JOURNAL OF DATA SCIENCE AND INFORMATICS
Volume 1 Issue 1 Year 2023 Pages 1 - 8
e–ISSN XXXX–XXXX
Url: https://jdsi.etunas.com/index.php/JDSI/index

Analyzing Characteristics and Implementing Machine Learning Algorithms for Internet Search

Ananda Amalia[1]
[1]Teknik Informatika,Universitas Padjadjaran

Corespondence: ananda_amalia@unpad.ac.id

Abstract – With the rapid growth of online information, socializing on websites has become increasingly popular. However, finding relevant information within a limited time frame can be a daunting task, often resulting in irrelevant or incompatible search results. To address this challenge, researchers have explored different features used in information retrieval and their effectiveness in delivering relevant web pages to users. This paper provides an overview of various features utilized in information retrieval, including their classification and evaluation. The study also examines the relevance of web pages with users and compares different techniques, highlighting their pros and cons. Ultimately, this research aims to improve web search capabilities and enhance user satisfaction in finding relevant information.

keywords –; search engines; Information retrieval; User intention, Web Mining

I. INTRODUCTION

As more and more information becomes available on the world wide web, it becomes too difficult for search engines to maintain up-to-date and comprehensive search indexes. There is a problem of information overload, due to which searchers do not get relevant results. Most search engines only use keywords as queries. The main reason for irrelevant search results is keyword based search engines. Due to semantic ambiguity, the same word has many different meanings, the search results do not match the user's needs. Because different users may search for different information and even the same user may search for different information at different points in time. So, understanding the user's search objective and displaying the results according to the user's intention will definitely improve the quality of the search results. Web mining techniques can be used to solve information overload problems directly or indirectly. Web mining uses data mining techniques to automatically find and extract information from web documents and services [1].

Machine Learning is a field of computer science that gives computers the ability to learn without being explicitly programmed [2]. Machine learning techniques can be applied in the web mining process to improve search quality.

II. THEORETICAL BASIS

In this section, we review various features that we find useful in deciding whether a web page is relevant or not relevant to web users.

A. Web Page Features

- Textual content on web pages

Page content is the most obvious feature in determining page relevance. But because of the various voices on web pages, using directly the page content as a bag of words for all terms is not a

JOURNAL OF DATA SCIENCE AND INFORMATICS
Volume 1 Issue 1 Year 2023 Pages 1 - 8
e–ISSN XXXX–XXXX
Url: https://jdsi.etunas.com/index.php/JDSI/index

good choice. Therefore, there are various feature selection methods developed such as information acquisition, feedback information, document frequency, and x $^{2\ test}$.

- HTML tags

Web pages written in HTML apart from textual content contain HTML tags which provide additional information. HTML tags like title, body, anchor, heading, label, URL are widely used. Words and phrases that appear in titles or title tags are given higher weight. Another important feature is the web page URL, it is unique, meaningful and can help identify the web page domain [8].

B. Neighbor Features

Features that neighbors found useful include labels, partial content i.e. anchor text, surrounding anchor text, titles, headers, and full content. The label that is most often used is the neighboring feature and the anchor text is the feature that is rarely used [8].

C. Link Analysis

Web links have been widely used to provide important information about web pages. The widely used algorithms for this are the HITS (Hyper-Induced Topic Search) algorithm and the PageRank algorithm. In the PageRank algorithm [10], PageRank is calculated by weighting each inbound link to a page in proportion to the quality of the page containing the deep links. The quality of these referring pages is also determined by the PageRank algorithm. PageRank of page p is calculated recursively as followers:

$$PageRank(p) = (1-d) + d * \sum all \;\; q \;\; linking\ to\ p[\frac{PageRank(q)}{c(q)}] \qquad (1)$$

where d is the damping factor between 0 & 1, c(q) is the number of outgoing links in q.

The PageRank score of a web page is high if the page is referenced by many pages and the score is even higher if these reference pages are also of good quality. The PageRank score is calculated recursively, so it is very computationally expensive.

In (Hyperlink Induced Topic Search) [9] HITS algorithm, authority pages are high quality pages related to a particular topic or search query. Hub Pages are pages that do not need to be an authority on their own but that provide directions to other authority pages. A page where many other pages show good authority, and a page that points to many other pages should be a good hub. The authority value and hub score for each web page are calculated as follows:

$$Authority\ Score(p) = \sum all\ q\ linking\ to\ q\big(Hub\ Score(q)\big) \qquad (2)$$

$$Hub\ Score(p) = \sum all\ r\ linking\ to\ p(Authority\ Score(r)) \qquad (3)$$

A page has a high authority score if it is tagged by many hubs and a page has a high hub score if it points to many authorities. Another important web link is the anchor text, it is the text that describes the link. This is the clickable text of an exit link on a web page. The anchor text provides a nice description of the target page.

III. CASIFICATION

A. Web Page Content and Links as Features

Michael proposes a machine learning approach that combines web-based content and web-structure-based features. Two feedforward machine learning techniques / backpropagation neural networks and vector machine support are implemented and compared to traditional approaches -

keyword based approach and lexicon based approach. The proposed feature-web approach extracts all terms from a page and compares them with the domain lexicon. A total of 14 feature scores were determined [3]:

Page Content based features

Page titles p = Number of terms in page titles p found in the domain lexicon.

Frequency document inverse page p [TFIDF] = Number of TFIDF of terms on page p found in the domain lexicon. Neighbor-based feature page content The title and TFIDF score of the neighbor documents are determined as above. Sibling pages are the pages designated by one of the parent pages p.

Titles of all Entry pages q to p = Average (number of terms in page titles q found in the domain lexicon) for all incoming pages q of p.

TFIDF of all incoming Pages q to p = Average (number of TFIDF of terms on page q found in the domain lexicon) for all incoming pages q of p.

Titles of all outgoing pages q to p = Average (number of terms in the page titles r found in the domain lexicon) for all outgoing pages r of p.

Feedforward / backpropagation neural network (FF / BP NN) and Support Vector Machine (SVM) are compared against two traditional approaches namely lexicon based approach (LEXICON) and keyword based support vector machine approach (SVM-WORD).

Machine Learning Approach

In Feedforward / backpropagation neural network classifier - (FF / BP NN) a text classifying neural network is made with three layers namely input layer, hidden layer and output layer. The input layer consists of threshold units and 14 nodes corresponding to the 14 feature scores of each page. The output layer consists of a single node which determines the relevance of a page. The number of nodes in the hidden layer is set to 16 and the learning rate is at 0.10. After initializing the network, training and tuning is performed. For testing, each test document is passed to the network to predict whether it is relevant. Predictions are recorded and calculated to measure performance.

SVM Text Classifier- SVM Classifier also uses the same set of feature scores. A linear kernel function has been selected for the SVM classifier. Here, the training dataset consists of a vector of 14 features that are selected and presented to the SVM to learn feature weights. For testing, SVM tries to predict whether each document in the training set is relevant or not. The results are recorded and used to measure performance. For the evaluation medical field was chosen as the domain. A web page was tested and a medical lexicon created in the previous study was used [11].

Benchmarking Approach

LEXICON- In a lexicon-based approach, TFIDF scores are calculated for terms found in the medical lexicon. Arizona Noun Phraser (AZNP) [13] was used to extract noun phrases from each document and these phrases were compared to the lexicon. A Jaccard similarity score is calculated for each document in the training set and lexicon. The Jaccard score is widely used as a similarity score in information retrieval [12]. The threshold value that has the highest accuracy divides documents into relevant and irrelevant selected for testing.

The Keyword Vector Support Engine approach is based on preprocessing of the document which involves tokenization, stop-word removal, stemming using Porter's stemmer [14]. After the preprocessing stage, each document is represented as a keyword vector, which is then used for testing and training.

JOURNAL OF DATA SCIENCE AND INFORMATICS
Volume 1 Issue 1 Year 2023 Pages 1 - 8
e–ISSN XXXX–XXXX
Url: https://jdsi.etunas.com/index.php/JDSI/index

For evaluation, 50 cross-validation methodologies were used. Precision, recall, F-measure, accuracy and time were measured for all four approaches [3].

$$Precision = \frac{n\ document\ correctly\ classified\ as\ positive\ by\ system}{n\ all\ document\ classified\ as\ positive\ by\ system} \quad (3)$$

$$recall = \frac{n\ document\ correctly\ classified\ as\ posotive\ by\ system}{n\ positive\ document\ as\ testing\ set} \quad (4)$$

$$F - measure = \frac{2*precision*recall}{precision+recall} \quad (5)$$

Experimental results on accuracy, precision, recall and F-measure are given below.

Approach	Accuracy (%)	Precision (macro/ micro)(%)	Recall (macro/ micro)(%)	F-measure (macro/ micro) (%)	Time (min)
LEXICON	80.80	63.40/63.95	60.52/62.50	0.6005/0.6322	7.45
SVM-WORD	87.80	87.97/94.94	55.08/56.82	0.6646/0.7109	382.55
NN-WEB	89.40	81.38/82.38	76.19/76.14	0.7614/0.7913	103.45
SVM-WEB	87.30	85.35/86.24	61.99/61.74	0.7049/0.7196	37.60

The results of the experiment are discussed as follows: The keyword-based SVM approach takes the longest time because each document is represented as a large vector of keywords which creates a high dimension for the classifier. The lexicon-based approach takes the least amount of time, as it only needs to calculate the TFIDF and similarity scores for each document and define a threshold, which requires no complex processing. The NN classifier takes longer than the SVM-WEB based approach because the neural network has to be trained in several iterations.

The keyword based approach requires lots of training examples. Less training data is required for lexicons and web-based approaches.

The keyword-based SVM approach performed better than the lexicon-based approach. The proposed web features approach NN-WEB and SVM-WEB performed with comparable effectiveness. The two machine learning web feature approaches performed with higher effectiveness than the two benchmarking approaches namely, the -VM word-based approach (SVM-WORD) and the lexicon-based approach (LEXICON).

The direction of future work is to examine which of the 14 features are more important than the others in determining web page relevance. Another possibility of future work is to determine whether a combined approach of keywords and web features would perform better than using keywords or web features alone. Finally, the scope is to investigate how the proposed classification method can be used in other applications such as knowledge management and web content management.

B. Concepts as Features

This work presents a hybrid approach that uses document-based and concept-based profiles to re-personalize results obtained from existing search engines. User preferences have been extracted from past query and click-through data to predict document preferences for individuals. Browser extensions added to browsers that monitor user search, browsing, and click-through information. This browser extension will store queries and the results obtained for those requests as search engine results pages (SERPs) and click-through data. Click-through data conveys information about which result the

JOURNAL OF DATA SCIENCE AND INFORMATICS
Volume 1 Issue 1 Year 2023 Pages 1 - 8
e–ISSN XXXX–XXXX
Url: https://jdsi.etunas.com/index.php/JDSI/index

user clicked on, which in turn reflects the user's preferences. SERPs consist of a sequence of documents represented by triplets: title, content summary and source URL.

Each result returned by a web search engine is represented in the form of a web-snippet. The web-snippet will consist of a title, a content summary and a source URL. This Web., Snippet will be used further to extract the concept.

The user database consists of an aggregation of concepts. Concepts are a group of words that represent a user's preferences - and likes. Any group of words that appear frequently throughout the web-snippet will be considered a concept. - Word groups referred to as keywords are separated and - support for each is determined as follows [4]:

$$Support(ci) = \frac{sf(ci)}{n} * |Ci| \qquad (6)$$

sf(ci) is the snippet frequency of the ci keyword, n is the number of returned web-snippets and | ci | is the number of words in ci, If the support for ci is greater than the threshold value, then ci will be considered a concept. This way, the web snippet will be saved as a draft set instead of storing the title, URL, summary separately.

Next, user preferences are extracted from clicked as well as unclicked web scraps. All the concepts from the web-snippet are arranged in the form of a tree (ontology) using the different relationships that exist between these concepts. Two types of relationships are considered: similarity relationships and parent-child relationships.

After creation of this tree, the weight associated with the concept belonging to the web clips is increased and this increase in weight affects other concepts related to these concepts. After extracting user preferences from clicked web snippets, web snippets are re-summarized by assigning a score to them.

$$score(S1) = \sum fck * w(ck) \qquad (7)$$

fck is the number of times the ck appears in the snippet.

N is the total number of concepts included in that web snippet.

The strength of this study is that the concept information is obtained from user tracking data therefore no user manual intervention is required and it also provides dynamic learning that supports changing user needs with time. The future scope in this research is to introduce context and collaborative filtering. And more relationships can be associated with concepts.

C. URLs as Features

In this working machine learning technique is used and only URL features are considered for automatic classification of web pages into their categories [5]. The advantages of using URLs are that they are unique, meaningful, human readable and can help identify the category of a web page. No need to fetch and process web pages so it's faster too. The three machine learning techniques used are: Naïve Bayes classifier, Support Vector Machine and RBF. The dataset used is WEBKB and downloaded from the UCI repository. The Java program is used to parse the URL into tokens, to select features from the URL only and then each machine learning classifier learns the classification rules during the training phase.

Classifier performance results with the URL feature for test data.

JOURNAL OF DATA SCIENCE AND INFORMATICS
Volume 1 Issue 1 Year 2023 Pages 1 - 8
e–ISSN XXXX–XXXX
Url: https://jdsi.etunas.com/index.php/JDSI/index

Machine Learning Classifier	No.of Positive examples correctly classified	No. of Negative examples classified as positive	No.of examples classified as negative	No. of positive examples classified as negative	Overall Success rate
Naïve Bayes Classifier	50	93	27	80	0.308
SVM	48	73	47	82	0.38
RBF Network	130	120	NIL	NIL	0.52

Results show that: RBF has better success rate but classifying all instances into single category and if the number of instances is negative increases performance degradation. This is not useful for negative examples. The SVM classifier performs better than the other two classifiers. Naive Bayes classification shows an acceptable level of success. So, it is proved that URL features are sufficient for automatic classification of web pages using machine learning techniques.

The future work direction is to try to classify negatively classified pages by URL features using content based classification method to have better classification accuracy.

D. Nouns as Features

It proposes a method to extract user intent from web search logs and build an intention map that represents their information needs [6]. The user's search log contains the request and the associated clicked URL. This clicked URL was refined to remove duplicate and error page URLs. Intent features selected from this web document. Nouns are selected as features to extract intent. The nouns are retrieved from each web document through analysis of single morphemes and subordinate clauses. The TF/IDF values are used to assign weights to words that contain more information.

$$w_{ij} = tf_{ij} * \log(\frac{N}{df_j}) \quad (8)$$

w_{ij} means weight of jth feature in ith search log which contains user intent. tf_{ij} means frequency that feature j is presented in search log ith. dfj is the amount of data with the ith feature. A set of user intent is classified into categories using SOM's self-organizing map algorithm. Now user intent is extracted from each cluster using keywords representing the cluster. Intentions with high scores in each category were selected as representative intentions. Next, an intent map is constructed. The intention map adopts the schema model, a cognitive psychology knowledge representation method to represent user intentions. For the trial weblog a commercial commercial search engine was used. Behavioral experiments measured the expressiveness and effectiveness of the maps

The experimental results are given below:

JOURNAL OF DATA SCIENCE AND INFORMATICS
Volume 1 Issue 1 Year 2023 Pages 1 - 8
e–ISSN XXXX–XXXX
Url: https://jdsi.etunas.com/index.php/JDSI/index

Query	Expressiveness	Intention Map Effectiveness means(SD)	General Effectiveness means (SD)	Statistically differences
Bag	5.66	5.76(1.19)	4.14(1.60)	t(36)=5.025,p=0.000
Dream	5.62	5.86(1.08)	4.95(1.41)	t(36)=4.105,p=0.000
Fishing-tackle	5.04	5.23(1.38)	4.20(1.86)	t(29)=2.589,p=0.015
Constellation	5.24	5.45(1.04)	4.60(1.61)	t(29)=2.589,p=0.015
Mean	5.39	5.56	4.43	

The Likert scale is used to measure the expression of intention maps, namely how well intentions reveal questions that are appropriate to the relationship between search words, questions and intentions. User effectiveness was measured using a paired t-test. The main contribution of this research is that it can automatically extract user search intent based on search logs rather than keywords in queries, this enables basic studies which can provide query guidance for easy search and experimentation achieve reliable results on practical data from current search engines. This is from theoretical data. Future studies can focus on the structure of the intention map representation in the user service aspect.

E. Bag of Words as a Feature

This research proposes a technique based on machine learning involving three stages [7]:

- User Training - This is the process of taking input from users to convey their expectations from search queries.
- Infer knowledge from the training set.
- Use this knowledge to think about and refine new search results to determine their relevance classification.

User training

The user has to train the system by performing binary classification i.e. he has to classify each search result as "relevant" or "irrelevant" according to his requirements for the search query. Any search result flagged as relevant or irrelevant by a user is referred to as a training example.

Inferring Knowledge from Training Sets Training input from users stored as training sets is the raw data. This raw data needs to be processed and represented in a form that can be used to infer knowledge from it. This involves several steps:

- Gathering Information from the Training Set - A list of words that occur in all outcomes in the prepared training set. This list contains thousands of words.
- Data cleaning - The word list obtained in the above step contains parts of speech, and speech characters that need to be removed from the training data.
- Decision Matrix - A Decision Matrix is constructed consisting of a list of search results, their relevance classification and the appearance of all attributes in the search results.
- Decision Tree- The facts stated in the decision matrix need to be studied by the system to infer knowledge from it. An algorithm called ID3 (Iterative Dichotomizer 3) is used to study facts by building decision trees [15]. The input to this step is a decision matrix and the output will be a decision tree where the leaf nodes represent relevance classifications and the non-leaf nodes represent decision nodes, which are attributes.

JOURNAL OF DATA SCIENCE AND INFORMATICS
Volume 1 Issue 1 Year 2023 Pages 1 - 8
e–ISSN XXXX–XXXX
Url: https://jdsi.etunas.com/index.php/JDSI/index

Reasoning and Filtering of new search results

A top-down search in the decision tree is performed when new search results are considered. First, the attributes at the root of the decision tree are seen as a result. If found, the attribute on the right successor is shown, otherwise the attribute on the left successor is searched. Similarly, from top to bottom, the attributes in the tree iteratively look at the new search results until a leaf of the tree is reached, so that the search result relevance status is obtained.

Future work directions may use implicit feedback if the user is not interested in explicitly training the system or the system can be integrated with other systems or by combining implicit feedback with explicit feedback.

IV. CONCLUSION

In this paper, we have discussed various machine learning tools and techniques that filter results according to users' search intent. User search intent can be communicated to the system in two ways: implicit feedback and explicit feedback. Implicit feedback consists of user behavior and activity such as their page views, searches, and browsing activity. In research [4] - [5] [6] click-through data, URL and search logs are used, implicit feedback is used in their work. The advantage of using implicit feedback is that it does not interfere with user activity but the disadvantage is that it is only a system judgment of user intent, so it may not always accurately reflect user intent in every case. Between [4], [5] and [6], [5] is faster because it only uses the URL feature and does not need to fetch the web page and pre-process it. In explicit feedback, the user explicitly gives his preference so it is more accurate than implicit feedback. In [3] and [7] the user trains the system, so that explicit feedback from their work is used. The loss of explicit feedback is that the user has to train the system, so there is some training overhead but the system makes up for it by saving greater time and effort due to the high accuracy of the filtered results. The review shows that the feedforward/backpropagation neural network (FF/BP/NN) and Support vector machine (SVM-WEB) classifier performs better than the keyword-based approach, lexicon-based approach, RBF and Naïve Bayes classifier.

REFERENCE

[1] O. Etzioni," The world wide web: Quagmire or gold mine", Communications of the ACM, 39(11):65-68,1996.

[2] https://en.wikipedia.org/wiki/Machine_learning

[3] Michael Chau, Hsinchun Chen,"A machine learning approach to web page filtering using content and structure analysis",Decision Support Systems 44(2008) 482-494,2007

[4] Tarannum Bibi, Pratiksha Dixit, Rutuja Ghule, Rohini Jadhav, "Web search personalization using machine learning techniques", IEEE, 2014

[5] M.Indra Devi, Dr.R.Rajaram and K.Selvakuberan,"Machine learning techniques for automated web page classification using URL features",International conference on computational intelligence and multimedia applications 2007,IEEE.

[6] Kinam Park, Taemin Lee; Soonyoung Jung; Sangyep Nam; Heuiseok Lim, "Extracting Search Intentions from Web Search Logs", Information Technology Convergence and Services (ITCS),Pages: 1 – 6,IEEE,2010.

[7] Varun Gupta, Neeraj Garg and Tarun Gupta, "Search Bot: Search Intention Based Filtering Using Decision Tree Based Techniques", Third International Conference on Intelligent Systems Modeling and Simulation, IEEE, 2012.

[8] Xiaoguang Qi and Brian D. Davison,"Web page classification:features and algorithms", ACM computing surveys, vol.41,No. 2,Article 12(February 2009). [9] J. Kleinberg, "Authoritative sources in a hyperlinked environment", Proceedings of the ACM-SIAM Symposium on Discrete Algorithms, 1998. [10] S.Brin, L.Page, "The anatomy of a large-scale hypertextual web search engine", Proceedings of the 7th International World Wide Web Conference, Brisbane, Australia, Apr 1998. [11] M.Chau, H .Chen, "Comparison of three vertical search spiders", IEEE Computer 36(5) (2003a) 56-62.

[12] CJ van Rijsbergen, "Information Retrieval", Second edition, Butterworths, London, 1979.

[13] K. Tolle, H. Chen, "Comparing noun phrasing techniques for use with medical digital library tools", Journal of the American Society for Information Science 51(4) (2000) 352-370.

[14] MF Porter, "An algorithm for suffix stripping", Program 14 (3) (1980) 130-137.

[15] JR Quinlan,"Induction of Decision Tree",Machine Learning,Vol. 1,pp 81-106,1986.