings of the HLT-NAACL Workshop on Computational Lexical Semantics*, CLS '04, pages 46–51. Association for Computational Linguistics, 2004.

[6] P. Resnik. Using information content to evaluate semantic similarity. In *Joint Conference on Artificial Intelligence*, pages 448–453, Montreal, Canada, August 1995.

[7] G. Tsatsaronis, I. Varlamis, and M. Vazirgiannis. Text relatedness based on a word thesaurus. *Journal of Artificial Intelligence Research*, 37:1–39, 2010.

[8] J. Ye and Q. Li. A two-stage linear discriminant analysis via QR-decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27:929–941, 2005.