Web Page Structure Enhanced Feature Selection for Classification of Web Pages

B. Leeladevi
Research Scholar, Anna University
India

A. Sankar, PhD.
Associate Professor
PSG College of Technology
Coimbatore
India

ABSTRACT

Web page classification is achieved using text classification techniques. Web page classification is different from traditional text classification due to additional information, provided by web page structure which provides much information on content importance. HTML tags provide visual web page representation and can be considered a parameter to highlight content importance. Textual keywords are base on which Information retrieval systems rely to index and retrieve documents. Keyword-based retrieval returns inaccurate/incomplete results when differing keywords describe the same document and queries concept. Concept-based retrieval tried to tackle this by using manual thesauri with term co-occurrence data, or by extracting latent word relationships and concepts from a corpus. Semantic search motivates Semantic Web from inception for classification and retrieval processes. In this paper, a model for the exploitation of semantic-based feature selection is proposed to improve search and retrieval of web pages over large document repositories. The features are classified using Support Vector Machine (SVM) using different kernels. The experimental results show improved precision and recall with the proposed method with respect to keyword-based search..

General Terms

Web Mining

Keywords

Web Mining, Feature extraction, Inverse document frequency, HTML Tag, Support Vector Machines

1. INTRODUCTION

Basic Information Retrieval (IR) concerns effective/efficient information retrieval from a repository for subsequent use. The main IR issue is locating a set of relevant information resources from a big repository which has information sought and hence satisfies information need which is usually expressed by a user query. Information resources can be objects (items) in a medium, text, image, audio, or a combination of all three. An IR process has two steps represented in Figure 1 [1].

Increase in amount and complexity of reachable information on the Web led to excessive demands for tools/techniques for semantic data handling. Current information retrieval practices rely on keyword-based search over full text data modelled with bag-of-words. But, models miss actual semantic information in text. To offset this, ontologies are proposed [2] for knowledge representation, the backbone of semantic web applications. Both information extraction and retrieval processes benefit from metadata, which provides semantics to plain text.

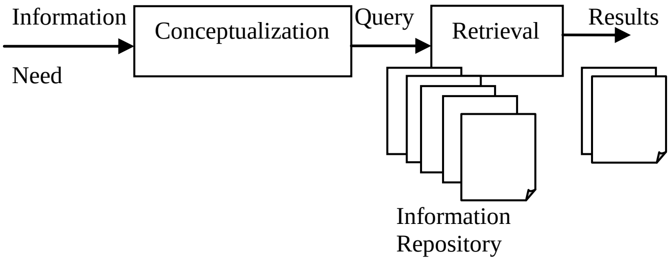

Figure 1: Basic Information Retrieval process

Conventional text classification is undertaken on “structured corpora with controlled authoring styles” [3], but web collections lack this. The web pages are semi-structured HTML documents to render the content visually for users. In general, the markups embedded in the document collections are not utilized for classification. Another feature of the web documents is that there exist within a hypertext, being interconnected to other documents. Though not web unique, this feature is central to web definition and does not present typical text classification issues.

Semantic technologies try to overcome IR limitation through explicit descriptions, internal structure and content and services overall structure [4]. All meanings and information conveyed by content in unstructured form (such as text or audio-visual content) cannot be fully translated to a clear/formal semantic representation, for pragmatic reasons. But, it is possible to describe parts of conveyed information, albeit to an incomplete extent, as metadata. Metadata is data about other data (e.g., the ISBN number and the author’s name are metadata about a book) [5]. For similar reasons it is useful to keep both information (data and metadata) parts in the system and relevant to have a link connecting both commonly called annotation. Different syntactic supports standards were proposed for metadata and annotation representations. Markup languages like HTML and XML are used with their features being effectively used for web pages classification.

Having got semantic knowledge and represented it via ontologies, the next step is to query semantic data, also called semantic search. Though many query languages are designed for this formal query languages cannot be used by end-users.

Formulating a query with such languages requires domain ontology knowledge and language syntax Hence, semantic web community works on simplifies query formulating for end-user. Current studies on semantic query interfaces are carried in four categories, namely, keyword-based, form based, view-based and natural language-based systems reviewed in [6].

Classification is important in information management and retrieval. Currently, web content is written in HTML (Hyper Text Markup Language), due to displaying content as syntax based HTML Web is for human use. Query ambiguity undermines HTML retrieval quality. For example, a query “bank” can be border of a water area or a financial establishment. Web pages have extra information like HTML tags, hyperlinks and anchor text with the regular textual content visible in a browser. These features located on the page are used for classification.

Semantic search is a motivation for semantic web from inception. An exploitation model for ontology-based knowledge bases to improve search over large document repositories is proposed. The 4 Universities Dataset contains WWW-pages used by the computer science departments of different universities used to evaluate the proposed method. Features extraction is through stemming, stop words, and locating IDF. The proposed feature extraction uses word importance based on html tag and ontology mapping to assign features extra weights..

2. RELATED WORKS

Du et al [7] proposed a novel ontology extractor, called OntoSpider, to extract ontology from HTML Web. This work’s contribution is design and implementation of a six-phase process including preparation, transformation, clustering, recognition, refinement, and revision to extract ontology from unstructured HTML pages. Extracted ontology provides structured/relevant information for e-commerce and knowledge management applications for effective comparison and analysis. Experiments validate system design and illustrate OntoSpider’s effectiveness.

Riboni [8] analysed web page classification peculiarities based on HTML structure and hyperlinks, trying to exploit them to represent web pages to improve categorization accuracy. Experiments on a corpus of 8000 documents of 10 Yahoo! categories, using Kernel Perceptron and Naive Bayes classifiers revealed the usefulness of dimensionality reduction and of a new, structure-oriented weighting technique. A new method to represent linked pages using local information ensuring hypertext categorization feasibility for real-time applications was introduced. It was seen that combining usual web pages representation using local words with a hypertextual one improved classification.

Qi et al [9] surveyed web page classification approaches from differing viewpoints, summarizing findings and contributions with discussion on benefits and utilization of web-specific features/methods. While appropriate use of textual and visual features residing on the page improved classification, neighboring pages features provide substantial information about the pages under classification. Feature selection and combining multiple techniques improve it further. The authors conclude that future web classification efforts will combine content and link information in some form.

Social bookmarking sites’ user-generated annotations provide promising metadata for web document management tasks like web page classification. User-generated annotations include diverse information like tags and comments. However, each annotation has a different nature/popularity level. Zubiaga et al [10] analyzed and evaluated these social annotations usefulness to classify web pages over a taxonomy like that of the Open Directory Project. Experiments compared them separately to content-based classification and also combined different data types to ensure augmented performance. They revealed encouraging results with social annotations for this purpose. It was also seen that combining metadata with web page content improves classifier's performance still further.

A promising recent approach to semantic web search is based on combining standard Web search with ontological background knowledge using standard Websearch engines as inference motor of Semantic Web search. Amato et al [11] proposed to enhance this further to Semantic Web search by using inductive reasoning techniques which ensures abilities to handle inconsistencies, noise, and incompleteness, likely to occur in distributed and heterogeneous environments like the web. A prototype was reported and implementations of the new approach with extensive experimental results were discussed.

3. METHODOLOGY

3.1 The 4 Universities Dataset

The 4 Universities Dataset includes WWW-pages from major universities computer science departments collected in January 1997 by CMU text learning group’s World Wide Knowledge Base (Web->Kb) project [12]. The dataset contains totally 8,282 pages which were manually classified into:

- Student
- Faculty
- Staff
- Department
- Course
- Project and
- Other.

The class other includes pages not considered the ``main page" and represents an instance of earlier six classes. Data set includes pages from Cornell, Texas, Washington, Wisconsin and 4,120 miscellaneous pages from other universities. Each class is assigned to a directory and each directory includes 5 subdirectories, one for each of the 4 universities and one for miscellaneous pages. The directories contain Web-pages.

3.2 Feature Extraction

Features are extracted from the documents using stemming, stop words, finding Inverse Document Frequency (IDF). The document and the query are represented as vectors in a high dimensional space corresponding to keywords in a vector space model. Similarity measures calculate similarity values between keywords and document. Ranking is based on similarity values. The first step is keywords identification for a document set. Next a list of unrelated/irrelevant words – called a stop list - avoid being indexed; words like the, a, of, for, with and so on are stop words.

In a document set d and a set of terms t, each document is modeled as a vector v in t dimensional space R^t, called a vector space model. Let frequency be denoted by $freq(d,t)$, as it expresses the number of occurrences of the term t in d document. The term-frequency matrix $TF(d,t)$ measures term t association regarding the given document d.

$TF(d,t)$ has nil value if the document does not contain the term, and a computed number otherwise. The number can be set as $TF(d,t) = 1$ when term t occurs in document d or uses relative term frequency which the frequency versus total occurrences of all document terms. Frequency is generally normalized by:

$$TF(d,t) = \begin{cases} 0 & freq(d,t) = 0 \\ 1 + \log\left(1 + \log\left(freq(d,t)\right)\right) & otherwise \end{cases}$$

Inverse Document Frequency (IDF), represents the scaling factor. If term t occurs frequently in many documents, its IDF value is less as the term has lower discriminative power. The $IDF(t)$ is defined as follows:

$$IDF(t) = \log \frac{1+|d|}{d_t}$$

d_t is the set of documents containing term t. Similar documents have similar relative term frequencies. Similarity is measured among a document set or between a document and query. Cosine measure locates documents similarity [15]; the cosine measure is got by:

$$sim(v_1, v_2) = \frac{v_1.v_2}{|v_1|\,|v_2|}$$

where v_1 and v_2 are two document vectors, $v_1.v_2$ defined as $\sum_{i=1}^{t} v_{1i} v_{2i}$ and $|v_1| = \sqrt{v_1.v_1}$.

An obvious feature in HTML documents but not in plain text documents are HTML tags and their respective attributes which create HTML documents to be viewed in browsers and other user agents. It was demonstrated that using information from tags boost classifier performance. Golub and Ardo [16] derived significance indicators for different tags textual content. Using tags is advantageous for structural information embedded in HTML files generally ignored by plain text approaches.

An HTML element is an individual component of a HTML document which in turn are made up of a tree of HTML elements and other nodes like text nodes. Every element has specified attributes. Elements have content, including other elements and text. HTML represents semantics or meaning. For example, title element represents document title. In HTML syntax, usually elements are written with a start and an end tag and with content in between. Tags have he element's name, surrounded by angle brackets. An end tag has a slash after opening angle bracket to differentiate it from a start tag.

Evaluation has two constituents: assessment of the system's performance in absolute terms, and regarding competing techniques, and assessment of knowledge adequacy in the system regarding its system performance impact. The main competing technique is to recall cases of query words occurrence. Recall and precision are measured for proposed semantic and keyword techniques allowing absolute and relative performance measures calculated using standard measures.

3.3 Support vector machine (SVM)

Support vector machine (SVM) is an algorithm using nonlinear mapping to convert original training data to a higher dimension [17]. Data from two classes are separated by a hyperplane with nonlinear mapping. SVM uses support vectors and margins to locate hyperplane. This methods disadvantage is that it is time consuming. Its advantages are its accuracy and being less prone to overfitting.

The margin is the distance between hyperplane and entity [18]. Output for a SVM with input vector $\vec{x}$ and $\vec{w}$ the normal vector to hyperplane, output u being given by equation: $u = \vec{w}.\vec{x} - b$

The separating hyperplane is the plane $u = 0$. The margin is given by equation

$$m = \frac{1}{\|w\|_2}$$

then maximizing margin is equal to solving the following optimization problem as shown in equation

$$\min_{w,b} \quad \frac{1}{2} \square\, w \,\square^{2}$$

$$\text{subject to } y_i = \left(\vec{w}.\vec{x} - b\right) \geq 1$$

b is a bias variable, and N is the training examples number. It follows that margin corresponds to quantity 1/□ *w*□ and margin maximization is achieved by minimizing □ *w*□^{2}.

Optimization problem is converted to quadratic programming where objective function ψ is dependent on Lagrange multipliers α_i as in equation,

$$\min_{\alpha} \psi(\vec{\alpha}) = \min_{\alpha} \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} y_i y_j \left(x_i.x_j\right) \alpha_i \alpha_j - \sum_{i=1}^{N} \alpha_i$$

Subject to constraints,

$$\alpha_i \geq 0 \quad \text{and}$$

$$\sum_{i=1}^{N} y_i \alpha_i = 0$$

The kernels of SVM are given as follows:

- Linear: $\left\langle x_i, x_j^{'} \right\rangle$
- Polynomial: $\left(\gamma < x, x' > + r\right)^d$. *d* is specified by keyword degree, *r* by coef ()
- RBF ($\exp\left(-\gamma \left|x - x'\right|^2\right), \gamma > 0$). γ is specified by gamma
- sigmoid $\tanh\left(< x_i, x_j > + r\right)$, where *r* is specified by coef ()

Classification techniques have advantages and disadvantages, whose importance is based on data being analysed and hence their relevance is relative. SVMs are a useful tool for insolvency analysis, in case of data non-regularity, for example when data is not regularly distributed/having an unknown distribution [19]. It evaluates information, to be transformed before entering classical classification techniques score. SVM techniques advantages are summarized as follows:

1. Through kernel introduction, SVMs gain flexibility in the choice of threshold form separating instances that need not be linear or have the same functional form for all data, as its function is non-parametric, operating locally.

2. As kernel contains a non-linear transformation, no assumptions are necessary about functional transformation form that ensures data is linearly separable. The transformation occurs implicitly on a robust theoretical basis with human judgment not being necessary.

3. SVMs provide a good out-of-sample generalization, when parameters C and r (in the case of a Gaussian kernel) are chosen correctly meaning that by selecting an appropriate generalization grade, SVMs are robust, even when having a biased training sample.

4. SVMs deliver unique solutions, as optimality problem is convex. This is advantageous compared to Neural Networks having multiple solutions linked to local minima and hence may not be robust over various samples.

4. RESULTS AND DISCUSSION

The proposed semantic based feature selection for web page classification using HTML tags is evaluated using the 4 Universities Dataset and compared with IDF feature extraction method. The goal of evaluation has two constituents: assessment of the system's performance in absolute terms, and with respect to competing techniques, and assessment of the adequacy of the knowledge represented in the system regarding its impact on system performance. The main competing technique is simply to recall cases where query words occur. Recall and precision can be measured for both proposed semantic and keyword techniques. This allows absolute and relative measures of performance to be calculated using standard measures.

SVM with various kernels (linear, polykernel, RBF, Sigmoid) classify keywords and semantic based features. Experimental results are detailed in the following tables and figures. Table 1 and Figure 2 detail classification accuracy and root mean squared error obtained for IDF and proposed feature extraction.

Table 1: Classification Accuracy and Root Mean Squared Error

Method Used	Classification Accuracy %	RMSE
SVM-linear-IDF	0.82	0.3
SVM-Polykernel-IDF	0.27	0.6
SVM-RBF-IDF	0.63	0.43
SVM-Sigmoid-IDF	0.7	0.39
SVM-linear-Proposed feature extraction	0.87	0.26
SVM-Polykernel-Proposed feature extraction	0.49	0.51
SVM-RBF-Proposed feature extraction	0.72	0.37
SVM-Sigmoid-Proposed feature extraction	0.73	0.38

The accuracy, precision, recall and f measure are computed as follows:

Accuracy (%) = (TN + TP) / (TN + FN + FP + TP)

$$precision = \frac{TP}{TP + FN}$$

$$recall = \frac{TP}{TP + FP}$$

$$f\ Measure = \frac{2 * recall * precision}{recall + precision}$$

where TN (True Negative) = Number of correct predictions that an instance is invalid, FP (False Positive) = Number of incorrect predictions that an instance is valid, FN (False Negative) = Number of incorrect predictions that an instance is invalid, TP (True Positive) = Number of correct predictions that an instance is valid

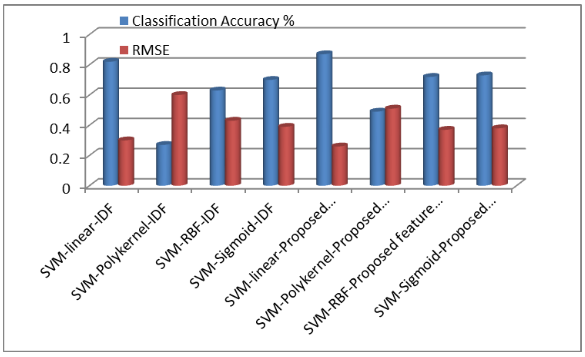

Figure 2: Classification Accuracy and Root Mean Squared Error

Figure 2 shows that the proposed feature extraction performs better than the IDF. The precision, recall and f measure for the different methods is shown in Table 2 and figure 3 and 4 shows the precision, recall and f measure respectively.

Table 2: Precision, Recall and F Measure

Method Used	Precision	Recall	F Measure
SVM-linear-IDF	0.826	0.82	0.822
SVM-Polykernel-IDF	0.564	0.27	0.14
SVM-RBF-IDF	0.713	0.63	0.632
SVM-Sigmoid-IDF	0.811	0.7	0.712
SVM-linear-Proposed feature extraction	0.887	0.87	0.869
SVM-Polykernel-Proposed-feature-extraction	0.789	0.49	0.422
SVM-RBF-Proposed feature extraction	0.779	0.72	0.716
SVM-Sigmoid-Proposed feature extraction	0.812	0.73	0.725

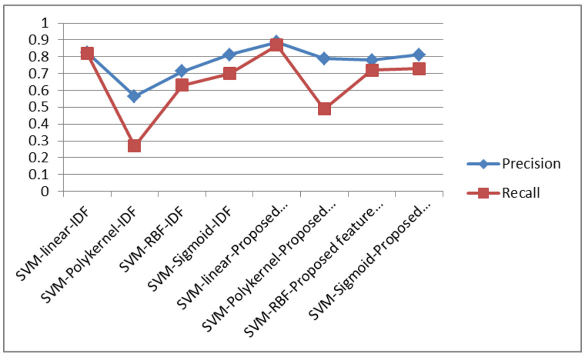

Figure 3: Precision and Recall

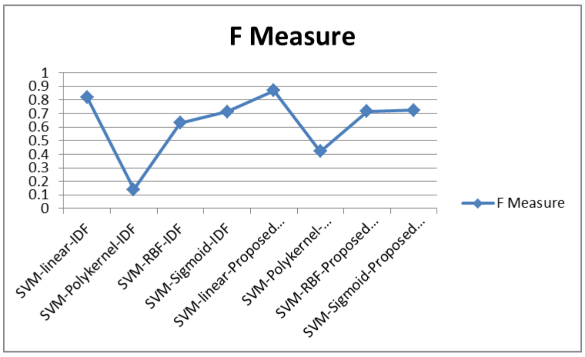

Figure 4: F Measure

F-Measure is resorted to despite Precision and Recall values being valid metrics in their own right with one being optimized at the other’s expense. It produces a high result when Precision and Recall are balanced which is significant. Figure 4 reveals that f measure for linear kernel is high.

5. CONCLUSION

As the web’s information volume increases, time required for locating information increases. So, when a user types keywords into a conventional search engine, search results volume is too large to locate useful information with the situation worsening when keyword search is unable to provide highly relevant results. Constructing ontology manually is time consuming and error prone and hence a method to extract semantics automatically from current Web resources such as Hyper Text Markup Language (HTML) documents is attractive. This paper proposes a model to exploit semantic-based feature selection to improve search and retrieval of web pages over huge document repositories. The features are classified using Support Vector Machine (SVM) using different kernels. The experimental results show improved precision and recall with the proposed method with respect to keyword-based search.

6. REFERENCES

[1] Stojanovic, N. (2005). Ontology-based information retrieval: methods and tools for cooperative query answering (Doctoral dissertation, PhD thesis, University of Karlsruhe.

[2] Thomas R. Gruber. Toward principles for the design of ontologies used for knowledge sharing. Int. J. Hum.-Comput. Stud., 43(5-6):907–928, 1995.

[3] Chekuri, C., M. Goldwasser, P. Raghavan, and E. Upfal (1997, April). Web search using automated classification. In Proceedings of the Sixth International World Wide Web Conference, Santa Clara, CA. Poster POS725.

[4] M. Fernández, V. López, M. Sabou, V. Uren, D. Vallet, E. Motta, and P. Castells. Semantic Search meets the Web. 2nd IEEE International Conference on Semantic Computing (ICSC 2008). Santa Clara, CA, USA, August 2008.

[5] V. López, M. Fernández, E. Motta, M. Sabou, V. Uren. Question Answering on the Real Semantic Web. Poster and demo at the 6th International Semantic Web Conference (ISWC 2007). Busan, Korea, November 2007.

[6] Victoria Uren, Yuangui Lei, Vanessa Lopez, Haiming Liu, Enrico Motta, and Marina Giordanino. The usability of semantic search tools: A review. Knowl. Eng. Rev., 22(4):361–377, 2007.

[7] Du, T. C., Li, F., & King, I. (2009). Managing knowledge on the Web–Extracting ontology from HTML Web. Decision Support Systems, 47(4), 319-331.

[8] Riboni, D. (2002). Feature selection for web page classification. In EURASIA-ICT 2002 Proceedings of the Workshop (pp. 473-477).

[9] Qi, X., & Davison, B. D. (2009). Web page classification: Features and algorithms. ACM Computing Surveys (CSUR), 41(2), 12.

[10] Zubiaga, A., Martínez, R., & Fresno, V. (2009, September). Getting the most out of social annotations for web page classification. In Proceedings of the 9th ACM symposium on Document engineering (pp. 74-83). ACM.

[11] d’Amato, C., Fanizzi, N., Fazzinga, B., Gottlob, G., & Lukasiewicz, T. (2010). Combining Semantic Web search with the power of inductive reasoning. Scalable Uncertainty Management, 137-150.

[12] Nigam, K., McCallum, A. K., Thrun, S., & Mitchell, T. (2000). Text classification from labeled and unlabeled documents using EM. Machine learning, 39(2), 103-134.

[13] Robertson, S. (2004). Understanding inverse document frequency: on theoretical arguments for IDF. Journal of Documentation, 60(5), 503-520.

[14] Papineni, K. (2001, June). Why inverse document frequency?. In Proceedings of the second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies (pp. 1-8). Association for Computational Linguistics.

[15] Steinbach, M., Karypis, G., & Kumar, V. (2000, August). A comparison of document clustering techniques. In KDD workshop on text mining (Vol. 400, pp. 525-526).

[16] Golub, K. and A. Ardo (2005, September). Importance of HTML structural elements and metadata in automated subject classification. In Proceedings of the 9th European Conference on Research and Advanced Technology for Digital Libraries (ECDL), Volume 3652 of LNCS, Berlin, pp. 368–378. Springer.

[17] Suykens, J. A., & Vandewalle, J. (1999). Least squares support vector machine classifiers. Neural processing letters, 9(3), 293-300.

[18] Gunn, S. R. (1998). Support vector machines for classification and regression. ISIS technical report, 14.

[19] Zhang, L., Lin, F., & Zhang, B. (2001, October). Support vector machine learning for image retrieval. In Image Processing, 2001. Proceedings. 2001 International Conference on (Vol. 2, pp. 721-724). IEEE.