URL Classification Using Non Negative Matrix Factorization

Shreya Khare, Akshay Bhandari and Hema A. Murthy
Department of Computer Science and Engineering
Indian Institute of Technology Madras
Chennai, India
E-mail: {shreya, akshayb, hema}@cse.iitm.ac.in

***Abstract*—Internet availability on a campus is not metered. Internet link bandwidths are vulnerable as they can be misused. Moreover, websites blacklist campuses for misuse. Especially blacklisting by academic websites like IEEE and ACM can lead to serious researchers being denied access to information. The objective of this paper is to proactively classify anomalous accesses. This will enable campus ISPs to deny access to users, misusing the Internet. In particular URLs are classified using the short snippets(meta-data) that are available. New Features, namely** *random walk term weights, within class popularity* **in tandem with** *non negative matrix factorization* **show a lot of promise for classifying URLs. The classification accuracy is as a high as 92.96% on 10 gigabytes of proxy data.**

Keywords- Web Page Classification, Non Negative Matrix Factorization

I. INTRODUCTION

Use of Internet in college campuses has increased drastically in recent years, leading to pathological use or Internet addiction. Campus wide availability of Internet has triggered a drift from the primary focus of the Internet namely data sharing for academic purposes to abusive use of Internet especially for videos and entertainment based content which consumes a major share of the bandwidth. This has forced the administrators to exercise certain measures to ensure that all users get a fair share of the bandwidth. Shaping of network traffic by blocking youtube, facebook, etc or restricting the time of usage prevents the user from accessing useful information that is also available through these channels.

The primary objective of any policy on a campus is that it should be seamless while at the same time exercising necessary control. The goal of this paper is to exploit the data from proxy logs to classify web content which can be used to shape traffic. Earlier efforts in controlling traffic primarily relied on classifying users based on their share of bandwidth [1]. A standard control mechanism adopted by most companies is to block sites video hosting websites namely youtube, etc. Today, youtube is not primarily used for entertainment as huge amount of academic content is readily available eg: NPTEL [2]. These accesses involves high bandwidth usage similar to that required for entertainment based content. Therefore, a finer classification of URLs is required. The content of the URL request must be classified and appropriate control policies must be developed to restrict access to sites that are purely for entertainment purposes. This kind of control may be exercised at the application layer of a proxy server.

978-1-4799-2361-8/14/$31.00

In many campus installations, access to the Internet is controlled by a password protected proxy server [3]–[5]. In [3], the authors developed a proxy based prediction service. The service was able to capture both inter-site and intra-site access patterns. They incorporated a mechanism that could exploit reference locality in proxy logs. On the similar lines, [4] tried to remove noisy transactions by clustering the user access to websites. They assumed that a user visits multiple related websites, thus enabling cluster formation. [6] analyzed the proxy logs to discover the amount of time spent by user on the Internet, and proposed certain control mechanisms based on the traffic pattern generated.

We propose a framework to comprehend the intention of usage of Internet. This intention is captured by the category of URL requested. URL categorization is expected to play a critical role in understanding the user behavior. This can be deemed as a problem of Web page classification. Web page classification is considered as a supervised learning problem which aims to categorize Web pages into a set of predefined categories based on labeled training data.

In addition to text classification of Web content, [7] emphasized the importance of Web-specific features and algorithm for classification of Web pages. This approach attributes to presence of loosely structured text, lack of consistency and finally abundance of noisy and misleading information in a Web page. Researchers have used content or subjects to perform topic and genre based classification of Web pages [8]. They analyzed the content of the Web pages, along with URLs, HTML tags, Java scripts and VB scripts and developed a genre based Web page classification system. This system was able to classify 1000 web pages into four categories namely online shopping, discussion forum, university homepage and frequently asked questions with an accuracy of 93%. [9] deployed features like intended audience, function of the Web pages, the type of links contained in them and relationship among them for classifying a Web page into categories like organizational pages, documentation, text, homepage, multimedia, database entry, and tools. [10] proposed a mechanism considering only lexical and host based features associated with a url and framing ground rules for classification purposes. [11] represented a web page with a vector of features and perform principal component analysis and Naive Bayes Classification to classify 4338 web pages with an accuracy of 92%.

In our study, the meta-data embedded in an HTML page

is used to perform Web page classification. [12] showed that meta-data is a useful feature that relates to the keywords in the Web page. The motivation for using meta-data lies in the fact that data present in the Web page is huge, noisy and difficult to process. For the case of video-hosting or flash based websites, owing to low textual content and on-page features,, meta-data can provide some characteristics of the content embedded in the flash object or video.

The paper is organised as follows Section II describes the dataset used for developing the framework to classify URLs. Section III describes the text mining based techniques to classify URLs based on the meta-data obtained for each URL. Section IV describes the classification. Section V provides details about the experiments and results. Finally, conclusions and future work are described in Section VI.

II. Dataset Used

A proxy server acts as an intermediary for requests from clients to other servers across the Internet. A proxy server is generally used in a university campus environment to reduce bandwidth usage, improve response time by caching Web documents and to perform content filtering. It also logs the details of all requests from the client. Proxy Logs contain a lot of information about the types of requests and access patterns of users. For each request, it stores the time stamp, content-type, size, URI information.

The proxy logs were parsed for extracting URLs. For each URL, the meta-data was obtained by parsing HTML Web-page. In order to train the model, a labelled dataset was developed using the McAfee(TM) Trusted source web database [12]. The database contains URLs organized into 104 categories. For the purpose of this work the number of categories were reduced to 44 manually, thus reducing sparsity across categories. Database includes categories like Entertainment, Social Networking, Pornography, etc. URLs under similar categories were grouped together to form a more general category. For instance, URLs under visual search engine were merged to search engine category. Figure 1 indicates the distribution of various URLs across different categories.

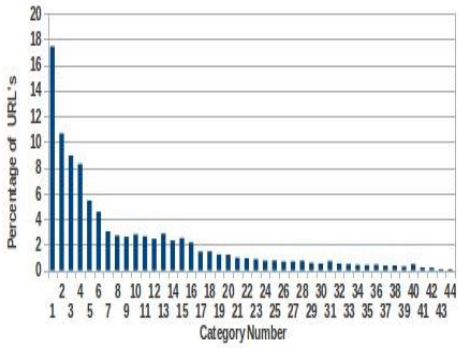


Fig. 1: Distribution of URLs across different Categories.

III. Proposed Work

Meta-data obtained from the urls are very noisy and contain a lot of irrelevant information. For a classifier to work well, the data must be structured appropriately. Hence for obtaining such a representation the following tasks were performed.

A. Text Preprocessing

The continuous character stream obtained from the meta-data of the URLs was converted to meaningful linguistic units (words). The text was converted to lowercase and tokens were extracted using regular expressions. This step of preprocessing known as tokenization was followed by the removal of topic-neutral (stop) words such as articles, prepositions, conjunctions, etc (such as 'the', 'a','of', 'and') . Stemming was later performed, to merge the words with the same root . Stemming mapped all the inflected forms of the word like parts of speech and plural sense into the root of the word. Stemming was performed using Word Net [13] which uses heuristics based approaches like Porter's stemming [14] and combines words with same root. Due to the noisy structure of the data, a number of spelling errors were present. A spell checker was used to correct these spelling errors. In case of multiple suggestions from the spell checker, the existence of the suggested word was checked across the vocabulary built. While, in case of no suggestions recursive segmentation was applied and matched with vocabulary built. Dealing with Web data, entities and proper nouns play a major role in categorization of Web pages of the URL's. These entities were categorized using freebase database [15]. In essence, the sequential process of tokenization, stemming, spell checking and entity extraction are the essential steps for structuring the Web data obtained from the meta-data of the URLs.

B. Text Representation using Feature Weights

Vector space model has been a very popular model used to represent text documents. Using this model, a document is represented as a vector, whose components are the weight that we assign to each term in a document [16]. In our study, document refers to the meta-data of a particular URL. The terms extracted after preprocessing are referred to as features of the documents. Henceforth, the document (D_j) can be represented as specified in Equation 1

$$D_j = \{w_1, w_2, w_3, w_4, ...w_m\} \tag{1}$$

Term Frequency i.e. frequency of a word within a document is commonly used to represent documents. This term frequency factor ignores the distribution of terms across the collection of documents. The terms which have a low frequency in one document, but are present across a set of documents in the corpus gets ignored. To overcome this problem, Inverse Document Frequency factor (IDF) is calculated. The IDF factor varies inversely with the number of documents to which a term is assigned in a collection of documents. The IDF of a term w in a document d can be calculated as

$$idf(w,d) = log\frac{|D|}{df(w,D)} \tag{2}$$

where $|D|$ is the number of documents in the corpus and $df(w,D)$ is the number of documents in which the word w appears.

Within Class popularity (WCP) seems to be a useful technique for the purpose of document classification. This feature weight addresses the issue of uneven distribution of prior class probabilities [17]. Using WCP, global goodness of a term is derived as a weight representing the distribution of a term across all classes of documents.

For computing WCP, initially the popularity weight of a feature f within a class c_i is calculated as

$$Pr(f|c_i) = \frac{1 + N(f, c_i)}{|V| + \sum\limits_{f \epsilon V} N(f, c_i)} \tag{3}$$

where $N(f, c_i)$ is the number of occurrences of a term f in all the documents in c_i and $|V|$ is the cardinality of vocabulary set. Equation 3 is used to create a normalized sample space of equal size without altering the intra class feature distribution. This popularity weight obtained, is then formalized across all the classes to obtain within class popularity as below

$$wcp(f, c_i) = \frac{Pr(f|c_i)}{\sum\limits_{k=1}^{|C|} Pr(f|c_i)} \tag{4}$$

where C is the set of class labels. The characteristics of these weights is that if $wcp(f, c_1) > wcp(f, c_2)$, then it can be concluded that term f is present more densely in the class c_1 than in class c_2. This exemplifies the usefulness of this score for classification of documents into classes.

A term weighting scheme based on random walk was studied to quantify the term co-occurrences within a document. In this scheme, the local contribution of a term is measured by capturing the locality of a term and its relation to surrounding context. Term co-occurence is computed by obtaining a graph encoding for each document with unique terms as vertices. An edge is drawn if the two nodes fall in the vicinity of a window. Each term is assigned a score using Random walk algorithm as mentioned in [18].

The score of a term is obtained from a graph, $G = (V, E)$ where the set of vertices is V and the set of Edges is E, s.t E is a subset of $V \times V$. For a vertex (V_a), let $In(V_a)$ be a set of vertices that point to Vertex V_a (predecessor), and Out (V_a) be the set of vertices that vertex V_a points to (successors) . The score of a vertex (term) is defined as

$$S(V_a) = (1 - d) + d. \sum_{V_b \epsilon In(V_a)} \frac{S(V_b)}{|Out(V_b)|} \tag{5}$$

where d is a damping factor, which is set between 0 and 1 [19].

C. Feature Extraction Based on Non Negative matrix factorization

The feature vector space obtained from the text spans a very high dimensional space. At times, this feature vector space can be very sparse. Besides the computational load, owing to the high dimensional data there is an abundance of noisy features. This results in an increase in the number of incorrect classification, as the discrimination across classes may not be significant. A dimension reduction is therefore required for meaningful classification [20].

Since, input data reduces to a matrix, several low rank approximation algorithms based on dictionary learning can be employed for dimensional reduction. In Dictionary Learning based methods, training data is used to learn a dictionary over which training set admits a maximal sparse representation. Dimensionality reduction is achieved by selecting a subset of functions from a larger dictionary. These functions can then be described as the reduced dimension space for classification tasks. The popular approaches for dimension reduction includes Singular Value Decomposition (SVD).

Singular Value Decomposition (SVD) [21] factorizes a matrix (X) of size $m \times n$ into three matrix described as

$$X = U \Sigma V^T \tag{6}$$

where Σ is a diagonal matrix containing the singular values while columns of U and V are orthonormal columns. Dimension reduction can be achieved by setting reducing the value of k, (where $k << n$) singular values in Σ to zero. This results in a reduced model as

$$X_{reduced} = U_{mk} \Sigma_{kk} V^T_{kn} \tag{7}$$

The choice of k is crucial. Ideally, k is chosen large enough to fit all the real structure in the data but small enough so that unimportant details are removed [20].

However, one of the pitfalls of using SVD is that the decomposition produces both negative and positive coefficient values in U and V. Negative coefficients imply possible cancellations in linear combination giving rise to difficulties in interpretation. In the textual context, when a term is not present in a document then the corresponding coefficient should be strictly zero or at least a very small positive value. There is no interpretability of a term being negatively present in reduced dimension. An alternative to SVD is dictionary based Non Negative Matrix Factorization (NMF). The fundamental difference between SVD and NMF is that the coefficients of basis vectors are positive in NMF while they are not guaranteed to be positive in SVD. As a result, NMF enables model interpretability in reduced dimension by imposing non negativity constraints on all the coefficients.

NMF is a popular algorithm for non negative dictionary learning. It forms additive parts-based representation of the data [22]. The low rank approximation of feature vector space representing a set of documents as a $m \times n$ matrix, where m are the features and n is the number of documents, can be found using NMF. NMF decomposes this term-document matrix into two matrices W and H, of dimension $m \times k$ and $k \times n$ respectively, where $k << (m, n)$.

The matrix W is called the basis document vector spanning m dimensional space and H weight matrix which give relevance to different elements of the basis vectors of H. NMF is usually used to approximate X by computing a pair of W and H to minimize the Frobenius norm of the difference $X - WH$. Mathematically, the problem can be formulated as follows:

1) Let $X \in R^{m \times n}$ be a data matrix of non negative entries.

2) Let $W \in R^{m \times r}$ and $H \in R^{r \times n}$ for some positive integer $r < n$.
3) The objective is then to solve the optimization problem.

$$min_{W,H} \|X - WH\|_f^2 \quad (8)$$

The matrices W and H obtained are not unique and initialized to random values with a non negativity constraint. The NMF algorithm used in this study is Gradient Descent with Constrained Least Squares (GD-CLS) [23]. This algorithm requires lesser number of iterations until convergence, and comparatively less work is required per iteration than most other NMF algorithm [24]. This algorithm computes weight matrix H using a constrained least square mode in-order to penalize the non smoothness and non sparsity in H. If H is maximum sparse such that it has only one non-zero element in each column, then NMF forces the column vectors of W to get closer to the data which describes the data better.

Algorithm for GD-CLS

1) Initialize W and H with non negative values and scale the columns of W to unit norm.
2) Iterate until convergence or until k iterations:

   a) $W_{ic} \leftarrow W_{ic} \frac{XH_{ic}^T}{(WHH^T)_{ic}+\epsilon}$,for c and i $[\epsilon = 10^{-9}]$
   b) Rescale the columns of W to unit norm
   c) Solve the constrained least square problem:

$$min_{H_j} ||X_j - WH_j||_2^2 + \lambda ||H_j||_2^2$$

where the subscript j denotes the j_{th} column for $j = 1, .., m$. Any negative values in H_j are set to zero. The parameter λ is a regularization value that is used to balance the reduction of metric $||X_j - Wh_j||_2^2$ with the enforcement of smoothness and sparsity in H.

For accurate results of NMF based GD-CLS, the number of iterations are decided by observing the Frobenius Norm of error matrix for each category [25].

IV. Classification

In-order to predict the categories of a Web page using the above mentioned feature weighing and extraction technique, Naive Bayes Classifier was used. This classifier is particularly suited when the dimensionality of data is high. The underlying assumption for this model is that each of the features are conditionally independent of another given some class and the parameters are assumed to be independent. In this study, Naive Bayes Model is used to compute class probabilities for a given document [26]. The advantage of this model is its simplicity, computational efficiency and good performance.

V. Experiments and Results

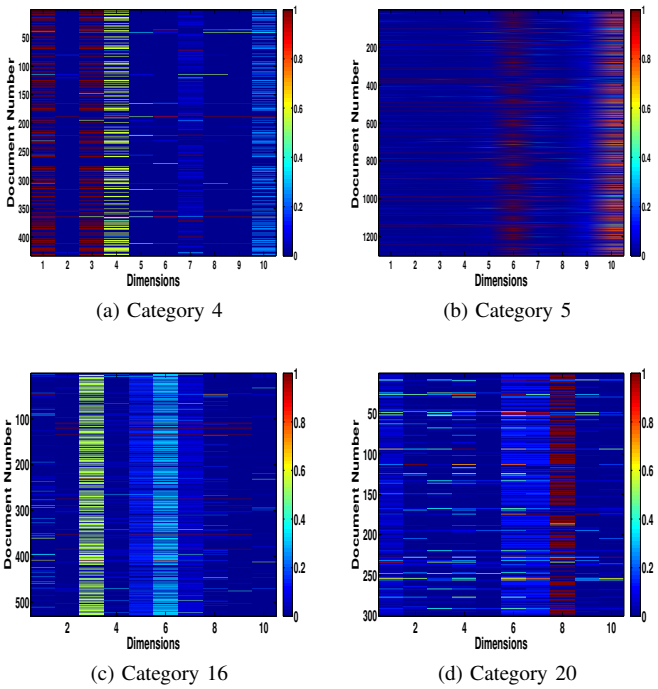
Proxy-server logs of user access requests were used for the experiments. The textual data extracted for each URL was preprocessed to build vocabulary and form term document matrix. Different feature weighing schemes were obtained, comprising of local (term frequency, random walk term weights) and global weights (inverse document frequency, within class popularity) for terms present in the vocabulary. The following composite schemes were considered for evaluation:

1) term frequency and inverse document frequency (TF-IDF)
2) term frequency and within class popularity (TF-WCP)
3) random walk term weights and inverse document frequency (RW-IDF)
4) random walk term weights and within class popularity (RW-WCP)

Afterwards a vector space model was obtained for each of the schemes.

The dimensionality of the vector space model was reduced by applying NMF based GD-CLS algorithm. The dimension of the reduced vector space (k) and initialization parameters (λ , W, H) of GD-CLS algorithm were determined empirically.

To appreciate the dimensional reduction performed by NMF, Heatmaps of the activations in the reduced space for each category were plotted. Heatmaps are general graphical representation of data (matrix) where each value of the matrix is shown as a color in the heat map. Heatmaps helps to visualize NMF's ability to identify patterns and capture the characteristics of the categories in the reduced space. Figure 2,3,4,5 show the heatmaps of weight matrix (H) for four feature weighing schemes.

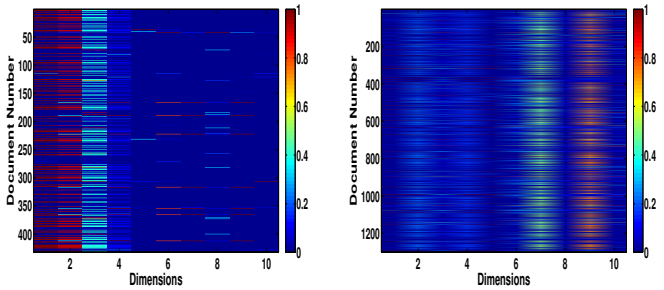


(a) Category 4 (b) Category 5

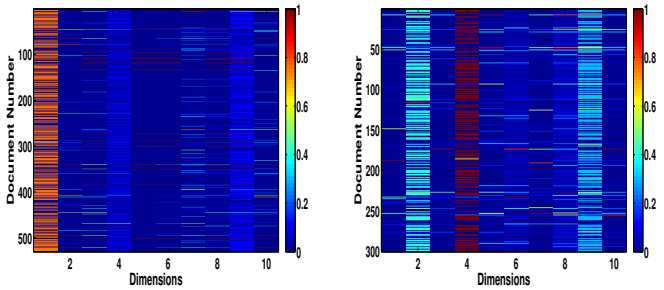
(c) Category 16 (d) Category 20

Fig. 2: Heat Map of the weight matrix H of RW-WCP.

It is clear from the plots that heatmaps for different categories are dissimilar. Each category has activations across different dimensions thus indicating cluster formation and separability. For a particular category, same dimensions are activated for majority of the urls. The small gaps in the predominant dimension of the heatmap for a category shows the absence of the particular feature for that URL. These plots also give insight into the performance of different feature weighting techniques. The color pattern in heatmaps shows

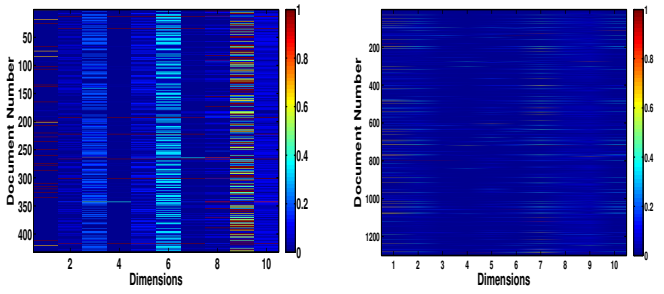


(a) Category 4 (b) Category 5

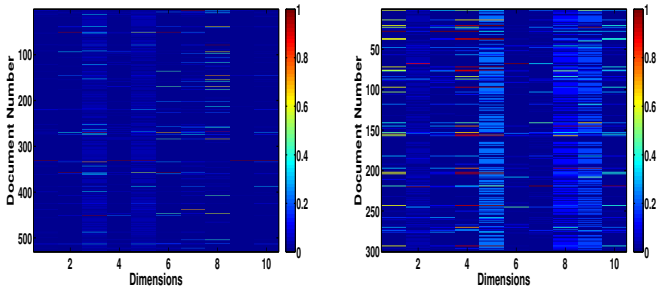


(c) Category 16 (d) Category 20

Fig. 3: Heat Map of the weight matrix H of RW-IDF.



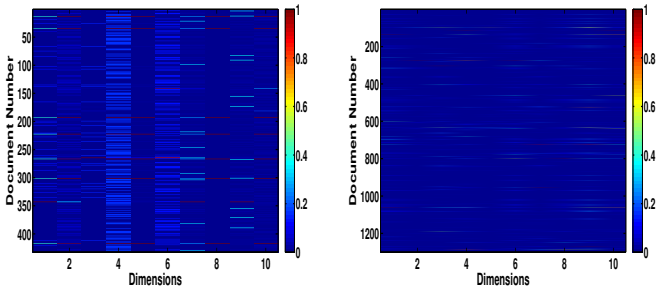
(a) Category 4 (b) Category 5



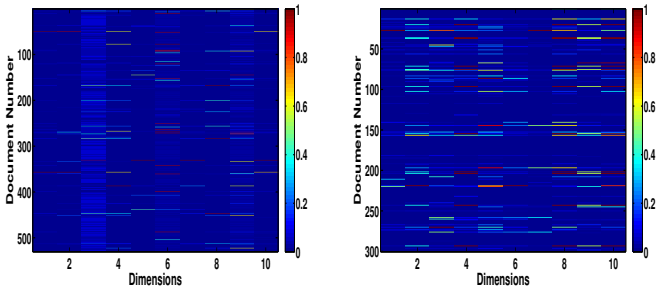
(c) Category 16 (d) Category 20

Fig. 4: Heat Map of the weight matrix H of TF-WCP.

less separability in reduced dimension for TF-IDF and TF-WCP weights as compared to RW-IDF and RW-WCP. This indicates importance of term correlation for classification.



(a) Category 4 (b) Category 5



(c) Category 16 (d) Category 20

Fig. 5: Heat Map of the weight matrix H of TF-IDF.

Further, evaluation of every feature weighting scheme was performed by computing the accuracy of Naive Bayes' classification on the dimensionally reduced data. For an accurate evaluation of the model, 10 fold cross validation was performed. F_1 measure was computed for better and comparative evaluation for each weighing scheme. F_1 measure is computed as

$$F_1 = 2.\frac{precision.recall}{precision + recall} \tag{9}$$

where precision is fraction of retrieved instances of relevant instances and recall is fraction of relevant instances of relevant instances. The classification results of dimensionally reduced data by NMF were compared with SVD. On obtaining the reduced data by SVD, test dataset was projected to the lower dimensions and classification was performed. The comparison assessed the effect of non negative coefficients obtained by NMF with positive and negative coefficients obtained from SVD. A large number of experiments were conducted by changing the initialization parameters and varying the number of reduced dimension from 5 to 100 in NMF and SVD. From these experiments it was observed that the best result was obtained at $k = 10$ and $\lambda = 0.2$. The results clearly indicate that the random walk weights in tandem with NMF shows considerable increase in accuracy in comparison with other weighing techniques. However, the accuracy of SVD is less as compared to that of NMF. Higher accuracy and F_1 score in NMF based classification indicates that presence of negative coefficients in SVD reduces the classification performance.

VI. Conclusion and Future Work

In this paper, a framework has been proposed to predict the category of Web page accessed by users according to the

Feature Weights	Feature Extraction	Accuracy (%)	F Measure
TF-IDF	NMF	89.62	0.8135
	SVD	74.49	78.19
TF-WCP	NMF	86.69	0.8456
	SVD	74.49	78.19
RW-IDF	NMF	91.35	0.8700
	SVD	75.82	78.21
RW-WCP	NMF	**92.96**	**0.8736**
	SVD	75.82	78.21

TABLE I: Accuracy and F_1 measures for model evaluation

short snippets available from URL. Solution provided here is unique as it uses meta-data and evaluates new composite scheme of feature weights. Using efficient dictionary learning and text mining based techniques as suggested, we can be proactive in classifying anomalous access. In the framework suggested, the new features random walk term weights, within class popularity as feature weight along with NMF for feature extraction and naive bayes classification has shown promising results. The ability of NMF to decipher the overlap across the classes, especially in the context of text classification, seems promising. This is an important aspect for classification of text, as topics have a considerable overlap. An overall accuracy of 92.96 % has been achieved in classification of the access to the Web. The primary reason for better performance, is that the term correlation captured by random walk term weights and popularity of this correlation in a particular category captures the separability among all the categories. These results can be used to comprehend the trend in the usage of Internet across the campus and thus devise policies for users and control Internet access. The category predicted can be combined with other features like bandwidth usage, time of access and others to develop bandwidth managements policies.

VII. Acknowledgement

This work was carried out under the IU-ATC project funded by the Department of Science and Technology (DST),Government of India, and the UK EPSRC Digital Economy Programme. The authors would like to acknowledge the use of computational resources at the High Performance Computing Facility at IIT Madras for this research work.

References

[1] S. Sait, M. Kumar, and H. Murthy, "User traffic classification for proxy-server based internet access control," in *Signal Processing and Communication Systems (ICSPCS), 2012 6th International Conference on*, 2012, pp. 1–9.

[2] [Online]. Available: http://nptel.iitm.ac.in/courses.php

[3] W. Lou and H. Lu, "Efficient prediction of web accesses on a proxy server," in *Proceedings of the eleventh international conference on Information and knowledge management*, ser. CIKM '02. New York, NY, USA: ACM, 2002, pp. 169–176. [Online]. Available: http://doi.acm.org/10.1145/584792.584823

[4] W. Lou, G. Liu, H. Lu, and Q. Yang, "Cut-and-pick transactions for proxy log mining," in *In: Proceedings of the 8th international conference on extending database technology (EDBT 2002). Prague, Czech Republic*. Springer-Verlag, 2002, pp. 88–105.

[5] M. Deshpande and G. Karypis, "Selective markov models for predicting web page accesses," *ACM Trans. Internet Technol.*, vol. 4, no. 2, pp. 163–184, May 2004.

[6] K. Bommepally, T. Glisa, J. J. Prakash, S. R. Singh, and H. A. Murthy, "Internet activity analysis through proxy log," in *Communications (NCC), 2010 National Conference on*. IEEE, 2010, pp. 1–5.

[7] X. Qi and B. D. Davison, "Web page classification: Features and algorithms," *ACM Comput. Surv.*, vol. 41, no. 2, pp. 12:1–12:31, feb 2009. [Online]. Available: http://doi.acm.org/10.1145/1459352.1459357

[8] B. Choi and Z. Yao, "Web page classification*," in *Foundations and Advances in Data Mining*. Springer, 2005, pp. 221–274.

[9] S. W. Haas and E. S. Grams, "Page and link classifications: connecting diverse resources," in *Proceedings of the third ACM conference on Digital libraries*, ser. DL '98. New York, NY, USA: ACM, 1998, pp. 99–107. [Online]. Available: http://doi.acm.org/10.1145/276675.276686

[10] M.-Y. Kan and H. O. N. Thi, "Fast webpage classification using url features," in *Proceedings of the 14th ACM international conference on Information and knowledge management*, ser. CIKM '05. ACM, 2005, pp. 325–326.

[11] Z. He and Z. Liu, "A novel approach to naive bayes web page automatic classification," in *Fuzzy Systems and Knowledge Discovery, 2008. FSKD '08. Fifth International Conference on*, vol. 2, Oct 2008, pp. 361–365.

[12] K. Golub and A. Ardö, "Importance of html structural elements and metadata in automated subject classification," in *Research and Advanced Technology for Digital Libraries*. Springer, 2005, pp. 368–378.

[13] G. A. Miller, "Wordnet: A lexical database for english," *COMMUNICATIONS OF THE ACM*, vol. 38, pp. 39–41, 1995.

[14] M. F. Porter, "An algorithm for suffix stripping," *Program: electronic library and information systems*, vol. 14, no. 3, pp. 130–137, 1980.

[15] Google. (2003) Freebase data dumps. [Online]. Available: https://developers.google.com/freebase/data

[16] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," in *INFORMATION PROCESSING AND MANAGEMENT*, 1988, pp. 513–523.

[17] S. R. Singh, H. A. Murthy, T. A. Gonsalves, H. Liu, H. Motoda, R. Setiono, and Z. Zhao, "Feature selection for text classification based on gini coefficient of inequality."

[18] R. Blanco and C. Lioma, "Random walk term weighting for information retrieval," in *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2007, pp. 829–830.

[19] S. Hassan, R. Mihalcea, and C. Banea, "Random walk term weighting for improved text classification," *International Journal of Semantic Computing*, vol. 1, no. 04, pp. 421–439, 2007.

[20] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American Society for Information Science*, vol. 41, pp. 391–407, 1990.

[21] G. Golub and C. Reinsch, "Singular value decomposition and least squares solutions," *Numerische Mathematik*, vol. 14, no. 5, pp. 403–420. [Online]. Available: http://dx.doi.org/10.1007/BF02163027

[22] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *In NIPS*. MIT Press, 2001, pp. 556–562.

[23] F. Shahnaz, M. W. Berry, V. P. Pauca, and R. J. Plemmons, "Document clustering using nonnegative matrix factorization," *Inf. Process. Manage.*, vol. 42, no. 2, pp. 373–386, Mar. 2006. [Online]. Available: http://dx.doi.org/10.1016/j.ipm.2004.11.005

[24] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca, and R. J. Plemmons, "Algorithms and applications for approximate nonnegative matrix factorization," *Computational Statistics & Data Analysis*, vol. 52, no. 1, pp. 155–173, 2007.

[25] R. Albright, J. Cox, D. Duling, A. N. Langville, and C. D. Meyer, "Algorithms, initializations, convergence for the nmf 2."

[26] V. M. Telecommunications and V. Metsis, "Spam filtering with naive bayes – which naive bayes?" in *Third Conference on Email and Anti-Spam (CEAS)*, 2006.