Design an Approach for Finding the Similarity between the Documents

Ms. Shilpa Satone
Dept of Wireless Communication
Tulsiramji Gaikwad-Patil College ofTechnology, Nagpur

Prof. Jayant Adhikari
Dept of Wireless Communication
Tulsiramji Gaikwad-Patil College of Engg & Technology, Nagpur

Prof. Jayant Rohankar
Dept of Information Technology& Computing
Tulsiramji Gaikwad-Patil College of Engg & Technology, Nagpur

Abstract:-Now a days Data Management is very important issue. Data on cloud is very large in size. Web users need tools to manage information easily. If tried to do manually this is cumbersome and time consuming process because there are many near-duplicate results. The efficient detection of near-duplicate articles is very important in many applications that have a large amount of data available for a specific requirement depending upon the task in hand. We are introducing algorithm for extracting key-phrases and matching signatures for near-duplicate articles detection. Based on N-gram (i.e. bigram & trigram) algorithm for key phrase extraction & JACCARD similarity for finding similarity between documents. Algorithms are applied on article and text Documents and result shows that our proposed methods are more effective than other existing method.

Keywords: *Keyphrase, Similarity, Extraction, Near-Duplicate.*

*****

I. INTRODUCTION:

The word keyphrase implies two features: phraseness and informativeness. It is the process of obtaining the keyphrases which are available in the body of the text document. Single document keyphrase extraction usually make use of only the information contained in the specified document. It is used to extract most frequent words which are significant with respect to the applications. This paper discusses various algorithms and tools for keyphrase extraction from documents. Application areas are also discussed.Keyphrase can be defined as a phrase of one to three words to capture the main topic.

In this project, we proposed 3approach Bigram, Trigram and Cosine similarity: algorithm of keyphrase extraction and algorithm of near-duplicate article detection. We also have given the established structures for keyphrases as well as combined order of them. This helps algorithm of keyphrase extraction is more exact. Based on a set of characteristic keyphrases in each article, algorithm of near-duplicate article detection is also presented. Experiment results show that the precision and recall of algorithms are good.

In the future, we are going to apply other similarity index calculations algorithms to improve more accuracy in many various fields.

II. RELATED WORK:

Here the author Nhon Do, LongVan Ho [2] propose a system Based on philosophy, key expressions of articles are removed naturally and likeness of two articles is computed by utilizing extricated key expressions. Calculations are connected on Vietnamese online daily papers for Labor and Employment. Exploratory results demonstrate thatour proposed strategies. The noteworthy increment in number of the online daily papers has given web clients a monster data source. The clients are truly hard to oversee content and check the accuracy of articles. In this paper, we present calculations of removing keyexpression and coordinating marks for close copy articles recognition. In view of cosmology, key expressions of articles are extricated consequently and comparability of two articles is computed by utilizing separated key expressions. Calculations areconnected on Vietnamese online daily papers for Labor and Employment. Exploratory results demonstrate that our proposed strategies are viable.

The author F. Sun, Y. Wang, J. Lu, B. Zhang, W. Kinsner & L.A. Zadeh [3] propose way to deal with the securing of the semantic components inside of expressions from a solitary archive. is proposed in this paper, Our trials show that the proposed key expression extraction technique dependably beats the gauge strategies TFIDF and Kea. Moreover, our methodology is not space particular and the technique sums up well when it is prepared on one area (diary articles) and tried on another (news site pages).

The author Luı´s Leitao, Pavel Calado, and Melanie Herschel [4] propose a novel technique for XML copy discovery, called XMLDup. XMLDup utilizes a Bayesian system to decide the likelihood of two XML components being copies, considering the data inside of the components, as well as the way that data is organized. What's more, to enhance the productivity of the system assessment, a novel pruning methodology, equipped for noteworthy increases over the upgraded variant of the calculation, is displayed. Through trials, we demonstrate that our calculation can accomplish high accuracy and review scores in a few information sets. XML Dup is additionally ready to beat another best in class copy recognition arrangement, both as far as proficiency and of viability.

The author Ahmed K. Elmagarmid, Panagiotis G. Ipeirotis, and Vassilios S. Verykios [5] show an exhaustive examination of the writing on copy record recognition. We cover closeness measurements that are usually used to identify comparable field sections, and we introduce a broad arrangement of copy discovery calculations that can distinguish roughly copy records in a database. We likewise cover different strategies for enhancing the proficiency and adaptability of estimated copy discovery calculations. We finish up with scope of

existing devices and with a brief exchange of the huge open issues in the zone. Frequently, in this present reality, elements have two or more representations in databases. Copy records don't share a typical key and/or they contain mistakes that make copy coordinating a troublesome errand. Mistakes are presented as the consequence of interpretation blunders, fragmented data, absence of standard organizations, or any mix of these components.

III. PROPOSED METHODOLOGY:

In this paper we implement the system for three phases like,
Phase-1(Preprocessing)
Phase-2(Keyphrase Extraction)
Phase-3(Finding similarity between documents)

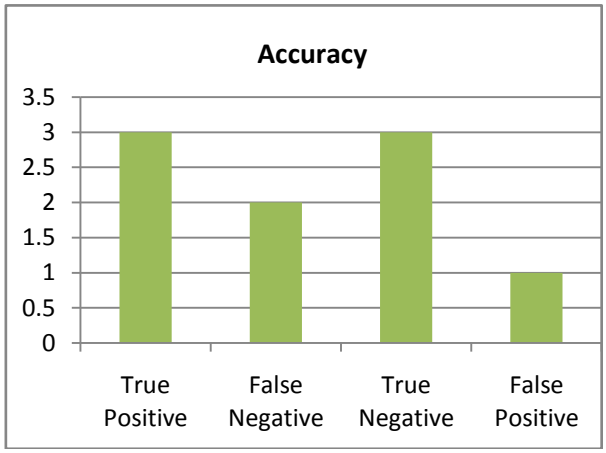
Phase-I:
In the proposed system phase-I we apply the Preprocessing Technique

Phase-II:
In the proposed system phase-II we use N-gram(i.e. bigram & trigram) algorithm for keyphrase extraction. After that implementation we implement third Phase.

Phase-III:
In Phase-III we usedJaccard similarity Function for finding similarity between the various documents. The Jaccard coefficient, which is sometimes referred to as the Tanimoto coefficient, measures similarity as the intersection divided by the union of the objects.For text document, the Jaccard coefficient compares the sum weight of shared terms to the sum weight of terms that are present in either of the two documents but are not the shared terms.
Definitions:Sim (dj, q) : This is the whole point of the exercise. We calculate a similarity score between our query and every document in the collection. (For most of these the document won't match at all and the score will be zero.) Then we order our retrieval from high score (the best possible is 1) to low.
q:q is the query. A query is a set of terms with each term given a weight between zero and one.
Remember, you can always give every term in the query a weight of one thus making them equally
important. To make the formula completely general we assume that a query gives a weight to every
term (and there are a total of t terms) used in the database. Thus, wi,qis the weight that term i has in the query. Remember, in any real query, almost every term in the database has a weight of zero, which is to say, it is not part of the query.
dj: this is one of the documents in the collection. A document is described by a set of terms with each term given a weight between one (this is a very important term for describing the document) and zero (this term has nothing to do with the document.) Again, to make the formula completely general we assume that for each document there is a weight for every term used in the database even though most terms will get a weight of zero. Remember that we are calculating this with a computer, so who cares about adding a bunch of extra zeros.
Thus;
wi,j is weight that term i has for document j.
Sim (dj, q) = ∑i=all t(w i,j x wi,q) / ((∑i=all tw2i,j)½x (∑i=all tw2i,q)½)
dj is the document, represented here by the blue arrow. And since we have only one document, we can call j = 1.
q is the query and is represented by the red arrow.
t is the total number of terms in our space. In this very simple case t=3.
Cosine Similarity Coefficient
Input:2 vectors
Output: The cosine similarity

```
cosine_ similarity(vector1,vector2):
//Calculate numerator of cosine similarity
dot = [vector1[i] * vector2[i] for i in range(vector1)]
//Normalize the first vector
sum_vector1 = 0.0
sum_vector1 += sum_vector1 + (vector1[i]*vector1[i] for i in range(vector1))
norm_vector1 = sqrt(sum_vector1)
//Normalize the second vector
sum_vector2 = 0.0
sum_vector2 += sum_vector2 + (vector2[i]*vector2[i] for i in range(vector2))
norm_vector2 = sqrt(sum_vector2)
//Calculate and return final similarity
return (dot/(norm_vector1*norm_vector2))
```

IV. RESULT AND DISCUSSION:



True Positive	False Negative	True Negative	False Positive
3	2	3	1

PRECISION & RECALL

Precision and recall are the basic measures used in evaluating search strategies.
As shown in the first two figures on the left, these measures assume:
There is a set of records in the database which is relevant to the search topic Records are assumed to be either relevant or irrelevant (these measures do not allow for degrees of

relevancy). The actual retrieval set may not perfectly match the set of relevant records.

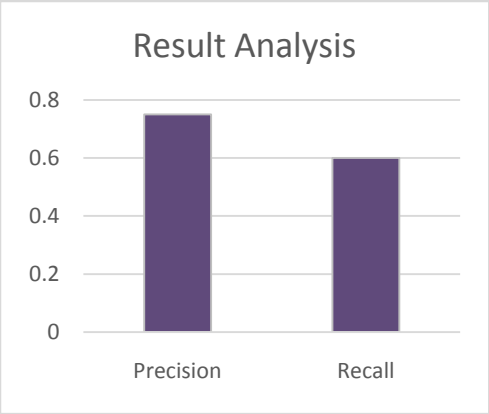
PRECISION: is the ratio of the number of relevant records retrieved to the total number of irrelevant and relevant records retrieved. It is usually expressed as a percentage.

PRECISION: ((A/A+C)*100%)

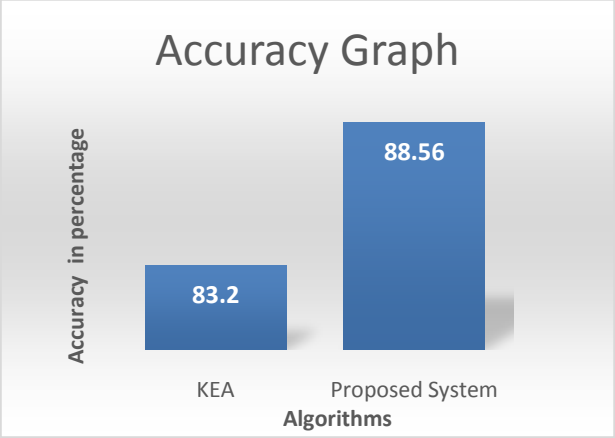
RECALL: is the ratio of the number of relevant records retrieved to the total number of relevant records in the database. It is usually expressed as a percentage.

RECALL: ((A/A+B)*100%)

Precision	0.75
Recall	0.6



KEA	83.2
Proposed System	88.56



V. CONCLUSION:

In this project, we proposed 3approach Bi-gram, Tri-gram And Cosine similarity: algorithm of keyphrase extraction and algorithm of near-duplicate article detection. We also have given the established structures for keyphrases as well as combined order of them. This helps algorithm of keyphrase extraction is more exact. Based on a set of characteristic keyphrases in each article, algorithm of near-duplicate article detection is also presented. Experiment results show that the precision and recall of algorithms are good.

In the future, we are going to apply other similarity index calculations algorithms to improve more accuracy in many various fields. We will extensively explore the feasibility of these ideas.

VI. REFERENCES:

[1] Second International Conference 0n Power, Circuit and Information Technologies(ICPCIT-16),Shilpa Satone, Jayant Rohankar,ISBN978-93-81693-07-1,9RRCE) MAY 2016, An Approach To Measure The Extent of Similarity Between two Text Documents using N-Gram.

[2] The 2015 IEEE RIVF International Conference on Computing & Communication Technologies Research, Innovation, and Vision for Future (RIVF) Domain-Specific Keyphrase Extraction and Near-Duplicate Article Detection based on Ontology

[3] Proc. 9th IEEE Int. Conf. on Cognitive Informatics (ICCI'10) F. Sun, Y. Wang, J. Lu, B. Zhang, W. Kinsner & L.A. Zadeh (Eds.) 978-1-4244-8040-1/10/$26.00 ©2010 IEEE, Keyphrase Extraction Based on Semantic Relatedness

[4] 1028 IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 25, NO. 5, MAY 2013, Luı´sLeitao, PavelCalado, and Melanie Herschel Efficient and Effective Duplicate Detection in Hierarchical Data

[5] IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 19, NO. 1, JANUARY 2007, Ahmed K. Elmagarmid, Senior Member, IEEE, Panagiotis G. Ipeirotis, Member, IEEE Computer Society, and Vassilios S. Verykios, Member, IEEE Computer Society, Duplicate Record Detection: A Survey.

[6] International Conference on Pervasive Computing (ICPC), ShitalGaikwad, NagarajuBogiri, A Survey Analysis On Duplicate Detection in Hierarchical Data.

[7] International Journal of Engineering Science Invention ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726 www.ijesi.org Volume 2 Issue 6 ǁ June. 2013 ǁ PP.75-78 ,Sapna Chauhan1, Pridhi Arora2 ,Pawan Bhadana3, Algorithm for Semantic Based Similarity Measure.

[8] International Journal of Computer Applications (0975 – 8887) Volume 68– No.13, April 2013, Wael H. Gomaa, Aly A. Fahmy, A Survey of Text Similarity Approaches.

[9] 978-1-4799-0174-6/13/$31.00 ©2013 IEEE, HidekazuYanagimoto, Mika Shimada, Akane Yoshimura, Document Similarity Estimation for Sentiment Analysis Using Neural Network.

[10] WWW 2007, May 8–12, 2007, Banff, Alberta, Canada. ACM 9781595936547/07/0005., Gurmeet Singh Manku, Arvind Jain, Anish Das Sarma,Detecting Near Duplicates for Web Crawling.

[11] Proceedings of the 5th International Joint Conference on Natural Language Processing, pages 1001–1009,Chiang Mai, Thailand, November 8 – 13, 2011. c2011 AFNLP, Yan Wu, Qi Zhang, Xuanjing Huang Efficient Near-Duplicate Detection for Q&A Forum.

[12] 2nd International Conference on Future Computer and Communication [Volume 2], 978-1-4244-5824-0/$26.00 c 2010 IEEE, JunpingQiu, Qian Zeng, Detection and Optimized Disposal of Near-Duplicate Pages.