Journal of Theoretical and Applied Information Technology
10th August 2013. Vol. 54 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

SEMANTIC BASED INFORMATION RETRIEVAL FOR WEB PAGES

[1]R.LENIN BABU, [2]Dr.S.VIJAYAN

[1]Assistant Professor ,Department of Information Technology , Hindusthan College of Engineering & Technology , Coimbatore , Tamilnadu , INDIA

[2]Principal, Surya Engineering College, Perundurai, Erode, Tamilnadu, INDIA

E-mail: [1]mdlenin@gmail.com , [2]svijayansurya@gmail.com

ABSTRACT

Content stored/shared on Web and document repositories has increased greatly leading to problems in locating required information from massive volumes. Progress in retrieving required information was achieved with search engine technology development that could collect, store and pre-process information globally, responding to users' needs instantly. Use of text classification techniques ensures web page classification. Presently, semantics are the basis for content description and query processing techniques required for Information Retrieval (IR). This paper presents an approach for information retrieval from web pages, based on the proposed extraction methods. AdaBoost algorithm is used to obtain and classify features and BF tree with the proposed feature extraction ensures high classification accuracy.

Keywords: *Information Retrieval (IR), Semantics, Latent Semantic Analysis (LSA), AdaBoost*

1. INTRODUCTION

Currently, the quantity of information accessible in document repositories has drastically increased, and most of the information is stored in digital format. Yahoo statistics estimated that the information content to be in the range of over 20 billion documents (in 2005) is available in the web which includes digital libraries and company intranets. But not all available content is useful. Users may also be unable to locate required information. This issue cropped up when the computer technology was still in its infancy.

Information Retrieval (IR) includes retrieving information for later use from a repository [1]. An issue is to locate information in a repository to satisfy user need raised through a user query. IR elements represent, extract and process user needs and content meanings. Retrieval process precision and increased user satisfaction are based on understanding semantics behind information items and user queries. IR systems use three responses including user interface, query processing operations and indexing resources [2]. User Interface: User interface flexibility enables users express information needs and possible constraints about information required (exact, similar or disjoint content and specific date, language, format content) [3].

Query processing operations: Based on query type, it is refined by various mechanisms, the most common being based on additional user input. Here relevance feedback approaches are most efficient, but as they reduce system visibility, other external resources like taxonomies and thesauri are used to classify/disambiguate/ expand query terms automatically.Resources for indexing: Document processing tools like thesauri and controlled vocabularies help select terms appropriate as index objects.IR systems' three main processes include item content features and descriptors extraction into a logic item representation (indexing); converting user information to an abstract representation (query processing) and matching both.(searching and ranking).

Indexing: All information is not significant equally to represent meaning. For example in written language some words have more meaning than others. Thus, information has to be pre-processed to select those which are to be used as index objects. Indices are data structures built to quicken search. An index should be built/maintained when item collection is large and semi-static. Inverted files are common text retrieval indexing structure, composed of two elements: vocabulary and term occurrences. Vocabulary included words in the text. Every word in the vocabulary lists all text positions where the word

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

appears is stored and such lists are called occurrences.

Query processing: User need and query are parsed before being compiled into an internal form [4]. Query terms are pre-processed by the same algorithms that select index objects in textual retrieval. Additional queries to be processed need the use of external resources like Thesauri or taxonomies. Searching: User queries and information items are matched resulting in potential information items returned while responding to user needs.

Semantic technologies which have IT technologies above abstraction layer ensure bridging and data interconnection, content and processes which have depth providing intelligent, capable, relevant, and responsive interaction than with information technologies alone. Semantic knowledge representation includes representational adequacy, fidelity, acquisition cost and computational cost trade off. Based on this four distinct semantic knowledge representations are seen in literature, from less semantically representative to the most complete as regards semantic knowledge representation and ontology [5].

Gabrilovich et al [6] proposed a novel method, Explicit Semantic Analysis (ESA), to represent text meaning in a high-dimensional concept space derived from Wikipedia. Machine learning techniques represent meaning of a text of Wikipedia-based concepts weighted vector. Trillo et al [7] suggested a semantic techniques set to group traditional search engine results into categories defined by input keywords differing meanings. Different from other proposals, this method considers web available ontology provided knowledge to dynamically define categories thereby making it independent of sources providing groupable results.

In this paper, an approach for information retrieval from web pages is presented. Features are extracted from the web pages based on proposed feature extraction method. The features obtained are then classified using AdaBoost algorithm.

2. METHODOLOGY

2.1 The 4 Universities Dataset

The 4 Universities Dataset contained WWW-pages from various university computer science departments collected by CMU text learning group's World Wide Knowledge Base (Web->Kb) project [8] in January 1997. This had a total of 8,282 pages classified manually into: Student, Faculty, Staff, Department, Course, Project and Others.

The class 'other' includes pages which are not the ``main page" and is representative of six classes. The data set has from Cornell, Texas, Washington, Wisconsin universities and another 4,120 miscellaneous pages from other universities. Each class is assigned a directory with latter each having 5 subdirectories, one for each university and another for miscellaneous pages. The directories contain Web-pages.

2.2 Latent Semantic Analysis (LSA)

Potential relations between keywords are usually ignored in conventional keyword-based IR approaches. So a text document key word's importance is assessed through an examination of the keyword occurrence in both document and collection without bothering about the existence of other related key words.

Latent Semantic Analysis (LSA) aka Latent Semantic Indexing (LSI), reaches beyond this restriction to analyse keywords documents co-occurrence in the collection.

Latent Semantic Analysis (LSA) [9] is a semantic space model based on word frequency. This is not based on discrete documents [10]. LSA builds a matrix where rows consist of words occurring in two documents and columns representing them. Cells (or features) indicate the many times a corresponding word (row) occurs in a document (column); so 0 means no occurrence in that document.

The most important LSA step is SVD application on the matrix as this reveals relationships between two words/ two documents [11, 12]. LSA has many applications for various works including information retrieval, automatic essay grading and synonym testing [13].

LSI is based on Singular Value Decomposition (SVD). SVD decomposes a term-by-document matrix, A, into three matrices: a term-by-dimension matrix, T, a singular-value matrix, S, and a document-by-dimension matrix, D. The number of dimensions is r, the rank of A. The original matrix can be obtained, through matrix multiplication of TSD^T. This decomposition is shown as

Journal of Theoretical and Applied Information Technology
10th August 2013. Vol. 54 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$A = TSD^T$$

T, S and D matrices are truncated to k dimensions in an LSI system being accomplished by removal of columns k+ 1 to r of T, columns and rows k+ 1 to r of S, and k + 1 to r of D^T. Dimensionality reduction reduces 'noise' in term-by-document matrix, causing a richer word relationship structure that shows hidden collection semantics [14, 15].

Queries are represented in reduced space by: qT_k , where T_k is the term-by-dimension matrix, after truncation to k dimensions. Queries scaled by the singular-values $S_k D_k^T$ are compared to reduced document vectors, providing each document a similarity score for a specific user query. The truncated term-by-document matrix is given as

$$A_k = T_k S_k D_k^T$$

and the result vector, w, is given as

$$w = qA_k$$

Similar to vector space retrieval, scores undergo descending order sorting with the system returning documents in rank order to users. Choosing an optimal dimensionality reduction parameter (k) has remained elusive for each collection. Optimal k is usually chosen by running a queries set with document sets for k's multiple values. The resultant k with good retrieval is chosen as each collection's optimal k whose values are generally in the 100-300 dimensions range [16].

2.3 Proposed Feature Extraction

Mathematically, k, approaches the rank of the term-by-document matrix, r as the truncation parameter; LSI approaches traditional vector space retrieval. Traditional vector space retrieval equals LSI when k = r. Term relationship information is resorted to when captured in the first few SVD vectors combined with traditional vector retrieval.

Document scores are attained through computing a weighted average of traditional LSI score computed using essential dimensions alone in the proposed model along with traditional vector space retrieval score. The resultant vector computation is revealed through:

$$w = (x)(qA_k) + (1-x)(qA)$$

where x is a weighting factor (0 <= x <= 1) and k is small.

Boosting" improves learning algorithms performance. Boosting greatly reduces "weak" learning algorithm's error when it regularly generates classifiers something akin to random guessing [17]. AdaBoost can lower any learning algorithm error and its pseudo code is given in Figure 1 below:

AlgorithmAdaBoost.M1

Input: sequence of *m* examples
$\langle (x_1, y_1), ..., (x_m, y_m) \rangle$ with labels
$y_i \in Y = \{1, ..., k\}$ weak learning algorithm
WeakLearn
integer T specifying number of iterations
Initialize $D_1(i) = 1/m$ for all *i*.
Do for $t = 1,2,\ldots T$

1. Call WeakLearn, providing it with the distribution D_t.
2. Get back a hypothesis $h_t : X \to Y$.
3. Calculate the error of
$h_t : \in_t = \sum_{i:h_t(x_i) \neq y_i} D_t(i)$ if $\in_t > 1/2$, then set
$T = t-1$ and abort loop.
4. Set $\beta_t = \in_t / (1 - \in_t)$.
5. Update distribution

$$D_t : D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} \beta_t & if \ \ h_t(x_i) = y_i \\ 1 & otherwise \end{cases}$$

where Z_t is a normalization constant (chosen so that D_{t+1} will be a distribution)

Output

the final hypothesis: $h_{fin}(x) = \arg\max_{y \in Y} \sum_{t:h_t(x)=y} \log \frac{1}{\beta_t}$

3. RESULTS AND DISCUSSION

The 4 Universities Dataset evaluates the proposed semantic based feature selection for web page classification and is compared with LSI feature extraction. Recall and precision are measured for both techniques as this ensures absolute and relative performance measures to be calculated using standard measures. Accuracy, precision, recall and f measure are computed as follows:

Journal of Theoretical and Applied Information Technology
10th August 2013. Vol. 54 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

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 is True Negative (Correct predictions that an instance is invalid)

FP is False Positive (Incorrect predictions that an instance is valid)

FN is False Negative (Incorrect predictions that an instance is invalid)

TP is True Positive (Correct predictions that an instance is valid)

ADA BOOST with decision stump, BF tree, and Random tree 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
Decision Stump - LSA	55%	0.3724
BF tree - LSA	87%	0.2177
Random Tree - LSA	73%	0.3674
Decision Stump - Proposed	61%	0.391
BF tree - Proposed	90%	0.2076
Random Tree - Proposed	83%	0.2915

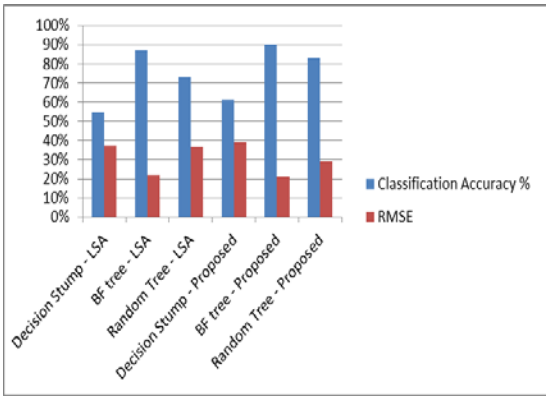

Figure 1: Classification Accuracy and Root Mean Squared Error

Figure 2 shows that the proposed feature extraction performs better than the LSA. 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
Decision Stump - LSA	*0.371*	*0.55*	*0.422*
BF tree - LSA	*0.869*	*0.87*	*0.867*
Random Tree - LSA	*0.73*	*0.73*	*0.725*
Decision Stump - Proposed	*0.447*	*0.61*	*0.493*
BF tree - Proposed	*0.901*	*0.9*	*0.9*
Random Tree - Proposed	*0.828*	*0.83*	*0.825*

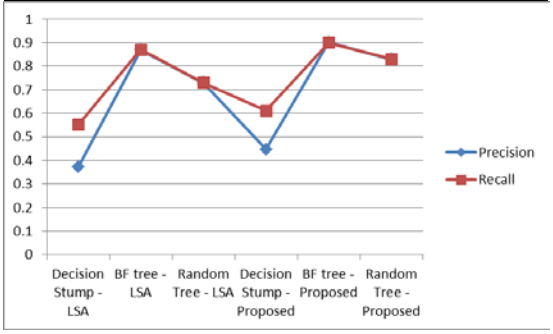

Figure 2: Precision and Recall

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

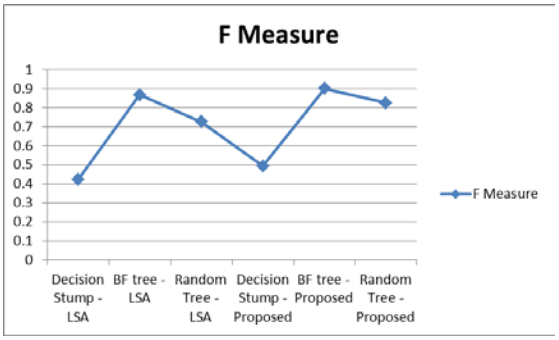

Figure 3: F Measure

F-Measure produces a high result when Precision and Recall are balanced which is significant. The proposed feature extraction does improve the classification accuracy and precision recall of the classifiers. BF tree with the proposed feature method achieves the best results.

4. CONCLUSION

Currently the most common content description and query processing techniques for Information Retrieval (IR) are based on semantics. In this paper, an approach for information retrieval from web pages is presented. Features are extracted from the web pages based on proposed feature extraction. The features obtained are then classified using AdaBoost algorithm. BF tree with the proposed feature extraction achieves the best classification accuracy of 90%.

REFERENCES:

[1] Gabrilovich, E., &Markovitch, S. (2007, January). Computing semantic relatedness using wikipedia-based explicit semantic analysis. In Proceedings of the 20th international joint conference on artificial intelligence (Vol. 6, p. 12).

[2] Kiryakov, A., Popov, B., Terziev, I., Manov, D., &Ognyanoff, D. (2011). Semantic annotation, indexing, and retrieval. Web Semantics: Science, Services and Agents on the World Wide Web, 2(1).

[3] Belkin, N. J., Marchetti, P. G., & Cool, C. (1993). BRAQUE: Design of an interface to support user interaction in information retrieval. Information processing & management, 29(3), 325-344.

[4] Ding, S., He, J., Yan, H., &Suel, T. (2009, April). Using graphics processors for high performance IR query processing. In Proceedings of the 18th international conference on World Wide Web (pp. 421-430). ACM.

[5] Mayfield, J., &Finin, T. (2003, August). Information retrieval on the Semantic Web: Integrating inference and retrieval. In Proceedings of the SIGIR Workshop on the Semantic Web.

[6] Gabrilovich, E., &Markovitch, S. (2007, January). Computing semantic relatedness using wikipedia-based explicit semantic analysis. In Proceedings of the 20th international joint conference on artificial intelligence (Vol. 6, p. 12).

[7] Trillo, R., Po, L., Ilarri, S., Bergamaschi, S., & Mena, E. (2011). Using semantic techniques to access web data. Information Systems, 36(2), 117-133.

[8] 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.

[9] Landauer T, Foltz P, Laham D. An introduction to latent semantic analysis. Discourse Processes: ABLEX PUBLISHING CO; 1998. p. 259-26.

[10] Rohde D, Gonnerman L, Plaut D. An improved method for deriving word meaning from lexical co-occurrence. Cognitive Science. 2004.

[11] McArthur R, Bruza P, Warren J, Kralik D, Projecting computational sense of self: A study of transition in a chronic illness online community. Proceedings of the 39th Hawii International Conference on System Sciences (HICSS-39); 2006.

[12] McArthur R, Bruza P, Discovery of implicit and explicit connections between people using email utterance. Proceedings of the Eighth European Conference of Computer-supported Cooperative Work; 2003: Kluwer Academic Publishers, Helsinki, pp. 21–40.

[13] Landauer T. Handbook of latent semantic analysis: Lawrence Erlbaum; 2007.

[14] 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

[15] S. T. Dumais. LSI meets TREC: A status report. In D. Harman, editor, The First Text REtrieval Conference (TREC-1), National Institute of Standards and Technology Special Publication 500-207, pages 137–152, 1992.

[16] T. A. Letsche and M. W. Berry. Large-scale information retrieval with latent semantic

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

indexing. Information Sciences, 100(1-4):105–137, 1997.

[17] Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. Computational Learning Theory: Second European Conference, EuroCOLT '95, pages 23–37, Springer-Verlag, 1995.