# Machine Learning Methods for Keyword Extraction and Indexing

**K S Sampada, N P Kavya**

***Abstract***: ***The digital age results in the creation of massive information. It is a common tradition among the users to digitalize almost every moment of daily life, since it has become convenient to fetch the information as and when needed from the Internet. User can able to retrieve information by providing query keyword. The objective of the search is to quickly return the set of most relevant documents given a search string. Accomplishing this task for a fixed query involves determining the most relevant documents form the big-data. Queries given to the IR systems are enabled by the keywords. Keyword extraction is a process of identifying the document. Manual keyword extraction is cumbersome and it is in feasible to efficiently identify all the keywords in the document. Therefore the machine learning approaches for keyword extraction are proposed. In this paper various machine learning approaches have discussed along with its merits and de-merits. Here we are also proposing a trained index structure which is efficient to identify the specific locus of the record.***

***Keywords*** **:** ***Keyword extraction, Indexing, Information system, Machine learning approaches***

## I. INTRODUCTION

The keywords are abstract form of the document. They assist the user to find out whether the given document is in the user's fields of interest. Keywords can be referred among multiple corpora and IR systems as they are self-determining. Modularity of the IR systems can be progressed by applying keywords. Documents can be classified or categorized by the keywords. Keyword Query processing comprises of two main stages keyword extraction and indexing.

Keyword extraction (KE) is a process of identifying set of terms that describes the theme of document. Mapping keywords to documents manually could be time inefficient and also an arid task. The digital documents are increasing rapidly hence, mechanized keyword extraction has been the area of research interest. To build an index for a document corpus the extracted keywords can be used. , building an effective index model for text representation with the exponential growth of data becomes a paramount importance. Novel techniques for KE encounter the problems of scalability.

Index structures are the solution to efficient data access. Different indexing methods have different approaches to identify the records within the corpora.

BTree-Index is used to find the locus of the records for the given range requests where the records are arranged in order. Hash-Index is an approach for an unorganized set of documents to find the locus of a record and a BitMap-Index technique can be exploited as filter to find the presence of data record.

This work presents a summary of the machine learning approaches for keyword extraction and the indexing structures in the graph-based representation. Text can be represented as a graph where words are denoted as nodes and their relations as links. The cluster of correlated documents can also be stated as Linked Data. The basics of these applications are to generate the "links" between objects (nodes). Using Resource Description Framework (RDF), data from multiple websites or databases together can be linked. Linked Data has provided significant improvements in search over the internet.

## II. RELATED WORK

### *A. Keyword extraction*

Machine learning techniques for keyword extraction are broadly categorized as: a) unsupervised and b) supervised

➢ *Unsupervised approach* require no prior annotations. They do not require the training data and hence it is unsupervised.

Further the keyword extraction can be classified as statistical and linguistic. The *statistical approach* are independent of language and domain. Therefore these approaches do not require the training data and hence it is unsupervised. *Linguistic approach* use the properties of the languages like words, sentences and documents. They are language dependent and they do require training and this approach can be supervised or unsupervised learning. The unsupervised techniques have been summarized in the table1.

**Table 1. Unsupervised techniques for keyword extraction**

| Reference Paper | Machine Learning Approaches | Techniques | Remarks |
|---|---|---|---|
| HaCohen-Kerner [1] | N-grams (N=1,2,3) | Statistical approach by extracting keywords from abstracts and titles. | Keywords from the body of the document is not extracted |
| Pasquier [2] | Markov Cluster Process (MCP), K-means and ClassDens | Latent Dirichlet Allocation (LDA) a Linguistic approach is used for sentence clustering | keyphrase extraction can be done for a single document |
| Pudota et al [3] | Association mining | Statistical n-gram technique in defining candidate phrases | Can be extracted from single document |

**Revised Manuscript Received on December 22, 2019.**

***** Correspondence Author

**K S Sampada***, Dept. of CSE, RNSIT, VTU, Bengaluru, India. Email: k.s.sampada@gmail.com

**N P Kavya**, Dept. of CSE, RNSIT, VTU, Bengaluru, India. Email: npkavya@gmail.com

*Retrieval Number: B10041292S19/2019©BEIESP*
*DOI: 10.35940/ijitee.B1004.1292S19*

*Published By:*
*Blue Eyes Intelligence Engineering*
*& Sciences Publication*

➢ *Supervised techniques* require a data source which to be annotated. Supervised machine learning methods are trained on a set of keywords to induce a model. The training dataset requires physical annotations which might be inconsistent and laborious. Therefore the supervised techniques can be used as binary classification to verify whether the keyword extracted is either a keyword or not. The supervised techniques have been summarized in the table2

**Table2. Supervised techniques for keyword extraction**

| Reference Paper | Machine Learning Approaches | Techniques | Remarks |
|---|---|---|---|
| Hulth[4] | N-gram | Linguistic approach by including POS tagging | The term choices are independent of the obtained results |
| Turney [5] | Naive Bayes | statistical connotations are used in KEA( Keyphrase Extraction Algorithm) | Rises the consistency of the Extracted keywords |
| HaCohen-Kerner et al.[6] | J48 decision tree | automatic extraction and learning of keyphrases are explored | Semantic information is not considered. |
| Medelyan and Witten[7] | Naive Bayes | KEA++, a novel technique proposed improves automatic keyphrase extraction on terms and phrases obtained from a domain-specific lexicon by using semantic information. | Uses only global context information. |
| Wang[8] | Neural networks | TF and IDF is used as a feature vector | Training the network is time consuming |
| Nguyen and Kan[9] |  | Salient morphological phenomena are captured using Linguistic approach | All features are not captured appropriately. |
| Zhang C. et al.[10] | SVM, Multiple Linear Regression model. | Conditional Random Fields[CRF] which uses linguistic approach by sequence labeling |  |
| Krapivin et al.[11] | SVM, Random Forest | Linguistic approach | Outperforms the KEA approaches. |

The exploration of new techniques of keyword extraction has become tedious with enormous collection of data and Web 2.0 tools. Semi-supervised methods have gained research focus to improve the performance of keyword extraction on enormous quantities of data. Supervised approaches have two vital issues i.) Manually annotating keywords for training data is a challenge ii.) Annotations done manually can be a concern of bias for the training data. This vast amount of data is growing every day, render an apt way of searching by the development of the Semantic Web, a graphical representation of the text. Hence this work focuses on graph-based unsupervised methods of query keyword processing.

The edge in the graph which represents the relationship between the words can be constituted by exploiting the principles such as:

I. *Co-occurrence relations*: in linguistic sense, the occurrences of the words together or simultaneously in a given document.

II. *Syntactic relations*: connecting the words based on lexical/ syntactic analysis.

III. *Semantic relations*: connecting words based semantic [synonyms, antonyms, homonyms] analysis.

The graph based techniques uses various text summarization methods for automatic indexing in query keyword processing.
The graph based techniques have been summarized in the table 3.

**Table 3. Graph based techniques for keyword extraction**

| Reference Paper | Techniques | Remarks |
|---|---|---|
| Ohsawa et al.[12] | KeyGraph method an automatic indexing constructed from metaphors by co- occurrence graphs. | Indexing is not domain dependent and it is content sensitive. |
| Matsou [13] | indexing system is known as Keyworld is constructed by Extracting pair of words, | Based on syntactic relations and the closeness to the centrality of the vertex. |
| Erkan and Radev [14] | Indexing is based on Lexrank by extracting most important sentences and cosine similarity. | Oblivious to the noise in data |
| Mihalcea and Tarau [15] | Indexing is done using Textrank by extracting and preprocessing lexical units | Only syntactical relations are captured. |
| Litvak and Last [16] | Syntactic representation are the basic idea of extracting keywords from text and web documents. They used HITS for text summarization | Degree based indexing |
| Tsatsaronis et al .[17] | SemanticRank algorithm is used to extract the keywords based on the network of entities. | Semantic relation is calculated between the linguistic entities as a closeness measure among them. |
| Zhou et al. [18] | Graph is constructed with keywords using Jaccard coefficient similarity measure. Closeness centrality is a measure to find keyword candidate as a vertex. | this approach outperforms the traditional TF_IDF on accuracy, recall and F-score. |

The keyword extraction and automatic indexing on graph based methods majorly depends on the structure of the graph and the centrality measure is its critical property.

## B. INDEXING

In order to perform the retrieval task from enormous datasets stored in distributed environment like cloud, which are also scalable indexing is done on those datasets [19].

Indexing methodology may improve the performance of data query operations with efficiency and high-throughput as it is impractical to look-up manually on such records [20]. Hence there is a need to have an effecient indexing methodology to access big data effectively. Researchers have used various indexing procedures with focus on big data. For example, precise searches for big data in cloud can be done using semantic indexing-based approach [21], File index works well for for efficient event stream in terms of large text collection in cloud computing [22,23] and to provide multi-dimensional data indexing in cloud, R-tree- based indexing [24].

Performance of traditional indexing approaches, such as bitmap, hashing, B-tree, and R-tree, for big-data is moderate, but they fail to detect ‘unknown’ behavior. They can be considered as classifiers for indexing.

Semantic indexing techniques which are Rule based, are not viable for the unknown pattern of big-data as they implements only known pattern for text summarization.

Consequently there is a need to address an efficient indexing technique for keyword query processing. The mechanism which efficiently indexes and searches the position-specific records has to address some of the challenges. Primarily, the range and text are two unlike data types entailing diverse data structures. Secondly, the ordering and exploration techniques are not being alienated. Thirdly, the hybrid technique of combining spatial relevance and textual relevance should fetch user the top-k results.

## III. RESULTS AND DISCUSSIONS

Among the Keyword extraction techniques explained in section 2, TF-IDF is the popular technique used in information retrieval which creates an inverted index. Inverted index is a structure consisting of keywords and the frequency of its occurrences. Each time the term appears it must be added to a list in the inverted index. That may lead to a quite considerable index overhead. For faster search we can choose structures such as BTree, hashindex. Each one of them has its own advantages and disadvantages. BTree-index are the best choice for range requests. Data distribution is not leveraged while indexing by any of the above tree.

Most of the recent data are mutidimensional which are featured with both textual and spatial content. Range inverted index [30] a hybrid approach can be used to retrieve the query results by considering spatial and textual information. R* tree is most relevant data structure used to represent multidimensional indexing, which identifies the spatial content [location] corresponding to the query keyword. Within that result the TF-IDF index for textual content is constructed which proves to be the best choice for textual indexing. Inverted index can be implemented as BTree, hash table, etc,. BTree-index is the best choice for handling range queries and to find the locus of a key in the sorted array as they are balanced and cache efficient trees. In the real-world use cases, the data does not perfectly follow a known pattern, and it is not worthwhile to engineer a specialized index for every use case.

In this paper, we propose an ingenious approach where the existing index structure (BTree) for search can be modeled as more flexible, faster using machine learning approaches. Data patterns, correlations of the data, etc. are analyzed to train the model. We propose to integrate R* tree and BTree for multidimensional indexing and searching to leverage the distribution of data for indexing. R* tree to identify the spatial content of the query keyword and trained BTree model to efficiently handle the range queries.

*R*Tree:* R* tree is most common spatial data structure. R-tree can contain rectangles, also extend to 3 or more dimensions. Each node has a fixed number of children. Fig 1 represents r-tree with 3 points.

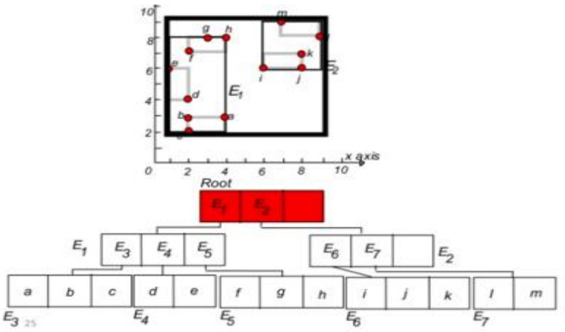

**Fig1. R-tree representation**

*Trained Btree*: B-Trees can be modeled as a form *f (key)→pos*, where *f* is a function which accepts key value and returns the locus of that key . Consider machine learning notation where the key is represented by *x* and the position is *y*. Because the data *x* is sorted, *f* can be modeled as the cumulative distribution function (CDF) of data. We train our model *f (x)* using ANN with backward propagation where the squared error is being propagated backwards to adjust the weights as shown in Fig 2.. This network is built at different stages as a hierarchy of models as shown Fig 3. We consider that we have y [0, N] and at stage A there are MÆ models. We train the model at stage 0, $f_0(x) = y$.

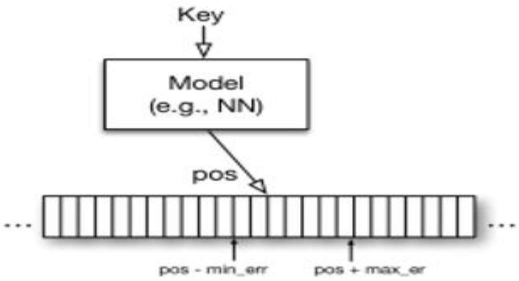

**Fig 2. Trained B-tree representation**

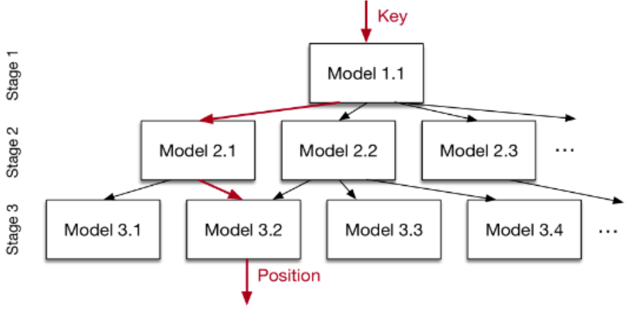

**Fig 3. B-tree training at 3 levels**

## IV. CONCLUSION

In this paper various keyword extraction methods with machine learning approaches are discussed. Keyword extraction with machine learning techniques can be categorized as supervised and unsupervised. Keyword extraction can also be classified as statistical approach and linguistic approach. In case of statistical approach they are language and domain independent and hence they can be considered as unsupervised approach. Whereas linguistic approaches are language dependent and is supervised. Here we are also proposing a unified index structure which is efficient to identify the specific locus of the record. R* tree is considered for multidimensional index to find the spatial data and within a specified range an inverted index is built for the given query which is a text. The inverted index is which is implemented as B-tree can be modeled as a regression and can be trained recursively using ANN with backpropogation. Unlike traditional machine learning model this model would provide us the specific locus of the record.

## REFERENCES

1. HaCohen-Kerner, Yaakov. "*Automatic extraction of keywords from abstracts*." In International Conference on Knowledge-Based and Intelligent Information and Engineering Systems, pp. 843-849. Springer, Berlin, Heidelberg, 2003.
2. Pasquier, Claude. "*Task 5: Single document keyphrase extraction using sentence clustering and Latent Dirichlet Allocation.*" In Proceedings of the 5th international workshop on semantic evaluation, pp. 154-157. Association for Computational Linguistics, 2010.
3. Pudota, Nirmala, Antonina Dattolo, Andrea Baruzzo, and Carlo Tasso. "*A new domain independent keyphrase extraction system*." In Italian Research Conference on Digital Libraries, pp. 67-78. Springer, Berlin, Heidelberg, 2010.
4. Hulth, Anette. "I*mproved 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, 2003..
5. Turney P. D. "*Coherent Keyphrase Extraction via Web Mining*". In Proceedings of the 18th International Joint Conference on AI, IJCAI'03, pp. 434-439, San Francisco, CA, USA, 2003.
6. HaCohen-Kerner, Yaakov, Zuriel Gross, and Asaf Masa. "*Automatic extraction and learning of keyphrases from scientific articles.*" In International Conference on Intelligent Text Processing and Computational Linguistics, pp. 657-669. Springer, Berlin, Heidelberg, 2005.
7. Witten, Ian H., and Olena Medelyan. "*Thesaurus based automatic keyphrase indexing.*" In Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL'06), pp. 296-297. IEEE, 2006.
8. Wang, Jiabing, Hong Peng, and Jing-song Hu. "*Automatic keyphrases extraction from document using neural network*." In Advances in Machine Learning and Cybernetics, pp. 633-641. Springer, Berlin, Heidelberg, 2006.
9. Nguyen, Thuy Dung, and Min-Yen Kan. "*Keyphrase extraction in scientific publications.*" In International conference on Asian digital libraries, pp. 317-326. Springer, Berlin, Heidelberg, 2007.
10. Zhang, Chengzhi. "*Automatic keyword extraction from documents using conditional random fields*." Journal of Computational Information Systems 4, no. 3 (2008): 1169-1180..
11. Krapivin, Mikalai, Aliaksandr Autayeu, Maurizio Marchese, Enrico Blanzieri, and Nicola Segata. "*Keyphrases extraction from scientific documents: improving machine learning approaches with natural language processing*." In International Conference on Asian Digital Libraries, pp. 102-111. Springer, Berlin, Heidelberg, 2010.
12. Ohsawa, Yukio, Nels E. Benson, and Masahiko Yachida. "*KeyGraph: Automatic indexing by co-occurrence graph based on building construction metaphor*." In Proceedings IEEE International Forum on Research and Technology Advances in Digital Libraries-ADL'98-, pp. 12-18. IEEE, 1998.
13. Matsuo, Yutaka, Yukio Ohsawa, and Mitsuru Ishizuka. "*Keyworld: Extracting keywords from document s small world*." In International conference on discovery science, pp. 271-281. Springer, Berlin, Heidelberg, 2001.
14. Erkan, Günes, and Dragomir R. Radev. "*Lexrank: Graph-based lexical centrality as salience in text summarization.*" Journal of artificial intelligence research 22 pp. 457-479, 2004
15. Mihalcea, Rada, and Paul Tarau. "*Textrank: Bringing order into text*." In Proceedings of the 2004 conference on empirical methods in natural language processing, pp. 404-411. 2004.
16. Litvak, Marina, Mark Last, Hen Aizenman, Inbal Gobits, and Abraham Kandel. "*DegExt—A language-independent graph-based keyphrase extractor.*" In Advances in Intelligent Web Mastering–3, pp. 121-130. Springer, Berlin, Heidelberg, 2011.
17. Tsatsaronis, George, Iraklis Varlamis, and Kjetil Nørvåg. "*SemanticRank: ranking keywords and sentences using semantic graphs*." In Proceedings of the 23rd International Conference on Computational Linguistics, pp. 1074-1082. Association for Computational Linguistics, 2010.
18. Zhou, Zhi, Xiaojun Zou, Xueqiang Lv, and Junfeng Hu. "*Research on weighted complex network based keywords extraction.*" In Workshop on Chinese Lexical Semantics, pp. 442-452. Springer, Berlin, Heidelberg, 2013.
19. Chen, Jinchuan, Yueguo Chen, Xiaoyong Du, Cuiping Li, Jiaheng Lu, Suyun Zhao, and Xuan Zhou. "*Big data challenge: a data management perspective.*" Frontiers of Computer Science 7, no. 2 (2013): 157-164.
20. Wang, Miao, Viliam Holub, John Murphy, and Patrick O'Sullivan. "*High volumes of event stream indexing and efficient multi-keyword searching for cloud monitoring*." Future Generation Computer Systems 29, no. 8 (2013): 1943-1962.
21. Rodríguez-García, Miguel Ángel, Rafael Valencia-García, Francisco García-Sánchez, and J. Javier Samper-Zapater. "*Creating a semantically-enhanced cloud services environment through ontology evolution*." Future Generation Computer Systems 32 (2014): 295-306.
22. Bast, Hannah, and Marjan Celikik. "*Efficient fuzzy search in large text collections*." ACM Transactions on Information Systems (TOIS) 31, no. 2 (2013): 10.
23. Wang, Jinbao, Sai Wu, Hong Gao, Jianzhong Li, and Beng Chin Ooi. "*Indexing multi-dimensional data in a cloud system.*" In Proceedings of the 2010 ACM SIGMOD International Conference on Management of data, pp. 591-602. ACM, 2010.
24. Sampada, K. S., Lalit Adithya, and N. P. Kavya. "*Performance Analysis of Multidimensional Indexing in Keyword Search.*" In Proceedings of International Conference on Recent Advancement on Computer and Communication, pp. 171-184. Springer, Singapore, 2018.

## AUTHORS PROFILE

Mrs K. S. Sampada currently working as assistant professor , Dept. Of CSE, RNSIT. She has an overall of 16.5 years experience with 3.5 years of industrial experience and 13 years of teaching with 4 years of research. She has been awarded BE and M.Tech Degree in specialization in Computer Science and Engineering. Currently pursuing PhD and her areas of interests are big-data,machine learning, Information retrieval and Text Mining in Visvesvaraya Technological University, Karnataka. She has published around 9 research papers in reputed international journals including IEEE, Springier.

Dr. N P Kavya holds Bachelor of Engineering in Computer Science and Engg. along with MS in software systems and Ph.D in computer science form VTU Belagavi . She has a vast experience of 24 years in the field education and research. She is currently a Professor in Computer science and Engg., Department , RNSIT, Bengaluru. She has published around 90 research papers in reputed international journals including IEEE, Elsevier , Springier ( SCI and Web of Science) . Has 94+ citations in Google scholar as on Oct 2019. Her main areas of expertise are Machine Learning, Artificial Intelligence, Big Data analytics etc.,