International Journal of Advance Research in Science and Engineering
Volume No.07, Issue No.02, February 2018
www.ijarse.com
IJARSE
ISSN: 2319-8354

DOMAIN IDENTIFICATION OF A WEBPAGE USING STATISTICAL AND LANGUAGE MODEL BASED KEYWORD EXTRACTION AND ONTOLOGY MAPPING

Swati Goel

Master of Technology in Computer Science, ABSS Institute of Technology Meerut,
AKTU (APJ Abdul Kalam Technical University, Formerly UPTU), Uttar Pradesh, INDIA

ABSTRACT

Nowadays sources of information are increasing as the technology is advancing, for ex: books, Journals, Internet etc. Among them internet is growing rapidly as source of information. Because large information is available on internet, finding useful and desired information from a huge amount of data may be a challenging task for users. As the technology rising information in various topics also increasing that's why information finding must be accurate and fast enough. Readers can easily determine if a webpage is relevant to them if the significant phrases or say keywords are provided with the document. In the past few years, Ontologies are increasing in the use for the knowledge representation. Also they are being used in several application contexts. One of the challenging applicationis the Web. Managing and searching thought huge amount of data on internet needs more efficient, fast and effective methods and technique for mining and representation. In this thesis an innovative and hybrid approach for automatically identifying domain of an internet page is proposed by using keywords and ontologies. The proposed keyword extraction method based on statistical and language model, works on a single document. It takes the full advantage of all the features of the document to extract the keywords.

Keyword: Domain identification, Keyword extraction, Ontology mapping

I INTRODUCTION

Everyday lots of books, papers and webpages are published containing different type of data. As the technology is advancing sources of data is also increasing on webpages, books etc. With the use of internet in day to day life it may be difficult to find the correct page between huge amounts of pages. There is a need of accurate and fast information extraction or summarization methods which provide the actual contents of a given webpage. This will

save efforts, cost and time by giving accurate and wanted results of a search query instead of giving bulky, unrequired, junk data or the results which are not required for a certain search result.

The most important unit in this requirement is 'Keyword.' It is the smallest unit which expresses meaning of the entire document or a webpage. If they are chosen correctly then they provide a compact representation of the entire document. For example, if the keywords are printed on the first page or on the heading of a Journal then the goal is summarization or to be precise for the related information. For instance, in digital libraries, authors assign keyphrases (short phrase) to their documents when they are instructed to do so [1].

Another example is in the case of the search engine. When a search engine has a field for input some words, the goal is to make the search more precise. They enable the reader to quickly find a relevant and required article fulfilling the need of reader.
Most existing approaches focus on the manual assignment of keywords by professional curators who may use a fixed taxonomy, or rely on the authors' judgment to provide a representative list. Manual keyword extraction is an extremely difficult and time consuming task. Research has therefore focused on methods to automatically extract keywords from documents as an aid either to suggest keywords for a professional indexer or to generate summary features for documents that would otherwise be inaccessible.

The goal of automatic extraction is to apply the power and speed of computation to the problems of access and discoverability, adding value to information organization and retrieval without the significant costs and drawbacks associated with human indexers. Automatic keyword Extraction should be done systematically and with either minimal or no human intervention, depending on the model.

1.1 Existing Approaches to Keyword Extraction

Many algorithms and systems for automatic keyword extraction have been proposed. Existing methods about Automatic Keyword Extraction can be divided into below categories,

1.1.1 The Supervised Approach

The supervised approach treats the problem as a classification task. In this approach, a model is constructed by using training documents, already labeled with key phrases assigned (by humans) to them. This model is applied in order to select key phrases from previously unseen documents. Peter Turney [2]is the first one who formulated key phrase extraction as a supervised learning problem. According to Turney, all phrases in a document are potential key phrases, but only phrases that match with human assigned ones are considered "correct" key phrases. Turney uses a set of parametric heuristic rules and a genetic algorithm for the extraction procedure.

Another notable keyphrases extraction system is KEA (Keyphrase Extraction Algorithm)[3], it builds a classifier based on the Bayes' theorem using training documents, and it uses the classifier in order to extract keyphrases from new documents. In the training and extraction phases, KEA analyzes the input document depending on orthographic boundaries (such as punctuation marks, newlines, etc.) and exploits two features : tf * idf (term frequency *inverse document frequency) and first occurrence of the term. All the above systems need a training data in small or large extent in order to construct an extraction system. However, acquiring training data with known keyphrases is not always feasible and human assignment is time-consuming. Furthermore, a model, trained on a specific domain, does not always yield to good classification results in other domains.

Hulth [4] introduces linguistic knowledge, i.e. part-of-speech (pos) tags, in determining the candidate sets. She uses 56 potential pos-patterns in identifying candidate phrases in the text. Her experimentationhas shown that, using a pos tag as a feature in candidate selection, a significant improvement of the keyphrase extraction results can be achieved.

1.1.2 The Unsupervised Approaches

The unsupervised approach eliminates the need of training data. It selects a general set of candidate phrases from the given document, and it uses some ranking strategy to select the most important candidates as key phrases for the document [5].

1.1.3 Statistical Techniques

These methods are simple and do not need the training data. The statistical information of the words can be used to identify the keywords in the document. They base themselves on term frequency to determine the term importance. Common ways to determine term importance include TF-IDF, entropy, mutual information, word frequency, word co-occurrence, and statistics. Sometimes, term importance is reinforced if the terms belong to title words, cue-phrases, and/or capitalized words. Cohen uses N-Gram statistical information to automatic index the document [6]. The benefits of purely statistical methods are their ease of use and the fact that they do generally produce good results.

1.1.4 Linguistics-based Approaches

These approaches use the linguistic features of the words mainly sentences and documents. The linguistic approach includes the lexical analysis, syntactic analysis discourse analysis and so on[7], [8].

1.1.5 Machine Learning Approaches

Machine Learning approach employs the extracted keywords from training documents to learn a model and applies the model to find keywords from new documents. This approach includes Naïve Bayes, Support Vector Machine,

etc. The machine learning mechanism works as follows. First a set of training documents is provided to the system, each of which has a range of human-chosen keywords as well. Then the gained knowledge is applied to find keywords from new documents.

1.1.6 Mixed Approaches

Other approaches about keyword extraction mainly combine the methods mentioned above or use some heuristic knowledge in the task of keyword extraction, such as the position, length, layout feature of the words, html tags around of the words, etc.

1.2 Ontologies

Ontologies are at the core of the well-known layer structure for the Semantic Web (http://www.w3.org/DesignIssues/Semantic.html), providing the opportunity of representing arbitrary worlds. Ontology is an explicit formalization of a shared understanding of a conceptualization. This high-level definition is realized differently by different research communities, but most definitions include a set of concepts, a hierarchy on them, and (n-ary) relations between concepts. The relations may be linked to one another by a relation hierarchy. Most of them also include axioms in some specific logic [9]. Depending on whether one needs axioms or not, one can use OWL or RDF Schema to formalize ontologies that conform to the definition.

Ontologies are formed of concepts. Web pages usually describe concrete instances of these concepts in human-readable form. Metadata are the intermediates between these two representations; their objects (identified by URIs) can be seen as instances of the ontology concepts [10].

Besides the formal languages for the Semantic Web, ontologies for general use are developed [11]. At present, there are mainly in practice two types of ontologies. The first type uses a small number of relations between concepts, usually the subclass relation and sometimes the part-of relation. Popular and commonly used are ontologies of Web documents, such as DMoz or Yahoo!, where the documents are hierarchically organized based on content (for example: \Computers" { \DataFormats" { \Markup Languages"). For each content topic (such as \XML" below\Markup Languages"), there is an ontology node, and this is associated with usually several hundreds of) Web pages identified by their URLs.

The other kinds of ontologies are rich with relations but have a rather limited description of concepts, usually consisting of a short definition. A well-known example of a general, manually constructed ontology is the semantic network Word Net (http://wordnet.princeton.edu) with 26 different relations (e.g., hypernym, synonym). For instance, the concept \bird" is connected to \animal" via \is a kind of" and to \wing" via \has part".

II PROBLEM STATEMENT AND PROPOSED WORK

2.1 Problem Statement

For a document, identifying an automatic methodology that generates a limited set of metadata (also called keywords, tags or keyphrases), which properly describes the given content, represents an open issue and a stimulating challenge. This request is strongly perceived on the Web, where the amount of user-generated content and its steady growth rate exacerbate the information overload, the disorientation, the cognitive overhead and the difficulty both to retrieve interesting documents and to classify them for future uses. of the collection in which the document ends. Although the manual process usually reaches high quality levels of classification for traditional document collections, it does not scale to the humongous size of the Web, both in terms of costs, time, and expertise of the human personnel required, and as such it cannot be proficiently put into existence for the whole Web. ; Research has therefore focused on methods to automatically extract keywords from documents as an aid either to suggest keywords for a professional indexer or to generate summary features for documents that would otherwise be inaccessible. Keyword Extraction Techniques provides a short list of keywords or keyphrases (typically five to fifteen phrases) that reflects the content of a single document, providing a brief summary of document's content. Many existing algorithms and systems aimed to perform automatic keywords extraction have been proposed. However, currently existing solutions for automatic keyword extraction require either training examples or domain specific knowledge.

A disadvantage of supervised approaches is that they require a lot of training data and yet show bias towards the domain on which they are trained, undermining their ability to generalize well to new domains. However, acquiring training data with known keyphrases is not always feasible and human assignment is time-consuming.

2.2 Proposed Framework

To cater to the above mentioned short comings an unsupervised algorithm is being proposed to extract keywords from a web page document and suggest the domain of a Web page. The proposed methodology combines keyword extraction, and ontology mining, and assists in domain specific Indexing of a Web page. The advantages expected of the proposed methodology are:

- Provide keywords which summarize [12] the content of a Web Page.
- Use a controlled, ontology-based vocabulary, not necessarily present in the original Web resource, and classify web page.
- Reduce the manual effort required to tag a web page.
- Provides efficient indexing technique that results in fast retrieval of documents based on user interest.

The proposed algorithm extracts significant keywords by combining statistical features and linguistic knowledge for a given Web document and then maps a webpage to a domain using built-in ontologies in order to classify and cluster webpages.

The Proposed system consists of two phases. The first phase is concerned on analyzing HTML web Pages. In this a document is fetched from a document collection, the fetched page is then analysed by keyword extractor which extract significant keywords from the HTML pages. In second phase the extracted keywords are matched with ontological constructs to determine the domain of the corresponding webpage. The webpage along with the significant keywords is then stored in domain specific webpage repository.

2.2.1 Keyword Extractor module

This is used to extract keywords from web pages. It uses a combination of statistical and linguistic features to extract the keywords. The general workflow of the keyword Extraction module is describe below,

Step-1: Cleaning and Initialization

The plain text form is then processed to delimit sentences. Separating sentences by inserting a sentence boundary is the main aim of this step. The following heuristics are applied in setting the sentence boundaries.

– Special symbols such as '.', '@', ' ', '&', '/', '-', '' are replaced with the sentence delimiter wherever they appear in the input document, but with the following exemptions:

_ The symbols '.', '@', ' ', '&', '/', '-' are allowed if they are surrounded by letters or digits (e.g., e-commerce, hiperlan/2).

_ The symbol '' is allowed if it is preceded by a letter or digit (e.g., pearson's correlation).

– Other punctuation marks (e.g., '?', '!') are simply replaced by sentence delimiter,

– Apostrophes are removed and the entire input text is converted into lowercase.

Step-2: Stemming and Stop-word removing

Stemming (i.e. removing word suffixes such as `ing', `ion', `s') consists of converting each word to its stem, i.e. a natural form with respect to tag-of-speech and verbal/plural inflections. Stop words (i.e. insignificant words like `can', `in', `this', `from', `then', `or', `the', `by') are words that occur very frequently in a document.

From the processed text, we remove all phrases that start and/or end with a stop-word and phrases containing the sentence delimiter. Partial stemming (i.e., unifying the plural forms and singular forms which mean essentially the same thing) is performed.

Step-3: Feature Calculation

The feature calculation step characterizes each candidate keyword by statistical and linguistic properties.

Statistical Features

The various Statistical features considered are specified in Table.1

TABLE 1: Features used in keyword extraction

No.	Features	Explanations	Normalization Method
1	TF	Term Frequency	Log(TF)
2	T	Whether the word has appeared in the Title	{0,1}
3	M	Whether the word has appeared in the Meta Tag	{0,1}
4	U	Whether the word has appeared in the URL	
5	A	Whether the word has appeared in the Anchor Text	{0,1}
6	H	Whether the word is Highlighted	{0,1}

1. Term Frequency

We use term frequency (TF) of the candidate as a feature. More the number of times a word/phrase occurs, better are the chances of it being a keyword.

2. Title

The title attribute is a flag that indicates if a term appears in the title of the document. A term that occurs in the title of the document is often more valuable than a term that does not. Titles may not provide enough information on their own, but they may contain some important words. Therefore if word appears in titles, the word carries more weight than other words.

3. Meta-Tag

Meta-Tag contains meta information of a webpage. The terms that appear in meta-tag are important.

4. URL

URL strings are interesting content available on the web.URLs follow a well-defined structure with hostname, category, query and other components. The content in URLs is usually precise and contains text that is highly condensedbut relevant to the topic of the web page.

5. Anchor Text

As the descriptions from people other than authors, anchor text phrases describe the target Web pages more precisely and objectively. Therefore, they could be effectively used in the Extraction of important keywords. Therefore, words appearing in anchor text phrases are given more weightage.

6. Highlighted Words

Highlighting is the practice of emphasizing key phrases and key passages (e.g., sentences or paragraphs) by underlining the key text, using a special font, or marking the key text with a special color. It is known that the words may encounter in various forms of writing in the documents. These forms of writing provide additional, but substantial information about the importance of words.

After the application of statistical features a score is assigned to each keyword for selecting the most appropriate candidate keywords of the document. The score of each keyword is calculated as linear combination of the above six features. We call the resulting score value as Key-Score of the keyword.

Key-Score=TF+T+M+U+A+H (2.1)

On the basis of the key-score top n keywords with maximum Key-score are selected as candidate keyword of the document. Now an information score is applied to these candidate keywords using the linguistic features described below.

Linguistic Features

The feature calculation step characterizes each candidate keyword by some linguistic properties. Following linguistic features for each candidate keyword are computed:

1. POS value

We assign a POS tag (noun, adjective, verb etc.) to each term, by using Stanford log-linear part-of-speech tagger. A candidate Keyword is assigned a POS value 1 if it qualifies to be noun keyword.

2. Keyword depth

This feature reflects the belief that important words appear in the initial part of the document especially in news articles and scientific publications(e.g. abstract,introduction).We compute the position in the document where the word first appears. The candidate depth value is calculate as:

$$Depth(T,D)=1-\frac{first\ index(T)}{Size(D)} \quad (2.2)$$

Where, first index(T) is the number of words preceding the candidate term first appearance; size(D) is the total number of words in web document D.

The result is a number between 0 and 1. Highest values represent the presence of a candidate term at the very beginning of the document. For instance, if a term appears at 16th position, while the whole document contains 700 words, the term depth value is 0.97, indicating the first appearance at the beginning of the document.

3. Last Occurrence

We also give importance to candidates that appear at the end of the document(i.e. in the conclusion and discussion parts).The last occurrence value of candidate is calculated as the number of words preceding the last occurrence of the word normalized with the total number of words in the document.

$$Last\ occurrence\ (T,D)=1-\frac{last\ index(T)}{Size(D)} \quad (2.3)$$

Where, last index(T) is the number of words preceding the candidate term last appearance; size(D) is the total number of words in web document D.

For instance, if a word appears for the last time at 500th position in a document that contains 700 words then the word last occurrence value is 0.71.

4. Lifespan

The span value of a candidate term depends on the portion of the text that is covered by the candidate term. The covered portion of the text is the distance between the first occurrence position and last occurrence position of the term in the document. The lifespan value is computed by calculating the difference between the candidate last occurrence and the candidate first occurrence. The lifespan value for phrase P in a document D is

$$\text{lifespan}(T,D) = \frac{\text{last index}(T) - \text{first index}(T)}{\text{Size}(D)} \quad (2.4)$$

where, last index(T) is the number of words preceding the candidate term last appearance; first index(P) is the number of words preceding the candidate term first appearance; size(D) is the total number of words in D.

The result is a number between 0 and 1. Highest values mean that the particular candidate term is introduced at the beginning of the document and carried until the end of the document. Terms that appear only once throughout the documenthave the lifespan value 0.

Step-4 Scoring

In this step a score is assigned to each candidate keyword. The Score is calculated as a linear combination of the four linguistic features and the resulting score value is referred as a Information score.

Information score=POS value + keyword depth+ Last occurrence+ Lifespan (2.5)

Top n candidate keywords are then selected as the significant keywords of the webpage.

2.2.2 Ontology Mapping

For each keyword extracted by the proposed Algorithm for a given document the ontology mapping module looks for a corresponding match in the built-in ontologies retrieving immediate Super class by following parent child relationship. The retrieved Super class is marked as the domain node for the given document. The document along with the domain node information is stored in the domain specific web-page repository.

III EXPERIMENT AND CONCLUSION

3.1 Experiment and result

The proposed system has been tested and the results are explained. The webpage having URL https://aktu.ac.in/history.htmlis used as the test set. We take the sample text from this webpage.

After the process of Cleaning, Stemming and Stop-word removing of sample text we got the tokens that qualify for feature selection.After this step we calculated feature (term frequency, title, meta-tag, URL, anchor text) by using “Screaming Frog” application.Similarly, we calculated all the six statistical features for each tokens. Tokens with key-score greater than 5 are selected as candidate keywords and linguistic features for these candidate words are calculated further.

We have shown all the steps in details in the complete research work. We have also implemented algorithm to find out the frequency count of words from a given text file.Candidate keywords with Information score greater than two are selected as Final Significant keyword and shown in left of below Table 2, where they are being compared to the results with thekeywords extracted by the Keyword Extraction Tool by "https://wordcountools.com".

TABLE 2: Result and Comparison

Keywords extracted by proposed algorithm	**Keywords extracted by"WordCountTool"**
Aktu	Aktu
Technical	Technical
University	university
Programmes	Programmes
Research	
Colleges	Colleges
Institutions	Institutions
	Technology
	Affiliated
	State

The statistical measures Precision, Recall and F-measure are used to evaluate the accuracy of the proposed algorithm.

Precision (P) = Relevant Keywords '
Total Number of Keywords (3.1)

Recall (R) = Number of Correct Results '
Number of Results that should have been return (3.2)

F-measure (F) = 2PR/ P+R (3.3)

Out of seven keywords Extracted six are true positive that is the exactly match six of the keywords assigned by online keyword Extraction tool. There are therefore one false positives in the set of extracted keywords resulting in the precision of 85.7%.Comparing the six true positives within the set of extracted keywords to the total of nine keywords assigned by online keyword extraction tool results in a recall of 55.5%.Equally weighting precision and recall generates an F-measure of 67%.

3.2 Conclusion

In this paper an innovative and hybrid approach for automatically identifying domain of a webpage using keywords and ontologies have been proposed. The statistical and language model based keyword extraction works on a single document without any previous parameter tuning and takes full advantage of all the features of the document to extract the keywords. In contrast to methods that depend on natural language processing techniques to achieve keyword extraction, the proposed algorithm takes a simple set of input parameters and automatically extract keywords in a single pass.

REFERENCES

1. T. R. Gruber. 1993, Towards Principles for the Design of Ontologies Used for Knowledge Sharing.In N. Guarino and R. Poli, editors, Formal Ontology in Conceptual Analysis and Knowledge Representation, Deventer, Netherlands, 1993. Kluwer.
2. Witten, I.H., Paynter, G.W., Frank, E., Gutwin, C., Nevill-Manning, C.G.: 1999, Kea: practical automatic keyphrase extraction. In: Proceedings of the fourth ACM conference on Digital libraries. pp. 254{255. ACM, New York, NY, USA (1999).
3. Andrade M and Valencia A 1998 Automatic extraction of keywords from scientific text: application to the knowledge domain of protein families. *Bioinformatics* 14(7),600–607.
4. Hulth. 2003, Improved automatic keyword extraction given more linguistic knowledge. In Proceedings of the 2003 Conference on Emprical Methods in Natural Language Processing, Sapporo, Japan, 2003
5. Matsuo Y and Ishizuka M 2004 Keyword extraction from a single document using word co-occurrence statistical information. *International Journal on Artificial IntelligenceTools* 13(1), 157–169.
6. J. D. Cohen. 1995, Language and domain-independent automatic indexing terms for abstracting. Journal of the American Society for Information Science, 1995.
7. A. Hulth. 2003, Improved automatic keyword extraction given more linguistic knowledge.In Proceedings of the 2003 Conference on Emprical Methods in Natural Language Processing, Sapporo, Japan, 2003.
8. Hulth, A. 2003, Improved automatic keyword extraction given more linguistic knowledge. In: Proceedings of the 2003 conference on Empirical methods in natural language processing. pp. 216{223. Association for Computational Linguistics, Morristown, NJ, USA (2003).
9. E. Bozsak, M. Ehrig, S. Handschuh, A. Hotho, A. Maedche, B. Motik, D. Oberle, C. Schmitz, S. Staab, L. Stojanovic, N. Stojanovic, R. Studer, G. Stumme, Y. Sure, J. Tane, R. Volz, and V. Zacharias. Kaon 2002, - towards a large scale semantic web. In K. Bauknecht, A. Min Tjoa, and G. Quirchmayr, editors, E-Commerce and Web Technologies, Third International Conference, EC-Web 2002, Proceedings, volume 2455 of LNCS, pages 304{313, Berlin, 2002. Springer.
10. A. Maedche and S. Staab. 2001, Ontology learning for the semantic web. IEEE Intelligent Systems, 16(2):72{79, 2001.

11. A. Maedche. Ontology Learning for the Semantic Web. Kluwer, 2002.)and agents [95](A.B. Williams and C. Tsatsoulis. 2000, An instance-based approach for identifying candidate ontology relations within a multi-agent system. In Proceedings of the First Workshop on Ontology Learning OL'2000, Berlin, Germany, 2000.
12. D'Avanzo, E., Magnini, B., Vallin, A. 2004, Keyphrase extraction for summarization purposes: the lake system at duc2004. In: DUC Workshop, Human Language Technology conference / North American chapter of the Association for Computational Linguistics annual meeting. Boston, USA (2004).