Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 115 (2017) 218–225

Procedia Computer Science

www.elsevier.com/locate/procedia

7th International Conference on Advances in Computing & Communications, ICACC-2017, 22-24 August 2017, Cochin, India

Experimental Study Of Feature Weighting Techniques For URL Based Webpage Classification

R. Rajalakshmi*, Sanju Xaviar

School of Computing Science and Engineering, VIT University, Chennai - 600127, Tamil Nadu, India

Abstract

Information retrieval task has become a difficult task due to the growing size of the web. This demands a simple method for classifying the web pages. We propose an URL based approach, as it avoids downloading the web page contents. Feature weighing methods play an important role in improving the performance of a classifier. In this paper, we explored different weighting methods and conducted various experiments on WebKB dataset. Results show that tf.mi feature weighting technique achieves F_1 measure of 79% and outperforms other weighting methods, which is an improvement of 19.6% over existing works on URL based classification.

Peer-review under responsibility of the scientific committee of the 7th International Conference on Advances in Computing & Communications.

Keywords: Feature weighting, URL Features, Web page classification

1. Introduction

Nowadays, information filtering has become a difficult task due to the increasing size of web. There are millions of data that are uploaded to web everyday and hence efficient web page classification methods are needed to deal with rapidly growing user generated content [1]. The problem of web page classification is a complex task, as a web page contains not only the textual information, but also hyperlinks, images and multimedia. It poses a challenge in determining the category of the web page and the content based web page classification methods are not suitable for information filtering applications. So, we are in need of a simple classification method that can handle this large scale web data. URL based classification methods are the most preferred one to address these issues. As the URL is a small fraction of a web page and contains less information, it is highly challenging to select the suitable features from URL. In order to classify the webpage quickly, we perform URL based webpage classification by deriving n-gram features.

* Corresponding author.
E-mail address: rajalakshmi.r@vit.ac.in

1877-0509
Peer-review under responsibility of the scientific committee of the 7th International Conference on Advances in Computing & Communications
10.1016/j.procs.2017.09.128

In text classification, feature selection is normally used as a dimensionality reduction technique, in which a small subset of relevant features are chosen removing the noisy and redundant ones. In case of feature selection, all the features are ranked based on some criteria and the features with high scores are selected. As the URL contains very few words, we do not remove the terms based on a threshold, but all the terms are considered along with their weights. It is evident from the existing literature [2, 3] that, the supervised term weighting methods improve classification accuracy. We prefer supervised feature weighting method, as it takes the category information also into account for data representation.

Many different approaches have been developed for URL based web page classification by various researchers [4, 5, 6, 7, 8, 9, 10, 11, 12]. Khan et. al [4, 5] used segmentation based approach for deriving features from URLs, which is a complex one. Baykan et.al [6, 7] proposed the use of n-gram (n = 4 to 8) features, but it does not scale with growing data. A heuristics dictionary based approach is followed by Rajalakshmi et. al [8] by deriving token features from URLs, but results in a large dictionary, the size of which depends on the training data. In another work [9], 3-gram features were considered to limit the dictionary size. To provide better scalability on large dataset, n-gram language model based method has been suggested by Tarek [11]. However, in all the above techniques feature weighting has not been given importance. The impact of the term weighting methods for URL classification in multiclass scenario was studied in our previous work [12] and the results showed the performance improvement while using Naive Bayes Classifier.

In this paper, we try to find out the suitable feature weighting method for binary classification of URLs by conducting experiments on a small dataset using SVM as the classifier. We performed experimental study of different feature weighting methods on tokens and character n-grams (n = 3, 4 and 5-grams) deriving from URL, by considering WebKB dataset. We have implemented different feature weighting techniques such as tf, tf.idf, tf.pidf, tf.dsidf and tf.mi and observed its performance over different character n-grams and token features. Results show that tf.mi outperforms other weighting methods.

The remainder of the paper is organized as follows. In section 2, we briefly discuss the related work. Section 3 gives a review of the different existing feature weighting techniques. Section 4 discusses the materials and methods used for our experiments in detail. Section 5 discusses experiment and results obtained from the feature weighting methods. Section 6 shows the comparison between the proposed method and the existing works. Finally in Section 7 we give our main conclusions and future work.

2. Related Work

Web page classification problem has been studied by many researchers and different techniques have been suggested in the literature. As the content based classification schemes are not suitable, Kan [4] suggested an approach in which the web pages can be classified with out downloading its contents. They used segmentation / expansion approach for classifying the web pages by deriving feautures from URLs. They also tried with features such as title text, anchor text and page text that are derived from the source document. They conducted experiments on a smaller data set WebKB and reported that, URL features based classifier is the best one compared to classifier that used source document based features. They designed a series of binary classifiers to classify course, faculty, student and project web pages of WebKB dataset. Baykan et. al [6, 7] have performed a detailed analysis of URL classification by considering various features such as tokens and all n-grams (n = 4 to 8). They have experimented with different datasets such as WebKB, ODP, Yahoo and Delicious with different classifiers such as Naive Bayes, SVM and Maximum Entropy classifier. They reported that, all-grams features are the best combination of features with Maximum Entropy classifer for this URL classification problem. Rajalakshmi et. al [8] performed direct multiclass classification on ODP dataset by deriving token features from the URLs and achieved an F_1 measure of 0.76. In another work [9], 3-gram features were alone derived from URLs and binary classification was performed using two datasets viz., a large ODP dataset and a small WebKB dataset. A dictionary learning method was proposed in [10], deriving 4-gram features of URLs for identifying the health domain web pages by conducting experiments on ODP dataset. The importance of supervised feature weighting methods for URL classification was studied for multiclass classification of URLs in our previous work [12]. It was shown by experimental results that, the peformance of the classifier improves when feature weighting methods are applied.

Various feature weighting methods are suggested in the literature for text classification. The most widely used term weighting schemes for classification are binary, tf and tf-idf. For question categorization, various term weighting methods have been suggested by Quan et. al [13], and they found that variant of relevant frequency performs better than other methods. Haibing Wu et. al [14] discusses about different local and global weighting schemes. They have suggested that, larger weights should not be assigned to imbalanced terms by the supervised term weighting methods and there should be a trade-off between over-weighting and under-weighting. They have used regularization techniques such as add-one smoothing, sub linear scaling and bias term to reduce over weighting and studied the performance on datasets such as IMDB and RT-2k.

In this paper we have implemented weighting schemes viz., tf, tf.idf, tf.pidf, tf.dsidf and tf.mi on WebKB dataset and compared the F1 measure for each method. We have analyzed the effect of different weighting methods on different features such as tokens, 3-grams, 4-grams and 5-grams that are derived from URLs for binary classification scenario.

3. Survey of feature weighting methods

In text classification, the documents are represented in the vector space model as terms. Depending on the importance of a term, each vector component is assigned a weight. The tf-idf weighting scheme, which is widely used in information retrieval not uses the class information. These kind of unsupervised weighting methods can be replaced by supervised weighting methods by considering the category information into account [2]. In traditional feature selection methods, a cut-off threshold is used to select the features based on a score and these weights/scores are not considered. But, these supervised weights can be considered and it changes the term space representation of the documents. It assigns lower weights to irrelevant terms and larger weights to predictive terms, thereby improving the performance of the classifier. In [14], various weighting methods such as tf, idf, pidf, mi and dsidf have been discussed. In this paper, we have applied the variants of tf and experimented with the following feature weighting methods viz., tf.idf, tf.pidf, tf.mi and tf.dsidf.

3.1. Term frequency (tf)

Term frequency is defined as the number of times term t appears in a document. It is an example of local term weighting scheme and it is called as raw term frequency. Local weight is obtained only from frequencies within the text. The terms appearing more in a document, is given importance in case of tf. It is an active weighting scheme which is used widely for text categorization. We adopted this method for URL classification also to perform a comparative study.

The following notations are used in the weighting schemes discussed below. a denotes the number of training URLs in positive category containing term t, b denotes the number of training URLs in the positive category which does not contain the term t, c denotes the number of training URLs in the negative category containing term t, d denotes the number of training URLs in negative category that do not contain term t. N is used to denote the total number of URLs in training document collection ($N=a + b + c + d$). N1 the number of training URLs in the positive category and N2 is the number of training URLs in the negative category.

3.2. Inverse Document Frequency (idf)

Inverse document frequency (idf) is defined as the logarithmic value of total number of URLs divided by number of URLs with term t_i in it. It is intended to reflect how important a word is in a document in a collection. In case of idf rare terms are more effective and must be used in discriminating URLs. Mathematically it is given by the following equation.

$$\text{idf} = \left[log_2\left(\frac{N}{a+c}\right)\right] \tag{1}$$

3.3. Probabilistic idf (pidf)

Probabilistic idf is a variant of idf, which follows probability distribution. Mathematically it is given by the following equation

$$\text{pidf} = \left[log_2\left(\frac{N}{a+c} - 1\right)\right] \tag{2}$$

3.4. Mutual Information (mi)

Mutual information is used for ranking features; it is especially used in cases when the number of features in the input vector is larger than 3 or 4 features. It is used to measure the association between a term and specific topic of interest. It works well on both balanced and unbalanced sets and gives better results on imbalanced datasets. It is a measure of association used in information theory and statistics, it measures how much the feature associates with the class.

$$\text{mi}= \left[log_2 \left(max \left(\frac{aN}{(a+c)N1}, \frac{cN}{(a+c)N2} \right) \right) \right] \tag{3}$$

3.5. Delta smoothened idf (dsidf)

Weighting method dsidf is a more sophisticated global weighting method, it helps in reducing computational errors by the introduction of smoothing function. It helps in boosting the importance of terms that are unevenly distributed between one category and other categories. Mathematically it is given by the following equation.

$$\text{dsidf} = \left[log_2 \frac{N2\,(a+0.5)}{N1(c+0.5)} \right] \tag{4}$$

In our work, we have used variants of *tf* such as *tf.idf*, *tf.pidf*, *tf.dsidf* and *tf.mi* and have evaluated its performance on tokens and character n-grams, where n = 3, 4 and 5 by deriving these features from URLs.

4. Materials and methods

4.1. Data collection

The main objective of our experiment is to find the suitable feature weighting method for URL based webpage classification. For conducting this experimental study, we have used WebKB corpus, the benchmark dataset that is widely used for web page classification problem. It contains pages collected from the Computer Science department in four Universities. It consists of 4,199 URLs from the Universities of Cornell, Texas, Washington and Wisconsin. Among the seven categories of web pages (course, faculty, student, project, staff, department and other), we have considered only four categories viz., course (930), faculty (1124), student (1641) and project (504) in our experiment. For each category, the number of URLs considered from the four universities are as follows: Cornell (867), Texas (827), Washington (1205) and Wisconsin (1263). The experiment was performed on a system with Intel Xeon Quad Core HT 3.5 GHz processor with 32GB RAM. Table 1 shows the data split for training and testing samples on WebKB dataset. We have used 70% of URLs for training the classifier and the remaining 30% for testing. The training data for SVM classifiers are balanced to get equal proportion of positive and negative training examples. We have done training on three of the universities and testing on the fourth held-out university. All experiments use leave-one-university-out for evaluation [15].

Table 1: WebKB Dataset

Class Label	Training Samples	Testing Samples
Course	651	279
Faculty	786	338
Project	352	152
Student	1148	493

We performed binary classification and the four binary classifiers are Course, Faculty, Student and Project. For example, the course classifier determines whether the given URL is a course page or not. We used Python for implementation and SVM^{light} [16] tool for classification.

4.2. URL features and weighting methods

We considered tokens and character n-grams (n = 3 to 5) from URLs as features. URLs are first preprocessed, in which http, www, and all special characters are removed. Each URL is then concatenated to a single word from which character n-grams are derived. For example, if we consider the URL https://www.cs.cornell.edu, after preprocessing and concatenation, it becomes cscornelledu. The derived n-grams are as follows:
3-grams : csc sco cor orn rne nel ell lle led edu
4-grams : csco scor corn orne rnel nell elle lled ledu
5-grams : cscor scorn corne ornel rnell nelle elled lledu

Then it is represented in vector space model. Feature weighting methods such as tf, tf.idf, tf.pidf, tf.mi and tf.dsidf are applied on these features separately.

5. Experiments and results

Precision (P) and Recall (R) are the two measures widely used for webpage classification. F_1 measure is used as the performance metric which is the harmonic mean of precision and recall. It is given by the following equation:

$$\mathrm{F}_1 = (2 * P * R)/(P + R) \tag{5}$$

The effect of different weighting methods on different features viz., tokens, 3-grams, 4-grams and 5-grams, in case of different leave-one-university-out for the course classifier is shown in Tables 2 to 6.
As shown in Table 2, when tf alone is used for feature weighting, 5-gram feature performs better than token, 3-gram

Table 2: F_1 measure using tf weighting scheme for course as classifier

			tf		
Features	Wisconsin	Texas	Cornell	Washington	Macro Average F_1
Tokens	40	32	49	51	43
3gram	60	62	58	42	55
4gram	60	64	60	40	56
5gram	60	60	70	62	**63**

and also 4-gram features with a macro-average F_1 of 63%. We get the best results for tf, using Wisconsin and Cornell as leave-one-university-out. In Table 3, it is observed that for tf.idf scheme, 4-gram feature performs better than token, 3-gram and 5-gram and a macro average of 70% was achieved. We get the best results for tf.idf, using Wisconsin and Washington as leave-one-university-out.
From Table 4, it is observed that for tf.pidf scheme, 3-gram feature performs better than other features with a macro

Table 3: F_1 measure using tf.idf weighting scheme for course as classifier

			tf.idf		
Features	Wisconsin	Texas	Cornell	Washington	Macro Average F_1
Tokens	50	38	44	46	44
3gram	77	62	65	73	69
4gram	77	63	70	71	**70**
5gram	74	61	73	64	68

average F_1 of 68%. We get the best results for tf.pidf, using Wisconsin and Washington as leave-one-university-out.

In Table 5, it is observed that for tf.dsidf scheme, 5-gram feature performs better than token, 3-gram and also 4-gram. It can be seen that using tf.dsidf for 5-gram feature gives a macro average of 71%. We get the best results for tf.dsidf, using Wisconsin and Washington as leave-one-university-out.

As shown in Table 6, while using tf.mi feature weighting method, the performance of n-grams is better than token features. The best performance with a macro-average of 79% was achieved for 5-gram features when using Wisconsin

Table 4: F_1 measure using tf.pidf weighting scheme for course as classifier

tf.pidf					
Features	Wisconsin	Texas	Cornell	Washington	Macro Average F_1
Tokens	57	36	49	44	46
3-gram	81	63	63	68	**68**
4-gram	78	64	64	61	66
5-gram	75	62	64	46	61

Table 5: F_1 measure using tf.dsidf weighting scheme for course as classifier

tf.dsidf					
Features	Wisconsin	Texas	Cornell	Washington	Macro Average F_1
Tokens	40	49	58	47	49
3gram	76	53	50	48	56
4gram	79	50	48	46	55
5gram	78	63	70	74	**71**

Table 6: F_1 measure using tf.mi weighting scheme for course as classifier

tf.mi					
Features	Wisconsin	Texas	Cornell	Washington	Macro Average F_1
Tokens	51	84	56	57	62
3-gram	81	81	78	76	**79**
4-gram	81	81	80	74	**79**
5-gram	82	82	73	78	**79**

and Texas as leave-one-university-out.
On comparing tables 2 to 5, we observe that tf.mi weighting method performs better than the other methods. 5-gram feature gives better results in most cases and better results are obtained for Wisconsin as leave-one-university-out university.

We observed that, among the considered five feature weighting methods, tf.mi performs better than other methods. We present the performance of all the four binary classifiers for the best performing feature weighting technique tf-mi in Table 7, while considering Wisconsin as leave-one-university-out. It can be seen that Course classifier performs better than all the other categories faculty, project and student. Among the features, tokens give the worst result of 49%, whereas 3-gram features give better result than tokens. It can be seen that 4-gram and 5-gram URL features give a similar performance with a macro average of 79%. The 5-gram and 4-gram features outperforms 3-grams and token features. We can observe that, the value of F_1 measure increases, as the value of character n-grams increases from 3 to 5. From the experiments conducted, we conlcude that, course classifier performs the best classification, along with tf.mi as the feature weighting method and Wisconsin as the leave-out university.

Table 7: Comparing F_1 measure for WebKb dataset

Classifier	Course	Faculty	Project	Student	F_1
Tokens	53	50	48	45	49
3gram	81	80	78	75	78
4gram	81	79	76	79	**79**
5gram	82	80	76	78	**79**
Average	**74.5**	72.25	69.5	69.25	71.25

6. Comparison with existing methods

We have compared our proposed method with the works done in this direction and reported in Table 8. In [9], they have used tf with 3-gram approach in WebKB datasets. In the case of course as classifier, it can be seen that they were able to achieve an F_1 measure of 48%. In our experiments, when 3-gram URL features were used on different weighting schemes, we were able to get better F_1 measure. Along with the weighting methods viz., tf, tf.dsidf, tf.pidf, tf.idf , tf.mi, we achieved 55%, 56%, 68%, 69% and 79% respectively.

In [11], n-gram language modelling (LM) scheme was used. While using 4-gram language modelling on WebKB dataset, they were able to achieve an F_1 measure of 59.25%. Baykan et.al [6], reported that using all-gram URL features (4, 5, 6, 7 and 8-grams) macro average of 66.5% was obtained. It can be seen that the performance of the proposed approach with tf.mi feature weighting method outperforms all the other existing works in the literature with a macro average of 79%. Our experimental study conducted on WebKB dataset clearly shows that, 5-grams features

Table 8: Performance comparison of proposed binary classification with existing methods

	F_1
SVM with 3-grams [9]	55%
4-gram LM SVM [11]	59.25%
SVM with all-grams (n = 4 to 8) [6]	66.5%
Feature weighting (Proposed approach)	**79%**

using tf.mi method is the best one among the five weighting methods considered. All the existing methods have not used feature weighting methods for URL classification and it can be applied to improve the classification accuracy.

7. Conclusion

Classification of web pages based on URL is useful as it avoids fetching the web pages unnecessarily for classification purpose. The classification can be done on the fly and it can be used for information filtering purpose also. We have chosen a small dataset (WebKB) that has pages collected from four universities and performed binary classification by considering the four categories viz., course, faculty, project and student. We derived tokens and character n-gram (n = 3, 4 and 5) features from these URLs and applied different feature weighting methods. We analysed the performance of these URL features on different feature weighting methods viz., tf, tf.idf, tf.pidf, tf.dsidf and tf.mi. Experimental results show that tf.mi feature weighting method performs the best when it is applied on 5-gram features. Using the tf.mi weighting method, we achieved an F_1 measure of 79% and it outperforms all the other methods in the existing literature. In our future work, we plan to implement this method on a larger dataset.

Acknowledgements

We would like to thank the Department of Science and Technology - Science and Engineering Research Board, Government of India for funding this work (Award Number : ECR/2016/000484). We would also like to thank the management of VIT University, Chennai for extending their support, where this research work was carried out.

References

[1] Renato M. Silva, Tiago A. Almeida, and Akebo Yamakami (2017) MDLText: An efficient and lightweight text classifier. *Knowledge-Based Systems* **118**:152 – 164.

[2] Iyad Batal and Milos Hauskrecht. (2009) Boosting KNN text classification accuracy by using Supervised term weighting schemes. *In : Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM 09, 2041 – 2044.

[3] Man Lan, Chew Lim Tan, Jian Su, and Yue Lu. (2009) Supervised and traditional term weighting methods for automatic text categorization. *IEEE transactions on pattern analysis and machine intelligence* **31** (4):721 – 735.

[4] Min-Yen Kan. (2004) Web page classification without the web page. *In : Proceedings of the 13th International World Wide Web Conference on Alternate Track Papers and Posters,* WWW Alt. 04, 262 – 263.

[5] Min-Yen Kan and Hoang Oanh Nguyen Thi. (2005) Fast webpage classification using URL features. *In :Proceedings of the 14th ACM International Conference on Information and Knowledge Management* CIKM '05, 325 – 326.

[6] Eda Baykan, Monika Henzinger, Ludmila Marian, and Ingmar Weber. (2011) A comprehensive study of features and algorithms for URL-based topic classification. *ACM Trans. Web* **5** (3):15:1 – 15:29.

[7] Eda Baykan, Monika Henzinger, Ludmila Marian, and Ingmar Weber. (2009) Purely URL-based topic classification. *In : Proceedings of the 18th International Conference on World Wide Web* WWW 09, 1109 – 1110.

[8] R Rajalakshmi and C Aravindan. (2011) Naive bayes approach for website classification. *Communications in Computer and Information Science* , **147** :323 – 326.

[9] R Rajalakshmi and Chandrabose Aravindan. (2013) Web page classification using n-gram based URL features. *In : Proceedings of Fifth International Conference on Advanced Computing (ICoAC)* 15 – 21.

[10] R Rajalakshmi. (2015) Identifying Health domain URLs using SVM. *In : Proceedings of the Third International Symposium on Women in Computing and Informatics* WCI 15, 203 – 208.

[11] Tarek Amr Abdallah and Beatriz de La Iglesia. (2014) URL-based web page classification: With n-gram language models. *In : Proceedings of the International Conference on Knowledge Discovery and Information Retrieval* **553** : 19 –33.

[12] R Rajalakshmi. Supervised term weighting methods for URL classification. (2014) *Journal of Computer Science* **10** (10):1969 – 1976.

[13] X. Quan, L. Wenyin, and B. Qiu. (2011) Term weighting schemes for Question categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. **33** (5):1009 – 1021.

[14] Haibing Wu, Xiaodong Gu, and Yiwei Gu. (2017) Balancing between Over-weighting and Under-weighting in Supervised term weighting. *Inf. Process. Manage.* **53** (2):547 – 557.

[15] Aixin Sun, Ee-Peng Lim, and Wee-Keong Ng. (2002) Web classification using support vector machine. *In : Proceedings of the 4th International Workshop on Web Information and Data Management* WIDM '02, 96 – 99.

[16] Thorsten Joachims. (1999) Making Large-scale Support Vector Machine Learning Practical. *Advances in Kernel Methods* 169 – 184.