IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
ISO 3297:2007 Certified
Vol. 5, Issue 6, June 2018

Contextual Query-Driven e-News Summarisation

Kailash Karthik S[1], Dakshina K[2], Hariprasad KR[3], Abha Suman[4]

Department of Computer Science and Engineering, National Institute of Technology, Tiruchirapalli, India [1, 2, 3, 4]

Abstract: Given the phenomenal abundance in quantity of news available over the electronic medium, extraction of consolidated information from raw data has become important. Extraction of relevant textual content from E-News portals has become increasingly challenging due to CMSs and the dynamic nature of web pages. In this paper, a technique that provides condensed news information based on user preferences is presented. Information retrieval from the articles is based on TF-IDF, augmented by a novel transitive closure algorithm that adds context to the search. The impact of techniques like article tagging and coreference resolution on the retrieval of semantically and contextually related content is studied. The proposed system improves search latency through the separation of text processing and information extraction processes. Once information is extracted, summarized content based on user queries from different source portals is presented. The system achieves high precision and recall for both generic and domain-specific user queries.

Keywords: Information retrieval, natural language processing, text summarisation, text mining

I. INTRODUCTION

The explosive growth of the internet and its information services has created a problem of information overload. The magnitude of information available has risen exponentially, making it impossible to consume it all. Extraction of relevant information has thus become indispensable at this juncture. Since the internet is a provider of superfluous information, there is a two-fold concern: searching for relevant documents among a prodigiously large document base, and selectively extracting germane content from the documents deemed relevant. Hence, a good summary must be diverse yet concise, and devoid of redundancies. Text summarization techniques can be classified as abstractive and extractive. An extractive summary concatenates important sections of the original document, evaluating the sections based on statistical and linguistic features. An abstractive summary attempts to comprehend the main concepts and relationships in a document and expresses them in an alternate, diminutive form. With the increase in number of news sources coupled with the existing information explosion predicament, there is a need to present users with abridged versions of news articles to facilitate a news reading experience that is faster but comprehensive nonetheless. This has already been tended to by the Android application InShorts, which aggregates and delivers news articles in 60 words or less. But with the inherent media bias, there is a need to consume news across sources to distil facts from opinions. In addition to this, an article may necessitate different summaries depending on the context of retrieval, which is facilitated by query based summaries. Thus, in this paper, an algorithm that aggregates news from a multitude of sources, extracting and summarizing information based on user preferences is proposed. A web news page contains in it information that may not all be relevant for an application. For instance, an article classifier may only require the title of the article, while a sentiment analysis engine may require only the body of the article. So the extraction of appropriate content becomes decisive. Extraction of text information from web pages is complicated, due to dynamically generated HTML from Content Management Systems that encompass front end scripts and advertisements among other noise. This dynamic generation also often leads to malformed tags making web page parsing complex. The gathering of information from multiple sources is essential to a holistic knowledge base. So the proposed technique gathers data from a multitude of electronic news portals. This is achieved by systematically crawling multiple news providers' repositories and local indexing on a regular basis. The pages obtained may not be well-formed. However, proper XHTML formatting is required for DOM parsing. They are thus processed, removing noise in the form of advertisements, scripts and social media discussions. A voluminous text base makes processing every article for each user request drastically worsen the search latency. To reduce time complexity, the articles are tagged using representative keywords extracted from their body. Article search is then facilitated using the generated tags, analogous to question-answering platforms prevalent in the internet like Stackoverflow. In order to incorporate the context of the user's query rather than just considering it as a bag-of-words, the proposed system employs a transitive closure algorithm using article tags. This tag closure algorithm enriched the user query and is shown to improve the recall of the system. The contents of the articles are then evaluated to find information blocks that are relevant to the search query. The system employs the TF-IDF algorithm to extract summaries, ranking and ordering different blocks based on content. Alternate forms of content words are taken into consideration by generating synsets to prevent linguistic bias in the search towards specific terms in the user's query. Coreferences in the articles are resolved to aid the performance of the existing TF-IDF based information extraction technique. This results in equating synonymous concepts in the

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
ISO 3297:2007 Certified
Vol. 5, Issue 6, June 2018

news content and mitigates one of the major drawbacks of the pure bag-of-words approach that TF-IDF provides. The rest of the paper is organized as follows. Section II discusses the preceding works in this domain. Section III details the proposed system for news extraction and summarization. Section IV presents the experiments conducted with their empirical results. Section V concludes with some final remarks and proposals for future work.

II. RELATED WORK

Research on information extraction has yielded a diverse range of algorithms and techniques to be employed in information processing systems. The major issues in information extraction, which range from named-entity recognition to handling data from unstructured or semi-structured web pages were discussed in [1]. A number of extraction methods were discussed which include extraction rules by encoding patterns, sequence labelling and a statistical model using Hidden Markov Models.

The existing information extraction methodologies were classified in [2] into the following categories: Rule-Learning, Classification-based and Sequence Labelling-based extraction models. Enhancements to each type of technique were proposed and analyzed. The authors further discussed a generative model based on Hidden Markov Models and its application in domains like digital libraries and internet mailing.

Information extraction can be approached as a two-stage problem as in [3], comprising of entity extraction and relationship extraction. Entities were determined using statistical methods using token-level and segment-level models and their relationships, using feature and kernel-based models. It was inferred that machine learning models can be trained using labelled data to perform the task. Two scenarios exist when extracting relationships: one where the goal is to classify the relationship type that exists between a given entity pair using an annotated corpus, and second to retrieve all instances of entity pairs for a given type of relationship as a semi-supervised method.

A news extraction system can be based on lexical chains, performing POS-tagging and synset generation to determine strong lexical chains as discoursed in [4]. The system presented also handled video segmentation based on a color-histogram based algorithm. Generation of lexical chains can leverage WordNet, but this technique leads to a search space explosion problem. Though search space pruning was suggested as a practical solution, optimality of the search can't be guaranteed.

Text summarization was classified in [5] based on different typologies, presenting a comprehensive taxonomy of text summarization. The influence of word semantics in extracting important sentences in an extractive summary was enumerated. Various features that may be used to extract information include content and title words, sentence location, sentence length and the presence of proper nouns. Techniques including TF-IDF statistics, graph theory, fuzzy logic, neural networks and LSA can be used for text summarization. These summarization techniques were compared against each other in [6]. Further, an evaluation was made between the TF-IDF algorithm (as employed in this paper) and analogous methods based on graph theory, semantic analysis and neural nets.

Abstractive summary generation was discussed in [7], where methods for sentence compression and information fusion were addressed. PCFGs, Integer Linear Programming (ILP) and Information Ordering can be used for generating context dependent summaries. The summaries generated are tested for performance using intrinsic evaluation methods like precision-recall-F1 and ROUGE. The optimal summarization technique may depend on the domain and genre of the text. For example, SVMs that use both linguistic and message-specific features are shown to perform well for email summarization.

The usage of machine learning in extractive summarization was demonstrated in [8]. The features engineered were classified into Surface features like sentence position, Event features like named entities and Content features like n-grams. These features extracted can be used to train a supervised model like Probabilistic SVM or Naïve Bayesian Classifier for the summarization task. The classifier makes a binary decision of whether to include a given sentence in the summary or not. Sentence similarity can be used to minimize redundancy of information in the generated summaries. The TextRank algorithm in [9] makes use of this information to generate a document graph.

The application of the TF-IDF algorithm, as employed in the proposed system, in information extraction was discoursed in [10], where the mathematical expressions related to information theory were presented. One of the main issues with the algorithm is non-linear scaling of the term frequency. Variations of the TF-IDF algorithm based on probabilistic models (p-tfidf) have been proposed to avoid these issues. The alternatives to TF-IDF for text representation are Latent Semantic Indexing and Multi-Word. These methods were evaluated against each other in [11], where the method used in the proposed system was shown to outperform the others statistically, as measured using precision, recall and term percentage. A personalized news search engine based on TF-IDF algorithm was proposed in [12]. It worked in an offline and online phase and performed query based summarization.

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
ISO 3297:2007 Certified
Vol. 5, Issue 6, June 2018

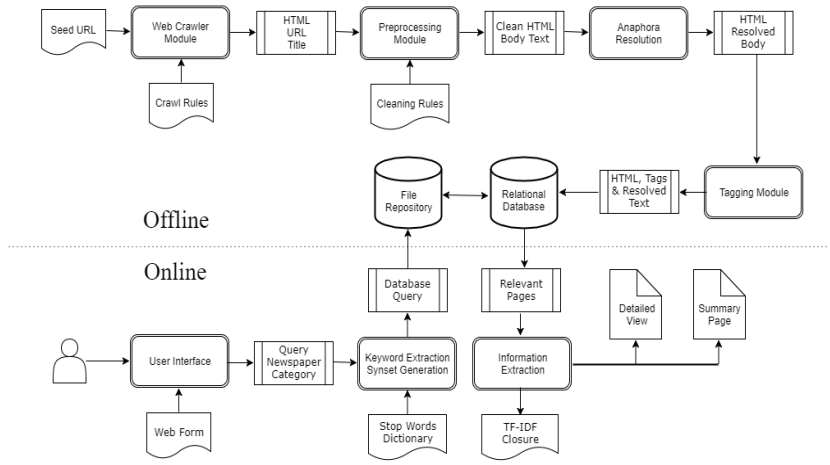

Fig. 1. Architecture of the proposed technique

Retrieval of e-news content requires multi-threaded web crawling of various source portals. Page fetching, parsing and DOM tree generation are the various phases of the crawling process [13]. The various types of web crawlers are: Naive Best-First Crawler, Focused Crawler and Context Focused Crawler. Popular crawling algorithms include InfoSpiders and SharkSearch. Web crawlers are evaluated by their ability to return important pages with short latency. The page importance may be determined by the keywords in the document, similarity to the query or seed pages. Precision-like parameters like Acquisition rate and average relevance are the evaluation parameters used to measure the performance of crawling algorithms.

Keyword extraction is an important component of the proposed system. A survey of the various approaches in text mining for keyword extraction were presented in [14]. The techniques are broadly classified into rule-based, statistical, machine learning (unsupervised, supervised and semi-supervised) and domain-specific approaches. The features used for keyword extraction are based on the phraseness (mutual information and mean/variance) informativeness (term weight and location in document) of candidate terms.

Two recently presented keyword extraction algorithms with good performance are TextRank [15] and RAKE [16]. Text is represented as a weighted graph in TextRank. A graph-based ranking model was then used to perform keyword extraction. During graph generation, syntactic filters can be used to restrict the addition of new vertices based on their part-of-speech. RAKE combined the existing word frequency parameter with word degree to produce better unbiased results. In the algorithm, candidate words were selected from the text in the form of a co-occurrence matrix. Keywords were then extracted using scores generated by a combination of the above mentioned parameters.

Anaphora resolution is the problem of resolving references to earlier or later items in the discourse [17]. Anaphora may occur in one of the following forms: Pronominal, where a referent is referred by a pronoun; Definite noun phrase, where the antecedent is referred by a phrase; or Quantifier/Ordinal, where the anaphor is a quantifier such as 'one' or an ordinal such as 'first'. The issue with anaphora resolution algorithms is their dependence on language and time-direction. A comparison of different pronoun resolution algorithms was carried out in [18]. The RAP algorithm for identifying inter-sentential antecedents of nouns was presented and compared with algorithms like Hobbs' Algorithm. The design and development of Stanford CoreNLP was detailed in [19]. It provides an API-based and command line-based interface to core natural language processing utilities, ranging from tokenization to coreference resolution.

WordNet is a lexical reference inspired by psycholinguistic theories of human memory [20]. WordNet is used in many text processing systems as it provides a lexicon and basic word-related information like word sense and synonymy. The technique used by WordNet in processing words of different parts of speech was also discoursed in the paper, along with its application in a number of diverse fields.

III. PROPOSED SYSTEM

The system's operation is broadly divided into two phases: an offline and an online phase. The architecture of the proposed system is presented in Fig. 1. As an analogy to machine learning paradigms, the system may be viewed as similar to an eager learner, performing a one-time data processing during data retrieval rather than during query-time. Just like a learning algorithm trains on a dataset, the system analyzes the news article set: normalizing the text,

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
ISO 3297:2007 Certified
Vol. 5, Issue 6, June 2018

resolving coreferences and generates topics related to each article for more efficient search in future. The advantage of such a separation of concerns is the improvement in latency of each search execution. Though in such a system, the bootstrapping latency is greater, there is a net gain in execution time when multiple user searches are performed. The improvement in search latency when directly executed on a pre-processed dataset is represented mathematically in Fig.2.
The system consists of the following modules:

A. Offline Phase
1) Web Crawler:
The proposed system consists of a web crawler which crawls through hyperlinks and retrieves information about web pages encountered. Information gathering is performed by means of a topical or focused crawler that accepts input parameters. Seed URLs are provided to specify the base from which crawling is to begin. Crawling rules are specified to ensure that only specific pages are crawled. For instance, news section pages (like Sports) should not be crawled as they contain a list of articles related to a topic and not an individual article in itself. To ensure that only article pages are crawled, page extensions are provided as rule parameters. Other parameters like number of parallel threads, crawling depth are configured. The pages obtained are subsequently stored locally in a database. The HTML source files are also maintained in a file repository maintaining reference to the database keys.

$$Latency = Time_{Bootstrap} + Time_{Search}$$
$$Latency_{Existing} = N * Time_{Bootstrap} + N * Time_{Search}$$
$$Latency_{Proposed} = 1 * Time_{Bootstrap} + N * Time_{Search}$$
$$Latency_{Gain} = 1 + \frac{Time_{Bootstrap}}{Time_{Search}} \ (when\ N\ is\ large)$$

Fig. 2. Improvement in search latency

2) Data Pre-processing:
The pages obtained from crawling are filled with intra-page noise which are incoherent with the page content. The pages are first made well-formed adhering to the WHATWG HTML5 specification so that they can be parsed. The DOM obtained thus is then pruned to remove standard noise and comments. Due to the presence of dynamic content, the scripts and styles are present which are also removed. The body content may also have noise in the form of standard closing textual comments and a user review section. Rules to remove these sections are determined and employed to remove all superfluous text from the documents.

3) Anaphora Resolution:
To improve the efficiency of the term frequency model, the article text processed to resolve coreferences. The coreferences scanned from the articles not only subsumes synonyms but also conceptually related phrases. Inter-sentential references are also resolved within a document. These references are then replaced with the principle reference. This increases the word scores during TF-IDF extraction which results in more relevant sentences being selected for the summary.

4) Article Tagging:
Once the pages are cleaned, the content keywords are extracted using the RAKE algorithm. The algorithm is based on the observation that keywords frequently contain multiple words but rarely contain standard punctuation or stop words, such as function words or other words with minimal lexical meaning. Candidate words are extracted from the text, which are sequences of content words as they occur in the text. Co-occurrences of words within these candidate keywords are meaningful and enable identification of word co-occurrence without the restriction of an arbitrarily sized sliding window. Word associations are thus measured in a manner that automatically adapts to the style and content of the text, enabling adaptive and fine-grained measurement of word co-occurrences that will be used to score candidate keywords.

Ratios of word frequency to word degree are computed for each candidate. While word frequency ensures that regularly occurring words within a document are more favorable as keywords, word degree ensures that topic independent words that occur in many documents are not favored. Keywords are then scored in a co-occurrence graph, the top scoring words being selected as article tags. The tags are then inserted into the database, maintaining referential integrity to the articles present in the database. These tags represent the prevalent concepts in the article and hence, define the context of the news article. These tags are then used for contextual information retrieval.

5) Database Updating:
Once the articles are processed and tags generated, the relational database is updated. The article title, URL, news source, category and tags are inserted into the table. Article contents (raw html, processed and unprocessed) are placed

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
ISO 3297:2007 Certified
Vol. 5, Issue 6, June 2018

in the file repository. Appropriate referential integrity parameters are established to avoid inconsistencies in data.

B. Online Phase

1) User Interface:

To use the proposed algorithm for news extraction and summarization, a web user interface to provide the search preferences is designed. These preferences include the query to be processed and the news source and category of interest. The interface should receive the preferences, perform appropriate validations and invoke the extraction and summarization algorithms in sequential order. The information returned is made available on the user interface.

2) Keyword Extraction:

The user's query is processed before information extraction. Stop words are removed to retain only the key terms. To do this a standard stop list like the Fox Stop list is used. The words are stemmed using a Porter Stemmer implementation to normalize content for matching. It is important to consider only the lemmas of words and not their lexical variations. Further, in order to consider alternate forms of the stemmed words, synsets are generated for each key term, using the online dictionary WordNet. This enables the system to convert the bag-of-words into a bag-of-similar-words. Thus, a synset of the stemmed keywords in the user's query is obtained. This set represents the normalized version of the user's query and its similarity variations.

3) Data Retrieval:

The user's news source and category preferences are formulated into a query to the data store. The returned set of articles are then processed to find relevance to the query specified. The tags and title of the retrieved articles are then matched with the synset generated to estimate article relevance. An article whose title or tags contain every one of the query synset parameter is deemed to be relevant to the query. Subsequently, the pruned DOM and processed content of these relevant articles are fetched from the file repository. To incorporate the context of the user's query, a transitive closure algorithm is implemented using article tags. In this algorithm, every time an article is declared as relevant, the user query synset is augmented with the article's tags. Since the article tags represent the context of the article, such an algorithm ensures that subsequent articles that are not directly related to the query synset but are related to the context of a previous article are also selected as relevant. This results in the inclusion of contextually related articles in the search results, thereby improving the overall recall of the system. The algorithm is presented in Fig. 3.

Algorithm 1 Transitive Closure

```
procedure RETRIEVEDATA(parameters,querySynset)
    articles ← query(database,parameters)
    for article ∈ articles do
        matched ← match(article,querySynset)
        if matched == true then
            querySynset ← querySynset ∪ tags(article)
        text ← text ∪ content(article)
        html ← html ∪ fetch(article,repository)
    return (text,html)
```

Fig. 3. Transitive closure algorithm for data retrieval

4) Information Extraction:

The relevant articles are processed to extract important information blocks for summarization. Sentences are segmented and considered as separate content blocks. The words in each sentence are stemmed and synsets generated, thereby ensuring that the normalized alternate forms of each word in the text are taken into consideration. TD-IDF algorithm is used to score each sentence, matching the sentence with the query synset. Typically, the TF-IDF weight is composed by two terms, as shown in Fig. 4. The first computes the normalized Term Frequency (TF), which is a measure of the document relevance of a term. Higher term frequency means that the document is more relevant to the search criteria with respect to that term. The second parameter is the Inverse Document Frequency (IDF), which is a measure of the relevance of the term as an information word. Higher IDF means that the term does not occur across documents and hence contains information unique to a limited set of documents.

$$TF = \frac{Words_{Match}}{Words_{Total}} \quad (1)$$

$$IDF = \log \frac{Documents_{Total}}{Documents_{Match}} \quad (2)$$

$$TFIDF = TF * IDF \quad (3)$$

Fig. 4. TF-IDF algorithm for information extraction

Since IDF calculation is computationally expensive, it is done offline. It is evident that a given set of articles have a fixed vocabulary. Thus the IDFs of each word in the vocabulary will remain a constant for a given document set. This

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
ISO 3297:2007 Certified
Vol. 5, Issue 6, June 2018

fact is exploited to reduce computational complexity of the system. A threshold is computed as the mean score of individual sentences, thereby preventing the bias to longer articles that a static threshold would provide. Sentences scoring higher than the threshold are made part of the article summary. Sentence scoring is done using the anaphora resolved text, and the corresponding original text sentences are included in the summary.

5) Result Generation Module:

Once information is extracted and summaries generated, they made accessible through the web interface. The user can then read summaries of articles germane to the query provided. The summaries presented thus provide condensed information across multiple sources.

IV. EXPERIMENTS AND RESULTS

The proposed system was developed using HTML/CSS/JavaScript stack for the front-end and Java for the back-end functionalities. Server integration was achieved using JSP. Separate classes were programmed for the modules mentioned before. The user is presented with a UI where preferences like search query, news source and category can be provided. When presented with the summaries of relevant articles, the user may select a particular article to view its original web page.

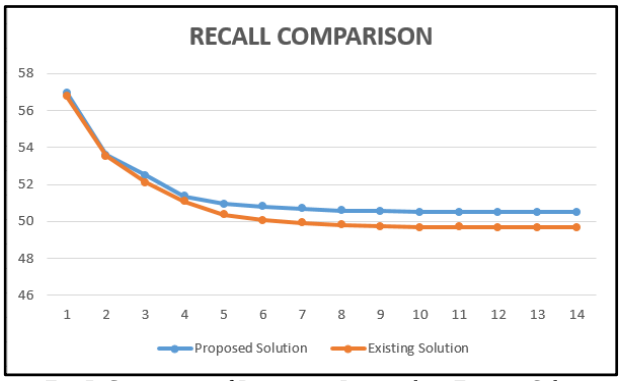

Fig. 5. Comparison of Precision – Proposed vs. Existing Solution

In order to evaluate the system, the metrics precision and recall were utilized. The proposed system was evaluated against an existing TF-IDF based information extraction algorithm, as presented in [12], which also has an offline-online architecture. Separate experiments were conducted to assess precision and recall of the systems. Additionally, an experiment was conducted to evaluate the increase in scope of query results the proposed transitive closure algorithm provides.

TABLE I.
Performance Impact on Summary Generated

Parameter	Improvement (in %)
Precision	18.18
Recall	11.71
F-Score	14.24

A. Impact on Summary Generated

Experiments testing the precision and recall of the proposed system were conducted using the DUC 2004 dataset along with a set of articles crawled from electronic news portals. The source for crawling were three online news portals: Times of India, The Hindu and Deccan Chronicle. Manual summaries were created for each of these crawled articles. 12 manual summarizers were employed to nullify any personal bias that a summarizer might have psychologically. When evaluated against the existing TF-IDF based system, it was observed that the proposed system had a better performance, scoring higher precision and recall values. The proposed system exhibited an 18.18% increase in precision, 11.71% increase in recall, and a 14.24% increase in F-Score over the existing system. The results are presented in Table I. The resolution of coreferences resulted in the increased precision while contextual searching contributed to improving the recall of the proposed system.

Additionally, the summaries generated by the proposed system were also evaluated against generic summaries. The system was evaluated against a native Python summarizer Sumy, which produces query-agnostic summaries of variable lengths. The performance was measured across various lengths of the generated baseline summary. For different lengths of summaries, the parameters precision and recall were computed and compared. It was observed

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
ISO 3297:2007 Certified
Vol. 5, Issue 6, June 2018

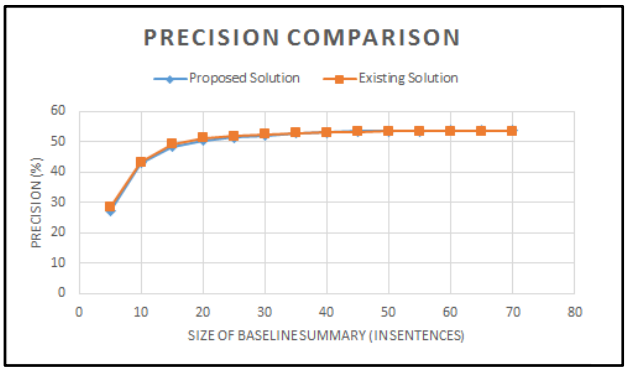

Fig. 6. Comparison of Recall – Proposed vs. Existing Solution

that although the precision was lower for small summaries, the proposed solution resulted in a marginally better precision for longer summaries. For an article summary of mean length, the proposed algorithm performed competently. A marginal improvement in precision of 0.25% was observed, as shown in Fig. 5. The recall of the proposed solution was also comparable, with an improvement of up to 4.1%, as depicted in Fig. 6. The system was evaluated against both query-driven and generic summaries and was observed to perform competently.

TABLE II
Performance Impact on Article Search

Query Type	Improvement (in %)
Generic	33
Domain-Specific	8.33

B. Impact on Article Search

To estimate the improvement in scope of results returned by the information extraction algorithm, the number of relevant articles returned was compared. Generic and domain-specific queries were evaluated separately. Generic queries like academic branches and domain-specific queries like names of specific institutions were used for evaluation. It was observed that the proposed system returned 8.33% more relevant search results for domain-specific queries while for generic queries the improvement was 33%, as illustrated in Table II. This disparity in performance improvements is due to the higher number of similar context words for a generic query. The algorithm was able to expand the context words in the query to retrieve more relevant results. Thus, the proposed system was able to gather more number of related articles that are contextually related.

V. CONCLUSION

In this paper, a novel news extraction and summarization technique was presented. The system operates in an online and offline phase, separating the pre-processing and query processing processes. The offline phase collects content from multiple sources and processes the data, resolving textual coreferences and tagging the article with the context of information that it contains. The online phase consumes this processed data, extracting only relevant text and presents it as a summary to the users. Several experiments were conducted to evaluate the efficacy of the proposed system. In contrast to existing solutions, the proposed system performs anaphora resolution to resolve coreferences in the text and contains article tagging cum transitive closure algorithm for information extraction with improved accuracy. While anaphora resolution resulted in enhanced precision and recall of the summaries produced, the transitive closure algorithm was able to excerpt contextually related articles from the data store, in addition to the lexically related articles that existing solutions provide.Though the proposed system improves upon the performance of existing solutions, there is scope for further enhancement. To improve the relevance of the results produced, ranking algorithms may be employed as part of the search algorithm. Giving articles scores based on relevance and ranking them will enable more relevant results to be presented to the user in preference to other results. Furthermore, techniques like LSA (Latent Semantic Analysis) can be employed as an alternative to the bag-of-words approach used in this system. While the proposed solution improvises on the existing bag of words approach by incorporating anaphora resolution, semantic relation between words are not considered by the underlying summarization algorithm. By employing such techniques, the system will be able to provide better summaries. The system delivers extractive summaries that are terse but incoherent nonetheless in nature. Such a semantic analysis will enable creation of abstractive summaries that both convey the gist of the article and yet is linguistically coherent.

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
ISO 3297:2007 Certified
Vol. 5, Issue 6, June 2018

REFERENCES

[1] Raymond J. Mooney and Razvan Bunescu, "Mining Knowledge from Text Using Information Extraction," in ACM SIGKDD Explorations Newsletter - Natural language processing and text mining, Volume 7 Issue 1, June 2005, Pages 3-10.
[2] Jie Tang, Mingcai Hong, Duo Zhang, Bangyong Liang, and Juanzi Li, "Information Extraction: Methodologies and Applications," in Emerging Technologies of Text Mining: Techniques and Applications, Chapter 1, Pages 1-33.
[3] Sunita Sarawagi, "Information Extraction," in Foundations and Trends in Databases, Vol. 1 No. 3, 2007, Pages 261–377.
[4] Lawrence Wong, "ANSES Automatic News Summarization and Extraction System".
[5] Karel Jezek and Josef Steinberger, "Automatic Text Summarization (The state of the art 2007 and new challenges)," Proceedings of Znalosti, 2008, Pages 1-12.
[6] Vishal Gupta and Gurpreet Singh Lehal, "A Survey of Text Summarization Extractive Techniques," in Journal of Emerging Technology in Web Intelligence, Vol. 2, No. 3, 2010, Pages 258–268.
[7] Ani Nenkova and Kathleen McKeown, "Automatic Summarization," in Foundations and Trends in Information Retrieval, Vol. 5, Nos. 2–3, 2011, Pages 103-233.
[8] Kam-Fai Wong, Mingli Wu and Wenjie Li, "Extractive Summarization Using Supervised and Semi-supervised Learning," Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), 2008, Pages 985-992.
[9] Dipti.D.Pawar, M.S.Bewoor and S.H.Patil, "Text Rank: A Novel Concept for Extraction Based Text Summarization," in International Journal of Computer Science and Information Technologies, Vol. 5 (3), 2014, Pages 3301-3304.
[10] Akiko Aizawa, "An information-theoretic perspective of tf–idf measures," in Information Processing and Management 39, 2003, Pages 45-65.
[11] Wen Zhang, Taketoshi Yoshida and Xijin Tang, "A comparative study of TF*IDF, LSI and multi-words for text classification," in Expert Systems with Applications 38, 2011, Pages 2758–2765.
[12] Monisha Kanakaraj, Sowmya Kamath S, "NLP based Intelligent News Search Engine using Information Extraction from e-Newspapers," in IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), 2014.
[13] Gautam Pant, Padmini Srinivasan and Filippo Menczer, "Crawling the Web," in Web Dynamics: Adapting to Change in Content, Size, Topology and Use. Edited by M. Levene and A. Poulovassilis, 2004, Pages 153-178.
[14] Sifatullah Siddiqi and Aditi Sharan, "Keyword and Keyphrase Extraction Techniques: A Literature Review," in International Journal of Computer Applications (0975 – 8887), Volume 109 – No. 2, 2015, Pages 18-23.
[15] Rada Mihalcea and Paul Tarau, "TextRank: Bringing Order into Texts," in Proceedings of EMNLP, 2004, Pages 404-411.
[16] Stuart Rose, Dave Engel, Nick Cramer and Wendy Cowley, "Automatic keyword extraction from individual documents," in Text Mining: Applications and Theory, 2010, Pages 1-20.
[17] Imran Q. Sayed, "Issues in Anaphora Resolution," 2003.
[18] Shalom Lappin and Herbert J. Leass, "An Algorithm for Pronominal Anaphora Resolution," in Journal of Computational Linguistics, Volume 20 Issue 4, 1994, Pages 535-561.
[19] Christopher D. Manning, Mihai Surdeanu et al., "The Stanford CoreNLP Natural Language Processing Toolkit," in Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics, 2014.
[20] George A. Miller, Richard Beckwith et al., "Introduction to WordNet: An On-line Lexical Database," in International Journal of Lexicography 3(4), 1991, Pages 235-244