Query-Based Extractive Text Summarization Using Sense-Oriented Semantic Relatedness Measure

Nazreena Rahman (✉ nazreena.rehman@gmail.com)

Kaziranga University <https://orcid.org/0000-0001-7188-0111>

Bhogeswar Borah

Tezpur University School of Engineering

Research Article

Keywords: Query-based extractive text summarization, Sense-oriented semantic relatedness measure, Word Sense Disambiguation (WSD) technique, Clusters of query-relevant sentences, Redundancy removal technique, Senseval and SemEval datasets, Document Understanding Conference (DUC), ROUGE (Recall-Oriented Understudy for Gisting Evaluation)

Posted Date: November 30th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-1102477/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Query-Based Extractive Text Summarization Using Sense-Oriented Semantic Relatedness Measure

Nazreena Rahman^{1*} · Bhogeswar Borah²

the date of receipt and acceptance should be inserted later

Abstract This paper presents a query-based extractive text summarization method by using sense-oriented semantic relatedness measure. We have proposed a Word Sense Disambiguation (WSD) technique to find the exact sense of a word present in the sentence. It helps in extracting query relevance sentences while calculating the sense-oriented sentence semantic relatedness score between the query and input text sentence. The proposed method uses five unique features to make clusters of query-relevant sentences. A redundancy removal technique is also put forward to eliminate redundant sentences. We have evaluated our proposed WSD technique with other existing methods by using Senseval and SemEval datasets. Experimental evaluation and discussion signifies the better performance of proposed WSD method over current systems in terms of F-score. We compare our proposed query-based extractive text summarization method with other methods participated in Document Understanding Conference (DUC) and as well as with current methods. Evaluation and comparison state that the proposed query-based extractive

text summarization method outperforms many existing methods. As an unsupervised learning algorithm, we obtained highest ROUGE (Recall-Oriented Understudy for Gisting Evaluation) score for all three DUC 2005, 2006 and 2007 datasets. Our proposed method is also quite comparable with other supervised learning based algorithms. We also observe that our query-based extractive text summarization method can recognize query relevance sentences which meet the query need.

Keywords Query-based extractive text summarization · Sense-oriented semantic relatedness measure · Word Sense Disambiguation (WSD) technique · Clusters of query-relevant sentences · Redundancy removal technique · Senseval and SemEval datasets · Document Understanding Conference (DUC) · ROUGE (Recall-Oriented Understudy for Gisting Evaluation)

1 Introduction

With the growing textual information from the Internet or other sources, extracting useful information in minimum time is a tough job. Text summarization solves the problem of overloading of information. According to Hovy and Lin (1998), text summarization is a process of finding short, accurate, and fluent form of a single or multiple text documents where the length of the summary is not more than half of the original texts. Summarization techniques can be either extractive or abstractive. Extractive summarization techniques extract major snippets, sentences or passages from the input text documents, while the abstractive summarization technique rephrases the information content present in text documents. Query-based text summarization helps in extracting the required information needed by the

¹ Nazreena Rahman
Assistant Professor
Department of Computer Science and Engineering
Kaziranga University
Jorhat
Assam, India
Tel.: +09706148321
E-mail: nazeena@kazirangauniversity.in

² Bhogeswar Borah
Professor
Department of Computer Science and Engineering
Tezpur University
Tezpur
Assam, India
Tel.: +09435490984
Fax: +123-45-678910
E-mail: bgb@tezu.ernet.in

user. Sources of the text summary either can be single text document or multiple text documents. This query-specific text summary is constructed based on the calculation of the relevancy between sentences in the text documents and the given query. Hence, this type of summarization should be concise and should satisfy the user's need. Three major advantages of automatic query-based text summary are:

- User gets the specific data with minimal loss of information without having to read large volume of text data.
- User gets the information in concise fashion.
- It reduces the reading time.

While doing the query-based text summarization, most of the summarization systems depend on surface features like TF-IDF cosine similarity between input text sentences and a query (Wu et al. (2008), Park et al. (2006), Rastogi (2020), Ramos et al. (2003)). TF-IDF cosine similarity measures the query relevance. However, relevance and similarity both are not same. For example: we consider the document cluster 'd307b' present in DUC 2005. The query is: "What hydroelectric projects are planned or in progress and what problems are associated with them?" Here, hydroelectric projects are the keywords having higher TF-IDF scores. But main purpose of the query is to find out planned, progress and problems associated with hydroelectric projects. These general words are not present in the source text document or may be their frequency of occurrence is quite low. It is obvious that if TF-IDF cosine similarity measure is applied, top scoring sentences will contain mainly the words "hydroelectric" and "projects". Unfortunately, none of them gives the exact planned, progress and problems associated with hydroelectric projects. Hence, TF-IDF measure is not sufficient to measure the query relevance which in turn gives information less query-oriented summary. It means that only the presence of query words in input sentences are not enough to make an informative-rich summary (Han et al. (2000), Zhao et al. (2009), Carpineto and Romano (2012), Afuan et al. (2019), Strzalkowski et al. (1998)). But, indeed, there are some underlying relationships with the query words which are semantically related with the query. Our aim is to find semantically related sentences which will help in getting the summary in a gist way with relevant information.

Before doing the summarization, we need to find query relevant sentences. To find the relatedness score between query and input text sentence, sense-oriented semantic relatedness measure is used. We know that English language have many ambiguous words. A word carries more than one sense and can be used in many ways. For example: *key*. Sometimes we use key as a

small piece of shaped metal or sometimes it is used to describe something which is crucial important. The actual sense of the word could also be calculated by generating contextualized word embeddings from language models such as ELMO, BERT, GPT, etc (Kutuzov and Kuzmenko (2019), Devlin et al. (2018), Radford et al. (2019), Wiedemann et al. (2019), Hadiwinoto et al. (2019)). But sizes of these models are humongous. In contrast, real world applications need small model size, low response times and low computational power wattage (Gupta and Agrawal (2020)). BERT and other neural network models are very compute-intensive at inference time.

Ambiguous word always degrades the quality of query-based text summarization. Ambiguity in words minimizes the query relevance for query-oriented text summarization. Ambiguity removal or detecting senses is an emerging problem in natural language processing and ontology (Yadav et al. (2021), Zamanifar et al. (2008), Abu Nada et al. (2020), Maulud et al. (2021)). Sense makes an important role while finding semantic relatedness score between two content words. To find query relevant sentences, we find sense-oriented sentence semantic relatedness score between query and input text sentence. This relatedness score includes sense wise semantic relatedness score, sense relatedness score and word order similarity score. Therefore, a word sense disambiguation method is proposed with less time complexity. We consider accurate sense of the words while finding relatedness score between two words. It helps in increasing the accuracy of relatedness score. After extracting the query relevant sentences, we use cluster based redundancy removal method to create query-based text summarization. This redundancy removal technique increases the information diversity and reduces the summary size.

Query-based text summarization can produce concise information according to the users queries from the given texts (Wang et al. (2010)). Most of the queries are complex real-world questions related to the input text documents (Canhasi and Kononenko (2014)). In spite of its relevance and significance of query-based text summary, the extracted sentences for query-based text summary has not been fully explored (Bayatmakou et al. (2021)). Our main objective is to create a redundancy free query-based text summary. Existing query-based text summarization methods use many different features. It is also seen that most of the existing methods use to give more focus on query-independent features and usually less consideration is given to the query-dependent feature (Damova and Koychev (2010), Afsharizadeh et al. (2018)). Those works do not give much importance to select the features according to the re-

quirement. Many features are irrelevant and redundant according to the requirement (Allahyari et al. (2017)). Keeping in view, to eliminate irrelevant features, proposed method extracts the query relevant sentences. Use of relevant features restrict doing unnecessary computation and enhances the performance (Balachandran et al. (2018)). We use only the relevant and redundancy free sentence-based features. Further we show that use of relevant features help in making clusters. Each cluster contains similar sentences. The probability of redundant sentences present in each cluster is very high. Based on its sense, existing redundancy removal methods fails to find the redundant sentences (Oufaida et al. (2014)). Therefore, we need to develop a redundancy removal method particularly for multiple text query-based text summarization. In this summary, more than one sentences may be present in the prepared summary and they convey the same meaning. Through redundancy removal method, only one sentence will be retained and others will be deleted. It helps in minimizing any redundant or repetitive content. Finally, we will get an informative query-based summary with minimum redundancy.

We need to calculate semantic relatedness measure to find the query relevant sentences for query-based text summary. One important key factor of ontology based semantic relatedness measure between two sentences is that it calculates the score on the basis of words. Existing semantic relatedness measures take the first sense of both words by default and give score or sometimes it uses that sense for which the score is highest (Meng et al. (2013)). It does not always give the correct score based on its actual meaning. Before calculating the measure, it is important to find the correct sense of each word present in the sentence. Therefore, we provide a word sense disambiguation method to find the correct sense. We match each sense of a word in word level which further helps in finding accurate sense of the complete sentence.

Query-based text summarization computes semantic relatedness measure between query and input text sentences. It helps in extracting more query relevant sentences. Existing query-based text summarization techniques encounter performance bottlenecks when extracting query relevant sentences (Afsharizadeh et al. (2018), Widyassari et al. (2020)). Extraction of semantically related sentence is a difficult task as machine does not possess a vast knowledge like a human being. We introduce a word-based sense-oriented sentence semantic relatedness measure to find query relevant sentences on the basis of word's sense. This relatedness measure finds the score between input text sentences and query.

In real-world applications, there is a high probability that all the query words may not be available in the input text documents. For example: let us take following query from DUC 2005 query-based text summarization dataset: *What hydroelectric projects are planned or in progress and what problems are associated with them?* Many related text documents are there, but in some text documents *hydroelectric* word is not present. Considering that issue, our method can find the semantically related sentences from them. In those text sentences a word *dam* is present which is related with *hydroelectric* word. Existing methods have not considered to find sense wise query related sentences (Chali and Joty (2008)). Our method has additional advantage of extracting query relevant sentences in case of absence of query words in the text documents. This makes our proposed sentence semantic relatedness measure a different one from the existing measures. The method uses word sense to identify how much these two words are semantically related. Our proposed method can extract the sense of a content word and find its meaning. The method expands each content word sense-wise and finds its relatedness with other words. Finally, word level relatedness score helps in getting sentence level relatedness score. It helps in extracting more semantically related sentences although same words are not present in the sentences. Extracting only query relevant sentences also help in eliminating irrelevant features.

In this paper, we introduce an extractive type redundancy free multi-text document query-based text summarization method. This proposed method investigates mainly following issues : (1) develop a content word sense disambiguation method (2) find query relevance sentences (3) create a cluster-based redundancy free summary so that final summary comes with more information diversity.

The specific contributions are as follows: (1) This query-based text summarization technique extracts sentences based on its sense. To the best of our knowledge, we are the first to use exact sense which will be suitable for the context of sentence. We have done the sense-tagging of our input text sentences. It helps in getting more query relevant sentences. (2) We use sense-oriented semantic relatedness score between query and input text sentence to extract query-based text sentences. This total relatedness score combines semantic relatedness score, word sense score and word order score. From sense-oriented semantic relatedness score, we find query relevance sentences. (3) We provide a redundancy removal technique which is suitable for finding redundant sentences in multiple text document query-based text summarization. The proposed method uses the correct sense while finding the redundancy between

two sentences. It helps in minimizing any redundant or repetitive content. In final stage of the summary, best sentences are selected having most important overlap with query and lowest redundancy among selected sentences. At the end, we get a query-based text summary with relevant information diversity. Experiments are performed on publicly available DUC datasets. Evaluation results show better than baseline systems and many state-of-the-art systems.

The remainder of the paper is as follows: Section 2 presents the literature review on different techniques used in query-based text summarization. Section 3 provides the brief introduction of the proposed technique. Section 4 explains the importance of sense for finding semantic relatedness measure. Section 5 describes the proposed unsupervised method for WSD. Section 6 gives the detailed description of the sense-oriented semantic relatedness calculation between two sentences. Section 7 provides how to generate the redundancy free query-based text summary. Section 8 presents the evaluation of WSD technique. Section 9 and 10 give the dataset description and the evaluation metric for query-based text summarization. Section 11 and 12 include the parameter setting and the performance comparison for query-based text summarization method. Section 13 discusses the query relevancy. Finally, the paper concludes in section 14 with some feasible future scopes.

2 Literature Review on Query-Based Text Summarization Techniques

Different classical, machine learning and neural network based techniques are used extensively in query-based text summarization from many years. Three different techniques will be reviewed separately in respective subsections.

2.1 Classical Techniques Applied in Query-Based Text Summarization

Luhn (1958) gives the first attempt for generating generic summary of a text by finding the frequency of the word present in a text. Before finding the frequency of words in a text document, a few steps like stop word removal, stemming were performed in his work. This work is further known as universal pre-processing step in text analytics. Baxendale (Baxendale (1958)) uses position features while creating the summary of a text. Edmundson (Edmundson (1969)) uses both frequency of words and positional features along with cue words and skeleton of structure of text to create summary. Different weights are assigned with different features

to give scores to each sentence. This method got 44% accuracy while finding similarity between system generated summary and human generated summary. In case of query based text summarization, relevance score (Berger and Mittal (2000)) is used to find the similarity between the query and input text sentences. Term frequency (tf) and inverse document frequency(idf) was used here to calculate the relevance score. The weighted tf and idf is also used to find query based text summary (Ulrich et al. (2008)). Different graph based approaches were used by many researchers. Mohamed and Rajasekaran (2006) have used a document graph where two types of relations “is a” and “related.to” are used. This document graph is used to represent the text document. They made three attempts in where, they created the centric graph with a little modification for generic summary. After that, they extended their work to query-based summary and finally they used extra information of query by introducing query modification technique. However the performance of their method is better than many baseline systems, but it fails to work when different sub topics are hidden in the query. However, Semantic relatedness is not also considered to find out related words with the query words. Ye and Wei (2008) combine two types of score: first score is based on word overlap between query and the sentence and second score is based on relation between sentences. First measure is calculated by finding the semantic similarity between the sentence and query and second measure is carried out by using the semantic graph. These two measures give high-quality relevant and high density based sentences for creating summary. Their method performs well only on single document query-based text summarization.

TextRank, LexRank are graph based ranking algorithms. They are widely used in text processing specially in text summarization (Mihalcea and Tarau (2004), Erkan and Radev (2004), Akhtar et al. (2019)). TextRank algorithm uses number of common words present in two sentences whereas LexRank algorithm is based on cosine similarity of TF-IDF vectors. TextRank algorithm relies on the Google’s PageRank algorithm (Brin and Page (1998)). One of the major drawback of TextRank algorithm is that it can only work for single document summarization. TextRank algorithm also neglects about the information about text structure and context (Yu et al. (2016)). In case of LexRank algorithm, the performance is not always constant (Patil and Brazdil (2007)). Semantic relation depends on cosine similarity of TF-IDF vectors. Hence, similarity computation

can be improved by incorporating more advanced techniques (Erkan and Radev (2004)).

2.2 Machine Learning Techniques Applied in Query-Based Text Summarization

Many supervised and unsupervised algorithms have been proposed by many researchers. Different machine learning approaches like Support Vector Machines (SVM), Support Vector Regression (SVR), Bayesian models, Hidden Markov Models (HMM) and Decision Trees are used extensively in query-based text summarization purpose. HMM model is used as a statistical model for text summarization. The machine learning techniques are mainly based on information retrieval methods. Ouyang et al. (2011) have used a regression style learning model to rank the sentences for query-based text summarization. Here, SVR is used as learning model. The technique is based on feature-based extractive framework. Extraction of sentences are done based on pre-defined features and a composite scoring function. Schilder and Kondadadi (2008) have used SVM model to rank the sentences based on two defined features. One feature is word-based and another is sentence-based. They present a fast query-based text summarization technique known as FastSum. FastSum is based on frequency of words of clusters, documents and topics. Word based feature is calculated based on cluster, document, topic title and description. Sentence based feature is computed depending on length and sentence position. Though both mentioned methods performs better than many baseline systems, but in future they can add more features based on structure of documents and relations between sentences. Peng et al. (2016) have used query-based text summarization technique while doing query reply in social networking sites. Summarization technique based on features like content weight of the sentences, location of the sentence, cosine-similarity between query and input sentence. Their technique also gives more attention in redundancy and noise removal in text messages as social network messages are highly unstructured. They can further enhance their method by incorporating text semantic analysis as features. Canhasi et al. (Canhasi and Kononenko (2014)) have applied a different matrix factorization method named as weighted Archetypal Analysis (wAA) for query-based text summarization purpose. In fact they have formalized the query-based text summarization problem as a weighted archetypal analysis problem. This work also presented how to add query information in Archetypal Analysis and how to add weighted Archetypal Analysis in sentence ranking and clustering. However, the performance of wAA can be improved in following ways (1) they can add semantic similarity by using lexical based ontol-

ogy, (2) expansion of query, (3) not only restricted to sentence to sentence relations and sentence to terms relations but also should consider n-grams, phrases and semantic role arguments levels.

2.3 Neural Network based Techniques Applied in Query-Based Text Summarization

Neural networks process information like a human brain does. The building block of neural network is the neuron. Artificial neural networks are computing systems inspired by the biological neural networks (Anderson (1995)). It is a composition of perceptrons connected in different ways and operating on different activation functions. A neural network with only one hidden layer is known as shallow neural network and more than one hidden layer is known as deep neural network. Deep neural network based techniques are more efficient than traditional machine learning techniques as deep learning skips the manual steps of extracting features. Svore et al. Svore et al. (2007) have proposed a neural network based single document text summarization system called NetSum. This algorithm is capable of doing text summarization using neural network and finding novel features using news search query logs and wikipedia. This technique is influenced by the method of Burges et al. (2005) as similar ranking algorithm is used to rank the sentences. In future, feature selection technique can be added for better performance of NetSum and also should consider extracting content across sentence boundaries. On the basis of the semantic networks, fuzzy logic and evolutionary programming, Shardan and Kulkarni (2010) put forward three approaches and compared with many baseline summarization systems. Semantic network is a knowledge base that represents the semantic relations between different concepts (Bird et al. (2009)). First approach carries part of speech tagging, semantic and pragmatic analysis and cohesion. Second approach performs summarization using a decision module based on fuzzy concepts. Third system takes the consideration of all three evolutionary, fuzzy and connectionist techniques. The limitation of their methods is that they consider only the small details related to small summarization process but not in entire summarization system. Prasad et al. (2009) have applied Recurrent Neural Networks (RNNs) in text summarization .

The deep learning techniques have been successfully applied in many fields like image, vision, speech and natural language processing (NLP) tasks. It is mainly used in optimum feature selection process. In fact, currently use of deep learning in NLP is one of the most promising field of research. Collobert et al. put forward a semi-supervised method for part of speech tagging,

chunking, named entity recognition, and semantic role labeling purpose. Application of deep learning techniques in text summarization have shown quite satisfactory results. In the works of Duraiswamy (2014), Verma and Nidhi (2017), their techniques use Restricted Boltzmann Machine (RBM) as an unsupervised approach where first two approaches used sentence level features and last approach used tf representations. In future, we can extract more features according to the user's requirement and also more hidden layers can be added in RBM. Yousefi-Azar and Hamey (2017) have used deep Auto-Encoder (AE) to calculate the optimum features based on tf representation of input text document. AE is a feed-forward neural network. AE is used as unsupervised feature learning technique. They propose an Ensemble Noisy Auto-Encoder (ENAE) where this ENAE adds noise to the input text documents and extracts top sentences from an ensemble of noisy runs. This method however is not implemented on newswire summarization. Cao et al. (2015) have used recurrent neural network to rank the sentences for multiple text document summarization. They have taken sentence ranking task as a hierarchical regression process. Ranking of words as well as sentences give more informative and redundancy free summary. This supervised technique can deal with variable length input for text summarization. It calculates the importance of text document sentences as well as phrases. However, their method is only restricted to generic text summarization. Zhong et al. (2015) use unsupervised deep learning technique in creating summary from multiple text documents. A query-oriented extracted sentence technique is proposed to extract the meaningful sentences as per the query requirements. In reconstruction validation phase, the whole deep architecture is adjusted by minimizing the loss of information. Finally, to maximize the importance of a sentence for including in final summary, dynamic programming is used. On the basis of the performance comparison of DUC 2007 datasets, the performance of this method is better than many baseline systems. However, the recall measure is much low than ranking SVM classifier. Afsharizadeh et al. show that use of suitable features can enhance the summary quality. They have used 11 important and best features to extract the sentences for summarization (Afsharizadeh et al. (2018)). The main drawback of their approach is that the lack of semantic similarity computation. For summarization of scientific documents, an interactive query-based approach is proposed by Bayatmakou et al. (Bayatmakou et al. (2021)). Initially, query is refined by user selected keywords or keyphrases. In the second step, system extracts the candidate sentences based on the keywords or keyphrases selected by the user in first step. Sys-

tem also expands the sentences by using Genetic Algorithm (GA) (Zhao and Tang (2010)). Finally, Maximal Marginal Relevance (MMR) algorithm is used for ranking the sentences (Carbonell and Goldstein (1998)). Although proposed method gets a higher degree of user satisfaction but also it has high computational time and cost complexity. Murarka and Singhal (2020) use a hybrid approach for query relevant single document text summarization. They have used Latent Semantic Analysis (LSA) (Yeh et al. (2005)) technique to make an intuitive semantic structure. They have also used PageRank algorithm to remove the redundancy (Thakkar et al. (2010)). Their method works well in comparison to many graph-based and semantic-based methods. In future, a semi supervised model can be developed to increase the semantic efficiency of the summary.

Now-a-days, transformer (Vaswani et al. (2017)) based language models like Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al. (2018), Miller (2019), Srikanth et al. (2020)), Generative Pre-preparing Transformer (GPT-2) (Radford et al. (2019)), Multi-task Deep Neural Network (MT-DNN) (Liu et al. (2019)), Extra-Long Network (XLNet) (Yang et al. (2019)), Text-to-message move transformer (T5) (Raffel et al. (2019)), T-NLG (Rosset (2020)) and GShard (Lepikhin et al. (2020)) are widely used in text summarization. Risne and Sitova have proposed two summarization approaches: Extractive and Abstractive. In their method, extractive summarization uses submodular functions and the language representation model BERT and abstractive summary uses the language model GPT-2 (RISNE and SIITOVA (2019)). Lamsiyah et al. use order and semantic relation based BERT sentence embedding model (Lamsiyah et al. (2021)). This unsupervised extractive text summarization method uses transfer learning from BERT. However, the high energy consumption and large inference latency are the most serious bottlenecks of transformer based language model (Strubell et al. (2019)).

Recently neural network based models are used widely in abstractive text summarization. Bahdanau et al. (2014) have used neural network for machine translation. Rush et al. (2015) have proposed a fully data-driven abstractive approach by using an attention based feed-forward neural network language model (NNLM). Developing on their work, In their follow-up work, Nallapati et al. (2016a) put forward a set of attentional Encoder Decoder Recurrent Neural Networks. Later on, they extend this work by giving focus on different critical text summarization issues such as modeling key-words, capturing the hierarchy of sentence-to-word structure, and emitting words that are rare or unseen at training time

(Nallapati et al. (2016b)). To do extensive experiments and comparison with other summarization systems, they also propose a novel dataset. Lopyrev (2015) uses an encoder-decoder recurrent neural network with LSTM units and attention to generate abstractive based news article headline. Filippova et al. (2015) use word embeddings and Long Short Term Memory (LSTM) models to obtain a readable and informative rich compression.

In spite of increasing rate of abstractive summarization techniques, extractive summarization techniques are still attractive as they are less complex and inexpensive. In fact, abstractive summarization needs good command of the subject and natural language which both can be difficult task for a machine. Extractive technique provides grammatically and semantically correct summaries in most cases. Cheng and Lapata (2016) have proposed a data-driven approach based on neural networks and continuous sentence features. They proposed a framework for single-document summarizer based on general hierarchical document encoder and an attention-based extractor. Nallapati et al. (2017) have used a recurrent neural network based sequence model for extractive summarization of documents. Their proposed interpretable neural sequence model allows intuitive visualization. Both these deep learning based supervised methods (Cheng et al. Cheng and Lapata (2016) and Nallapati et al. Nallapati et al. (2017)) suffer from domain adaptation issues when they tested on different corpuses. A query-focused text summarization system is developed by Cao et al. (2016) in which both query relevant ranking and sentence saliency ranking tasks are tackled jointly. System is composed of CNN (Convolutional Neural Networks), Pooling and Ranking layers. It is clear from the above mentioned works that though they try to find query-based text summarization but less importance is given to query dependent features. Without these query-dependent features, their performance variation is quite negligible. Hence, these methods fail to find query relevant sentences.

2.4 Literature Review on Redundancy Removal in Query-Based Text Summarization

This paper also gives focus on redundancy removal methods. We use it for query-based multi-document text summarization system as probability of redundant information is much more in there. The most important part of removal of redundancy is to find *similarity measure*. In case of redundancy removal we have to find out how dissimilar two sentences are. There are numerous ways to find similarity between two sentences. Here, we only consider redundancy removal techniques particu-

larly for query-based multi-document text summarization. Abdi et al. (2017) have used a redundancy removal technique at summary generation step. According to their method, sentences are ranked using a graph-based ranking model and sentences are selected having highest score values until it get the required summary length. Summary contents multiple extracted sentences from several contents. Since, various sentences may have similar content, it is required to reduce redundancy. To tackle this problem, two main levels are employed: first one is scoring level and second one is comparison level. Before adding a sentence to the final summary, it is compared with significant sentences which are already in final summary. If the sentence is not too similar with other significant sentences, then only it is allowed to be in final summary. They use a greedy algorithm (Wan et al. (2007), Zhang et al. (2005)) to impose a diversity penalty on the sentences. Lloret and Palomar (2013) have used three different levels of language analysis to tackle redundancy in text summarization. Different levels of language analysis are: lexical, syntactic and semantic. Cosine similarity is used in lexical-based redundancy detection method. Textual entailment is used in syntactic-based redundancy detection method and sentence alignment is used in semantic-based redundancy detection method. Result shows that better summary when employing syntactic or semantic based redundancy removal approaches. Mei and Chen (2012) have proposed a soft clustering method for query-based text summarization. A fuzzy medoid-based clustering method is used to create subsets of sentences where each of the subset corresponds to a sub topic of the related topic. Sub topic based feature captures the relevance of each sentence within different sub topics. Finally, it increases the chance of creating the summary with wide coverage and less redundancy. Ye and Wei (2008) have introduced a statistical model for query-oriented summary generation where Maximum Marginal Relevance (MMR) is used to reduce the redundancy. MMR strategy is used by many researchers in finding redundancy in query based text summarization. Ouyang et al. (2011) and Li and Li (2013) also use MMR technique to minimize redundancy. Peng et al. (2016) use Simpson distance to find how similar two sentences are. From this survey, it is found that though many different techniques are applied in query-based text summarization, still sense based semantic relatedness measure is not used yet to find the query-based summary.

3 Proposed Redundancy Free Query-Based Extractive Text Summarization Method

The steps to create query-based text summary are described briefly here.

- Pre-processing: To reduce computational time, pre-processing is done on text sentences by applying various techniques proposed by linguists. Pre-processing removes unwanted words from text document and makes it a lighter one. Following techniques are applied to do the pre-processing of text document:

Parts of tagging: To classify the words on the basis of parts of speech category, parts of speech tagging (Bird et al. (2009)) is done. Parts of speech tagging classifies the content words. Content words include noun, adjective, verb and adverb.

Named Entity Tagging: To distinguish different Names person, location or organization wise, we do the named entity tagging Bird et al. (2009). We will not consider person's name to find semantic relatedness as it is not present in Lexical resources.

Stop word removal: It is better to filter out *a, an, the, in* type of words which do not give any semantic meaning to the sentence. This is known as stop word removal in text mining applications. Here, we use stop word list stores in Natural Language Tool Kit (NLTK) in python.

Stemming: Finally, stemming is done to the content words. Stemming brings the word to its root or base form. For example to convert a word from plural to singular root form (girls to girl) or removing *ing* from a verb (singing to sing). Number of algorithms are available to do the stemming in natural language processing.

- Content word disambiguation: To find out the sense of a word present in the query and input text documents, we take the collocation-based score. Word sense disambiguation is done to improve the accuracy of the sense-oriented sentence semantic relatedness score between query and input text sentence. In section 5 the application of collocation-based feature for finding sense of a word is described.
- Calculating sense-oriented sentence semantic relatedness score between query and input text sentence: To get the query relevant sentences, sense-oriented sentence semantic relatedness score is calculated between query and input text sentence. Here, we are taking 60% as threshold value. We assume that sentences having higher than or equal to the threshold value are all equally important for creating query-based text summary. In section 6, description of the proposed sense-oriented sentence semantic relatedness score calculation is presented.
- Redundancy free query-based text summary generation: On the basis of distinct features, clusters are created. Each cluster contains the query relevant sentences. There is a high probability of presence of redundant sentences in each cluster. Therefore,

a redundancy free sentence extraction method is proposed to create query-based text summary. The method to find redundancy free sentences is already defined in section 7.

The overall process for creating redundancy free query-based text summary is shown in the Algorithm 1.

Data: Query (Q) and Input Text Documents (T)

Result: Query-Based Text Summary

- 1 Do the pre-processing of input text documents and query described in section 3
- 2 Do the word sense disambiguation of each content word by using the Algorithm 2
- 3 Find out the query relevance sentences by using the Algorithm 5
- 4 Create the redundancy free query-based extractive text summary by using Algorithm 6 and 7

Algorithm 1: Steps to Create a Redundancy Free Query-Based Extractive Text Summary

4 Importance of Sense for Finding Semantic Relatedness Measure

Query based text summarizer extracts semantically query related sentences from input text sentences. In most cases, to find the semantic relatedness score between two words using WordNet, existing measures find the score with all the senses and give maximum score.

WordNet is the most commonly used English lexical database used as a resource for English sense relations created by Miller (Charles (1988)). WordNet is a lexical dictionary which is used to find semantic similarity/relatedness score between two content words. A distinct difference is drawn by Resnik (1995) and Budanitsky and Hirst (2001) between semantic similarity and semantic relatedness. Similarity states near synonyms or can say as roughly substitutable in context. Relatedness implies larger set of potential relationship between words. Antonyms are highly related words but they less similar. *Human* and *women* are not similar but highly related than *human* and *car* while *girl* and *women* have similarity. Thus *Similarity* is a sub case of *Relatedness*. We prefer to use semantic relatedness score to find query relevance sentences. Semantic similarity is based on taxonomic relations (based on their meaning, it includes "is a" relations) whereas semantic relatedness is a more general concept. Semantic relatedness score between two content words can be found using WordNet. Miller started the use of WordNet in 1985. This English data-base is designed by Cognitive Science Laboratory of Princeton University. WordNet consists of three separate databases: one is for each Nouns and

Verbs and another is for adverbs and adjectives. Number of content words listed in WordNet 3.0 is shown in following Table 1:

Table 1: WordNet 3.0 Description

Content Word	Number
Noun	117,097
Verb	11,488
Adverd	4601
Adjective	22,141

In English language, most of the words have more than one senses. In WordNet, concepts are represented by the word, its part of speech and its sense number. For example: for the word *bank*, the concept is *bank#n#1*. It means the word *bank* is a noun here and it has the first sense. Concepts in WordNet are linked together in a hierarchical structure. In Fig. 1 an example of words present in WordNet is shown.

In WordNet, for different types of part of speech of a content word, we get different senses. Senses are the gloss or the definition. For a content word, if it has more than one sense then different number senses have different glosses. A content word may contain different senses for a same part of speech. Table 2 says about different gloss of word *interest* present in WordNet. Each synset of *interest* contains its parts-of-speech and sense number. Gloss implies a dictionary-style definition.

Ambiguous word always degrades the quality of information retrieval. Ambiguity in words minimizes the query relevance for query-oriented text summarization. According to Jurafsky and Martin (2014), the task of selection of correct sense for a word is known as word sense disambiguation. Ambiguity removal is an emerging problem in natural language processing and ontology. Sense makes an important role while finding semantic relatedness score between two content words. Two words can be said as semantically related/similar if there exist any relation between the words that are described in WordNet. Following Table 3 relations are present in WordNet to find semantic relatedness as well as similarity between content words. WordNet takes only the content words (noun, verb, adverb and adjective) as they carry vital information in a sentence.

For example: we take two sentences: (1) Ram went to the bank to deposit money and (2) Ram went to the bank of river Brahmaputra. Here, we find the semantic relatedness score between the content words of two sentences. In this example: we take two words that are *bank* and *river* where: *bank* word comes from the first

sentence and *river* word comes from the second sentence. Both *bank* and *river* words are noun here. While finding the semantic similarity score, we have to give the word, then it's part of speech and sense number. When we do not give any particular sense as an input, WordNet takes automatically that sense for which it gets the highest semantic similarity score. Using WordNet lexical dictionary, we get the following semantic relatedness scores for different measures listed in Table 4.

Table 4 shows that by default almost all semantic relatedness measures take the first sense of *bank* as it gives maximum score with *river*. Table 5 provides the different senses present for the word *bank*. For the word *river* only one noun sense is present in the WordNet described in Table 6.

Table 7 shows the trace definition present for *bank#n#1*, *bank#n#2*, *bank#n#3* and *river#n#1*. Trace definition shows how the word is present in WordNet taxonomy. From these tables, it is quite clear that though the word *bank* is actually related with the financial institution, here, by default all semantic relatedness measures take an incorrect sense of *bank*.

From this survey it is found that, till not no semantic relatedness or similarity measure is there in which score can be found on sentence level. In fact, a word should take that sense which is suitable with context of the sentence. Existing semantic measures depend on individual words rather than complete sentence. Therefore, it is important to find exact sense of both words for which the senses are appropriate for the sentences.

It is quite clear that semantic measures needs actual sense while finding its score between two words. Scores obtained from different measures are not appropriate. Default sense does not give the assurance of accuracy of score for specific sentences. A word's sense depends on other words present in the same sentence. Sense detection is important to increase the accuracy of measures. Therefore, finding sense of a word is much essential to get accurate relatedness score between two words as well as between two sentences.

5 Proposed Unsupervised Method for Word Sense Disambiguation (WSD)

The overall process for finding the sense of a target word is shown in the following Fig. 2:

To get the correct sense of a word, we take collocation feature. Following section shows how collocation feature can be applied for finding word sense.

Fig. 1: Fragment WordNet concept hierarchy

Table 2: Synset and gloss of word ‘interest’ in WordNet

Synset(‘interest.n.01’)	a sense of concern with and curiosity about someone or something
Synset(‘sake.n.01’)	a reason for wanting something done
Synset(‘interest.n.03’)	the power of attracting or holding one’s attention (because it is unusual or exciting etc.)
Synset(‘interest.n.04’)	a fixed charge for borrowing money; usually a percentage of the amount borrowed
Synset(‘interest.n.05’)	(law) a right or legal share of something; a financial involvement with something
Synset(‘interest.n.06’)	(usually plural) a social group whose members control some field of activity and who have common aims
Synset(‘pastime.n.01’)	a diversion that occupies one’s time thoughts (usually pleasantly)
Synset(‘interest.v.01’)	excite the curiosity of; engage the interest of
Synset(‘concern.v.02’)	be on the mind of
Synset(‘matter.to.v.01’)	be of importance or consequence

Table 3: Semantic Relations among Senses in WordNet

Relation Name	Meaning	Example
Synonymy:	identical or nearly identical	car/ automobile
Hyponymy:	one sense is subclass of other	car and vehicle
Hypernym:	one sense is super ordinate class of another	vehicle and car
Meronymy:	the part-whole relation	leg is a part of chair
Holonymy:	whole to part relation	body and hand
Troponymy:	manner of doing something	stroll and walk
Entailment:	deduction or implication	tease and disappoint
Antonymy:	senses with opposite meaning	up/down

5.1 Finding Collocation Score between Two Words

Collocation refers to the word or phrase that is often used with other word or phrase. With collocation, we can find what words occur near other words. The computational technique that finds commonly collocated words or phrases in a document or corpus is known as collocation extraction. Collocation score between two words is calculated by finding the number of occurrence

of those words together in a corpus. Here, Wikipedia corpus (Denoyer and Gallinari (2006)) is used. The co-occurrence between two terms is calculated by finding its bi-gram frequency. Collocation gives the associativity between two words. For example: car and bike are two words that are semantically similar. They have some common features like wheels or have common function like transport. In contrast, car and petrol both are associated as they occur frequently in language and

Table 4: Relatedness/similarity score between ‘bank’ and ‘river’

Different semantic relatedness/similarity measure	Relatedness Score
The relatedness of bank#n#1 and river#n#1 using vector_pairs Li et al. (2009)	0.0353
The relatedness of bank#n#3 and river#n#1 using vector Li et al. (2009)	0.1958
The relatedness of bank#n#1 and river#n#1 using hso Hirst et al. (1998)	0
The relatedness of bank#n#1 and river#n#1 using Adapted lesk Banerjee and Pedersen (2002)	16
The relatedness of bank#n#1 and river#n#1 using res Resnik (1995)	0.6144
The relatedness of bank#n#1 and river#n#1 using lch Leacock and Chodorow (1998)	1.4917
The relatedness of bank#n#1 and river#n#1 using lin Li et al. (2003)	0.0782
The relatedness of bank#n#1 and river#n#1 using jcn Jiang and Conrath (1997)	0.0691
The relatedness of bank#n#1 and river#n#1 using wup Wu and Palmer (1994)	0.4286
The relatedness of bank#n#1 and river#n#1 using path Rada et al. (1989)	0.1111

Table 5: Different senses present for the word ‘bank’ present in WordNet

Sense Number	Meaning
1	sloping land
2	a financial institution that accepts deposits and channels the money into lending activities
3	a long ridge or pile
4	an arrangement of similar objects in a row or in tiers
5	a supply or stock held in reserve for future use
6	the funds held by a gambling house or the dealer in some gambling games
7	a slope in the turn of a road or track
8	a container (usually with a slot in the top) for keeping money at home
9	a building in which the business of banking transacted
10	a flight maneuver; aircraft tips laterally about its longitudinal axis (especially in turning)

Table 6: Sense present for the word ‘river’

Sense Number	Meaning
1	a large natural stream of water (larger than a creek)

Table 7: Trace Definition present in WordNet

Concept	Trace Definition
bank#n#1	*Root*#n#1 entity#n#1 physical_entity#n#1 object#n#1 geological_formation#n#1 slope#n#1 bank#n#1
bank#n#2	*Root*#n#1 entity#n#1 abstraction#n#6 group#n#1 social_group#n#1 organization#n#1 institution#n#1 financial_institution#n#1 depository_financial_institution#n#1
bank#n#3	*Root*#n#1 entity#n#1 physical_entity#n#1 object#n#1 geological_formation#n#1 natural_elevation#n#1 ridge#n#1 bank#n#3
river#n#1	*Root*#n#1 entity#n#1 physical_entity#n#1 thing#n#12 body_of_water#n#1 stream#n#1 river#n#1

Fig. 2: Block Diagram of Word Sense Detection Method

space. This can be said as functional relationship. Association and similarity both are neither mutually exclusive nor independent. Car and patrol both are related to some degree (McRae et al. (2012)) (Plaut (1995)). To find the bi-gram collocation score for each sense of word w_1 (McKeown and Radev (2000)), we find frequency of occurrence of words present in the sense definition for w_1 with other words present in the sentences and take the maximum one. For example, we take one sentence *Mary treated John for his injuries*. To find the exact sense of the word *treat*, we first find out the all the senses present for *treat* in WordNet. Senses are the glosses or the definitions. The method finds the collocation score of each word present in the gloss with the word present in the sentence. Here, one gloss for *treat* is *interact in a certain way*. The content words present in the sentence is *injuries* and in the definitions are *interact*, *certain*, *way*. After finding collocation score of *interact*, *certain*, *way* with *injuries*, the method takes the highest score. In this way method will calculate for each sense and finally we take that sense for which the collocation score is maximum.

The proposed WSD method will work for all the content words present in WordNet except the person's name. We will not consider the person's name, but of course, we will include an organization or location's name using standard Named Entity Tagger (Perkins (2014)). The proposed method considers mainly the content words as they carry the salient information. Content word includes noun, main verb, adjective, and adverb. First, the proposed WSD method finds all the senses of the target words present in the WordNet. Senses are the glosses. For each sense of a target word, we have removed the stop words. We also remove the stop words from the sentence in where target word is present. To find the collocation score between two words (one word is from the gloss of w and other word is from the sentence w'), we use the following equation 1. Here, Wikipedia corpus is used (Denoyer and Gallinari (2006)).

$$collocation_score(w, w') = \frac{\log(\frac{x * sizeCorpus}{(w * w' * span)})}{\log(2)} \quad (1)$$

w = frequency of the word w present in the Wikipedia corpus

w' = frequency of the word w' present in the Wikipedia corpus

x = frequency of w' near w present in the Wikipedia corpus

sizeCorpus = size of the Wikipedia corpus

span = width of words (e.g. 3 to left and 3 to right of first word)

While finding the collocation extraction score, we are giving the flexibility that if the two words are not together in Wikipedia, we increase the window size up to 3. We consider the span size as 3 because it works best for our proposed WSD method. We will search for bi-gram frequency where the words may be separated by three other words in the text of Wikipedia. To find the sense of a word present in a sentence, initially, we get a set of probable senses of the word. Now for each sense, we calculate the collocation score between each content word of every gloss of a sense with all other content words present in the same sentence. Same process will be followed for every sense and finally we take that sense for which the collocation score is maximum. The collocation score (CS) of a sense (gloss) for a target word (TW) present in the sentence is:

$$CS(Sense, Sentence) = \max_{w \in Sense, w' \in Sentence} \sum (collocation_score(w, w')) \quad (2)$$

After finding the collocation score of the set of all senses of TW , we consider that sense of TW for which the collocation score is maximum.

5.2 Finding Exact Sense of a Word Present in a Sentence

The proposed WSD method is implemented to find the sense of a word which will further helps in calculating semantic relatedness score between query and input text sentences. Following Algorithm 2 gives the systematic steps to find the sense of a word:

6 Detailed Description of the Sense-Oriented Semantic Relatedness Calculation between Two Sentences

Relatedness implies larger set of potential relationship between words. Antonyms are highly related words but they are less similar. *Human* and *women* are not similar but highly related than *human* and *car* while *girl* and *women* have similarity. Thus *Similarity* is a sub case of *Relatedness*. We prefer to use semantic relatedness score to find query relevant sentences. It is quite clear that semantic measures needs actual sense while finding its score between two words. A word's sense depends on other words present in the same sentence. In case of semantic relatedness calculation, default sense does not give the assurance of accuracy of score for specific sentences. Hence, sense detection is important to

Data: Target Word (T) and Other Words Present in the Sentence (OT)

Result: Sense Number and Gloss of that Sense of T for Which the Collocation Score is Maximum

```

1 Do the part of speech of  $T$ 
2 if  $T$  is a person's name then
3   | Go to step 24
4 end
5 else
6   Find out the senses of  $T$  using WordNet where
7   if  $T$  has more than one sense then
8     for each sense ( $s$ ) of  $T$  do
9       Do the pre-processing of  $s$ 
10      Do the pre-processing of  $OT$ 
11      Find out the collocation score between  $s$ 
        and  $OT$  by using equation 2
12    end
13    Extract the sense number and the gloss of  $T$ 
        for which we get the maximum collocation
        score
14    if  $T$  has more than one sense having same
        maximum collocation score then
15      | Extract the sense number and gloss which
        | has lower sense number
16    end
17    else
18      | Go to step 21
19    end
20    else
21      | Get the sense number and the gloss
22    end
23  end
24 end

```

Algorithm 2: Steps to Find Exact Sense of a Word present in the Sentence

increase accuracy of measures. To find sense-oriented semantic relatedness score between query and input text sentences we combine sentence semantic relatedness score, sense relatedness score and word order relatedness score. Semantic relatedness measure gives how much two sentences are related to each other. Sense relatedness measure defines how much two sentences are related on the basis of its sense definition and word order similarity gives information relations between sentences. All these semantic and same sequence of words information play an important role while finding sense-oriented sentence semantic relatedness score between two sentences. In the following section, sense-oriented semantic relatedness score calculation is shown sequentially.

6.1 Finding Semantic Relatedness Score between Two Sentences

To find the relatedness between two sentences, semantic relatedness score is calculated between two words present in both sentences. Semantic relatedness mea-

sure gives the relatedness score between two sentences on the basis of the meaning of the sentences. This relatedness measure is based on word to word relatedness.

We consider only the content words while calculating the semantic relatedness score between two sentences. Initially, pre-processing step removes the stop words. Stemming is also done to get the root form of a word. Before finding the relatedness score between two words, we find the exact sense of that word.

This relatedness score uses Path Weight measure (Pedersen et al. (2004)) (Hirst et al. (1998)). It is a semantic relatedness measure between two words based on path which is described in WordNet lexical dictionary. It is found from the literature survey that this measure includes more relations and can find relatedness score between different part of speech (Patwardhan et al. (2003)). It classifies the relations in WordNet as upward, downward or horizontal directions. Higher score implies the shorter path length and less changes of directions. While finding the semantic relatedness score between two words, the proposed method will give the accurate sense number and its part of speech along with the respective words. The equation to find Semantic Relatedness Score (SEM_R_S) between two words $w1$ and $w2$ is:

$$SEM_R_S((w1, sense_no_1, p_o_s_1), (w2, sense_no_2, p_o_s_2)) = 2 \times c - PathLength(w1, w2) - k \times DirectionChange(w1, w2) \quad (3)$$

here,

$sense_no_1$ = sense number of $W1$

$p_o_s_1$ = part of speech of $W1$

$sense_no_2$ = sense number of $W2$

$p_o_s_2$ = part of speech of $W2$

Here, c and k are the constants and values are $c = 8$ and $k = 1$. The maximum semantic relatedness score between two word is 16 which signifies that two content words are identical. The minimum score is 0 which signifies there is no semantic relatedness between them. Following equation 4 is used to find the Semantic Relatedness Score (SEM_R_S) between two sentences $s1$ and $s2$:

$$SEM_R_S(s1, s2) = \frac{\sum_{w1 \in s1, w2 \in s2} SEM_R_S((w1, sense_no_1, p_o_s_1), (w2, sense_no_2, p_o_s_2))}{Maximum\ relatedness\ score} \quad (4)$$

The overall process to find semantic relatedness score between two sentences is shown in the Algorithm 3.

Table 8: Different gloss definition for different senses for the word *visit*

Synset('visit.n.01')	the act of going to see some person or place or thing for a short time
Synset('visit.n.02')	a meeting arranged by the visitor to see someone (such as a doctor or lawyer) for treatment or advice
Synset('visit.n.03')	the act of visiting in an official capacity (as for an inspection)
Synset('visit.n.04')	the act of going to see some person in a professional capacity
Synset('sojourn.n.01')	a temporary stay (e.g., as a guest)
Synset('visit.v.01')	go to see a place, as for entertainment
Synset('travel.to.v.01')	go to certain places as for sightseeing
Synset('visit.v.03')	pay a brief visit
Synset('visit.v.04')	come to see in an official or professional capacity
Synset('inflict.v.01')	impose something unpleasant
Synset('chew_the_fat.v.01')	talk socially without exchanging too much information
Synset('visit.v.07')	stay with as a guest
Synset('visit.v.08')	assail

Data: Two Sentences (s_1, s_2)

Result: Semantic Relatedness Score between s_1 and s_2 ($SEM_R_S(s_1, s_2)$)

- 1 Do the pre-processing of s_1, s_2 by using the steps mentioned in section 3
- 2 **for** each content word w_1 in s_1 and w_2 in s_2 **do**
- 3 Find out the sense number of w_1 and w_2 by using the Algorithm 2
- 4 **end**
- 5 **for** each content word w_1 and w_2 from s_1 and s_2 **do**
- 6 Find out the $SEM_R_S((w_1, \text{sense_no_1}, \text{p_o_s_1}), (w_2, \text{sense_no_2}, \text{p_o_s_2}))$ by using the equation 3
- 7 **end**
- 8 Find the $SEM_R_S(s_1, s_2)$ between s_1 and s_2 by using the equation 4

Algorithm 3: Steps to Find Semantic Relatedness Score between Two Sentences

6.2 Finding Sense Relatedness Score between Two Sentences

We have already discussed that finding relatedness measure between two sentences on the basis of its sense is quite important. Proposed WSD method finds the sense of each content word present in both text sentences by using the mentioned method described in chapter 3.5.2. After doing the pre-processing of two sentences as mentioned in section 3.5, we have disambiguated the sense of each content word if it has more than one sense. We get the gloss for each content word. We will do the stop word removal and stemming on the content words to get the root form of the content words present in the gloss. Here, we find the sense relatedness score between two sentences by finding the common content words present in the gloss of a sense of a content word in the first sentence with the words present in the second sentence. The method uses the equations 5, 6 and 7 to find the sense relatedness score (S_R_S) between two sentences

s_1 and s_2 .

$$S_R_S1(s_1, s_2) = \max_{w \in s_1} \sum \{ \text{Sense definition of } w \cap \text{Words present in } s_2 \} \quad (5)$$

$$S_R_S2(s_1, s_2) = \max_{w' \in s_2} \sum \{ \text{Sense definition of } w' \cap \text{Words present in } s_1 \} \quad (6)$$

$$S_R_S(s_1, s_2) = \frac{S_R_S1(s_1, s_2) + S_R_S2(s_1, s_2)}{\text{Total number of content words present in } s_1 \text{ and } s_2} \quad (7)$$

The overall process to find the sense relatedness score between two sentences is shown in the Algorithm 4.

Data: Two Sentences (s_1, s_2)

Result: Sense Relatedness Score between s_1 and s_2 ($S_R_S(s_1, s_2)$)

- 1 Do the pre-processing of s_1 and s_2 by using the steps mentioned in section 3
- 2 Find out the gloss of the sense of the content words by using the Algorithm 2
- 3 Do the pre-processing of the gloss of the sense of the content words by using the steps mentioned in section 3
- 4 Find the $S_R_S(s_1, s_2)$ between s_1 and s_2 by using the equation 7

Algorithm 4: Steps to Find Sense Relatedness Score between Two Sentences

6.3 Finding Word Order Similarity Score between Two Sentences

On the basis of same sequence of words present in the two sentences, word order similarity provides how much two sentences are similar. Finding longest common substring between two sentences adds more impact on similarity measure. Word order similarity method depends on the order of words present in both sentences. It helps in signifying the relatedness between two sentences though they share same words. Example: (a) Ram killed Shyam and (b) Shyam killed Ram. These two sentences use same content words but we can see that sentence *a* and sentence *b* have opposite meaning. Longest common substring can easily distinguish that meaning of sentence *a* is not similar to the meaning of sentence *b*. We can take another example: sentence 1 is Narendra Modi's visit to China and sentence 2 is Ram Nath Kovind's visit to China. Both sentences are different though they carry maximum common words. Hence finding longest common substring can identify the differences present in both sentences. It can also find similarity between two sentences if numerical data present in both sentences in a same order. Ontology based semantic relatedness measure can not find this type of similarity as lexical dictionary does not contain numerical data or some proper nouns. Example: sentence 1: In 2006, Ram came to Assam to meet his friend Rahim and sentence 2: In 2006, Ram came to Guwahati to meet Rahim. Here longest common substring is In 2006, Ram came to. Hence, it also helps in finding similarity for numerical and proper noun words. Here, we will not do any pre-processing task. The method uses the following equation 8 to find word order similarity between two sentences s_1 and s_2 :

$$W_O_S(s_1, s_2) = \frac{NCW(s_1 s_2)}{TNWLS(s_1 s_2)} \quad (8)$$

Here,

NCW=Number of Common Words between s_1 and s_2
 TNWLS=Total Number of Words present in the Longest Sentence between s_1 and s_2

6.4 Finding Sense-Oriented Sentence Semantic Relatedness Score between Two Sentences

The sense-oriented sentence semantic relatedness measure between the two sentences (query and input text sentence) is the combination of all these three measures: semantic relatedness, sense relatedness and word order similarity.

The sense-oriented sentence semantic relatedness score between sentences s_1 and s_2 is given in equation 9:

$$Sense_Sem_Rel(s_1, s_2) = \alpha \times SEM_R_S(s_1, s_2) + \beta \times S_R_S(s_1, s_2) + \gamma \times W_O_S(s_1, s_2) \quad (9)$$

Here $\alpha + \beta + \gamma = 1$ and α , β and γ are weighting parameters. They specify relative contributions to the sense-oriented sentence semantic relatedness measure of semantic, sense and word order measures. As semantic information carries more relevant information, therefore more weightage is given to semantic relatedness measure Wiemer-Hastings (2000). Considering the view that the word order information plays a subordinate role in finding relatedness between sentences, hence the weightage given to word order information is minimum.

The overall process to find sense-oriented sentence semantic relatedness score between two sentences is shown in the Algorithm 5.

Data: Two Sentences (s_1, s_2)

Result: Sense-Oriented Sentence Semantic

Relatedness Score (*Sense_Sem_Rel*) of s_1, s_2

- 1 Find semantic relatedness score between s_1 and s_2 by using the Algorithm 3
- 2 Find sense relatedness score between s_1 and s_2 by using the Algorithm 4
- 3 Find word order similarity score between s_1 and s_2 by using the equation 8
- 4 Find sense-oriented sentence semantic relatedness score between s_1 and s_2 by using equation 9

Algorithm 5: Steps to Find Sense-Oriented Sentence Semantic Relatedness Score

7 Generation of Redundancy Free Text Sentences

To find the redundant sentences from the highly semantically related input text sentences with query, we select certain features thorough which different clusters are created. We have taken five relevant features which are important for generating query-based text summary. Five features exploit a set of characteristics from the input text sentences. From the literature survey, it is seen that if a sentence carries five features, then the sentence must be an informative one Widyassari et al. (2020). Our main motive is to extract those semantically related non-redundant sentences which contain features. As our dataset is a newswire dataset, therefore, these mentioned features are highly feasible which will gives us an informative summary. The five features listed as:

1. Proper noun: Presence of proper noun or entity name in a sentence indicates an important sentence. We use a part of speech tagger (Bird et al. (2009)) to tag the proper nouns.
2. Numerical data: Numerical data gives more significant information in a textual document and consider it as an important one.
3. Thematic word: Frequently occurring words are considered as a thematic words. Thematic word makes the sentence an informative one. Here, we take top ten most frequently used words present in the text document.
4. Cue phrase: Cue phrase like *this letter*, *this report*, *summary*, *in conclusion*, *argue*, *purpose*, *development* are added as an important sentence. We use the English cue phrase list.
5. Sentence Length: We consider the length of the sentence.

7.1 The Process of Creating K Clusters from Sentences

We have created the clusters of sentences based on the frequency of occurrence of those features present in the sentence. To create clusters, we use K-Means clustering algorithm on text data Andrews and Fox (2007). We remove the stop words and have done the stemming on the sentences. We represent all the sentences by a corpus vector based on the frequency of occurrence of five mentioned features. Initially, five centroids are chosen randomly by the K-Means algorithm $\{C_1, C_2, C_3, C_4, C_5\}$. The next step is to calculate the Euclidean distance between the other sentences from the each five centroids. The equation to find the Euclidean distance between one sentence vector $[pf_s, nf_s, tf_s, cf_s, sl_s]$ and one centroid vector $[pf_c, nf_c, tf_c, cf_c, sl_c]$ is:

$$((pf_s - pf_c)^2 + (nf_s - nf_c)^2 + (tf_s - tf_c)^2 + (cf_s - cf_c)^2 + (sl_s - sl_c)^2)^{0.5} \quad (10)$$

Here,

pf_s = frequency of occurrence of proper noun in the sentence

nf_s = frequency of occurrence of numerical data in the sentence

tf_s = frequency of occurrence of thematic word in the sentence

cf_s = frequency of occurrence of cue phrase in the sentence

sl_s = sentence length of the sentence

pf_c = frequency of occurrence of proper noun in the centroid

nf_c = frequency of occurrence of numerical data in the centroid

tf_c = frequency of occurrence of thematic word in the centroid

cf_c = frequency of occurrence of cue phrase in the centroid

sl_c = sentence length of the centroid

We have to update the centroid value. The equation for updating the centroid value after each iteration is:

$$C_i = \left(\frac{1}{K_i}\right) \sum_{j=1}^{K_i} VS_i \quad (11)$$

Here,

K_i = Number of vectors present in C_i cluster

$VS_i = i^{th}$ Sentence vector

The K-Means algorithm will repeat the whole process until it converges.

The quality of cluster c_i can be measured by the within cluster variation, which is the sum of squared error between all objects in c_i and the centroid C_i , defined as

$$E = \sum_{i=1}^K \sum_{VS' \in c_i} dist(VS', C_i)^2 \quad (12)$$

where E is the sum of the squared error for all sentence vector in the data set;

VS' is the point in space representing a given sentence vector;

C_i is the centroid of cluster c_i (both VS' and C_i are multidimensional).

The overall process to create the clusters with redundancy free sentences is shown in the Algorithm 6.

7.2 Proposed Method to Extract Redundancy Free Sentences

There is a high probability of presence of redundant sentences in different clusters. We need to remove the redundant sentences. If two sentences are highly semantically related, then one of the sentences is redundant. We have already proposed one sense-oriented semantic relatedness measure between two sentences in section 6. We have taken threshold value as above 60%. If the sense-oriented semantic relatedness measure between two sentences is above 60%, then we will remove the redundant sentence. The overall process to remove redundant sentences from text documents is shown in the Algorithm 7.

```

1  Data: Query-Related Sentences ( $S$ ) and Number of
    Clusters =  $K$ 
Result:  $K$  no of Clusters Containing the Sentences
2  Do the stop word removal and stemming on  $S$  ( $S'$ );
3  Generate the sentence vectors on the basis of
    frequency of occurrence of the mentioned features in
    the sentence;
4  Arbitrarily choose  $K$  no of sentence vectors from  $S$  as
    the initial cluster centers;
5  Repeat
6  (Re)assign each sentence vector to the cluster to which
    the sentence vector is the most similar, based on the
    mean value of the sentence vectors in the cluster;
7  Update the cluster means, that is, calculate the mean
    value of the sentence vectors for each cluster;
8  Until no change;
Algorithm 6: Steps to Create Clusters from Query-
    Related Sentences

```

```

Data: Set  $Q$  of Query-related sentences
Result: Set  $RFQ$  containing redundancy free
    Query-related sentences
1  Create clusters containing the Query-related
    sentences,  $Q$  by using the Algorithm 6;
2  for each cluster  $C$  in set  $Q$  do
3       $S$  = Top sentence in Set  $Q$ ;
4      while  $S$  is not Null do
5           $S'$  = The sentence next to sentence  $S$ ;
6          while  $S'$  is not Null do
7              Find sense-oriented
                Sentence-Semantic-relatedness Score
                between sentences  $S$  and  $S'$  by using
                equation 4.7;
8              if  $Sentence\_Semantic\_relatedness\_Score > Threshold$ 
                then
9                  Remove sentence  $S'$  from cluster  $C$ ;
10             end
11              $S'$  = Sentence next to  $S'$ ;
12         end
13          $S$  = Sentence next to  $S$ ;
14     end
15 end
16 Output the set of remaining Query-Related sentences
    (set of remaining sentences in all clusters);
Algorithm 7: Steps to Find Redundancy Free
    Query-Related sentences

```

In the Algorithm 7, similarity of a sentence is not computed with all the sentences in all other clusters. Similarity is computed among the sentences present in a cluster. Highly query relevant sentences are extracted to form the initial summary that may contain redundant sentences. Clusters are created for the purpose of removing the redundancy. Each cluster contains similar sentences. Therefore, only a few sentences of a cluster will be selected for inclusion in the summary. Here comes the need for computing similarity of one sentence with other sentences in the cluster. If the similarity of the sentence with any other sentence is found

to be greater than a threshold then that sentence is rejected. Clusters are created based on five different features (proper noun, numerical data, thematic word, cue phrase and sentence length). We know that if sentence carries above mentioned five features, then the sentence must be an informative one (Widyassari et al. (2020)). Each of the sentences in the initial summary is represented by frequency of occurrence of the five features, then k-means algorithm is used for clustering. Here k is considered as an user input. Value of k should not be so large, a value in the range [5-10] should be sufficient. We have used 5 clusters in the experimental evaluations.

The proposed algorithm first selects the sentence at the top of the text in a cluster and it calculates the semantic relatedness score with each subsequent sentence present in that cluster. If the score is above threshold value, then the other sentence is removed from the cluster. Thus, some sentences may be removed from the cluster in the first pass. Then the control moves to the next sentence below the top sentence which is not removed. The second pass to remove redundant sentence begins. Semantic relatedness score of this sentence with all remaining subsequent sentences are calculated and those sentences having semantic relatedness score above threshold value are removed. The process continues until the last sentence of the cluster is examined for redundancy removal. In the same way redundancy removal is done for all the clusters. Set of sentences remaining in the clusters form the redundancy free summary in the original order. The result depends on the order of the sentences. The algorithm proceeds from top to bottom. If the order of the sentences in the text is changed, then a slightly different summary may result.

Say, a cluster contains 3 sentences $S1$, $S2$ and $S3$ and $Sentence_Semantic_relatedness_Score(S1, S2)$ is 0.6 and $Sentence_Semantic_relatedness_Score(S1, S3)$ is 0.3 and $Sentence_Semantic_relatedness_Score(S2, S3)$ is 0.7. If threshold value is 0.5, $S1$ and $S3$ sentences will be finally selected from the cluster. Let us assume that the algorithm starts with the $S1$ sentence. The value of $Sentence_Semantic_relatedness_Score(S1, S2)$ is 0.6. It is greater than the threshold value; therefore, $S2$ will be removed. The value of $Sentence_Semantic_relatedness_Score(S1, S3)$ is 0.3 which is smaller than the threshold value. Sentences $S3$ will be kept; therefore, the proposed method keeps $S1$ and $S3$ sentences.

8 Evaluation of Proposed Word Sense Disambiguation Technique

8.1 Brief Description of Comparison Systems

Different widely used supervised and knowledge-based word sense disambiguation systems are included for comparison purpose. All the supervised systems use the same corpus SemCor and Semcor+OMSTI for training purpose. This gives a fair comparison. Supervised WSD systems includes the following techniques:

- Support Vector Machine (SVM) (Zhong and Ng (2010)) classifier is used in It Makes Sense (IMS) Word Sense Disambiguation (WSD) system. Different features like: surrounding words, PoS tag of surrounding words and local collocations are taken.
- In the works of (Taghipour and Ng (2015), Rothe and Schütze (2015), Iacobacci et al. (2016)), word embeddings is used. Different methods are proposed by Iacobacci et al. (2016) where word embedding is used in current supervised WSD systems. They have made a deep analysis how different parameters are affecting the performance of WSD system. Here, two best configurations are considered having one with surrounding words (IMS+emb) and other one without surrounding words (IMS-s+emb). In both methods, they integrates word embeddings by using exponential decay. To train the word embeddings, Iacobacci et al.'s suggested learning strategy and hyper parameter's are used (Raganato et al. (2017b)).
- Now a days, Neural language based model is used widely for WSD task (Melamud et al. (2016), Kågebäck and Salomonsson (2016), Yuan et al. (2016)). Our experiment uses bidirectional LSTM (Melamud et al. (2016)) model. This context2vec neural model learns a generic embedding function for variable length contexts of target words.
- A baseline method is included in Table 9 by taking the Most Frequent Sense (MFS) heuristically. Senses are selected based on the highest number of occurrences in the training corpus (Raganato et al. (2017b)).

Knowledge-based system includes following three Word Sense Disambiguation models:

- Lesk is simple knowledge based WSD algorithm (Lesk (1986)) which finds similar words between the definition of each sense with context of the target word. For comparison purpose, extended version of Lesk algorithm is used where definition of related senses are also included. Here, for word weighting, conventional term frequency-inverse document frequency

is used (Sparck Jones (1972), Banerjee and Pedersen (2003)). For better analysis, word embedding is added in enhanced version of Lesk which helps in computing similarity between definition and context of the target word (Basile et al. (2014)).

- A graph based WSD system is proposed by Agirre et al. (Agirre and Soroa (2009)) where random walk is used over a WordNet semantic network (Agirre et al. (2014)). In their method, a personalized Page Rank algorithm (Haveliwala (2002)) is used.
- Babelfy is a graph-based WSD approach where random walk is used to find connections between synsets (Moro et al. (2014)). Babelfy uses random walks with restart (Tong et al. (2006) over BabelNet Navigli and Ponzetto (2012)). Babelfy includes the whole document while finding its sense.
- A baseline method is chosen to select first sense as a correct sense present in WordNet 3.0 (Bird et al. (2009)). All the existing semantic similarity or relatedness measure uses the first sense while finding its semantic similarity or relatedness score.

8.2 Evaluation Metric

We use F-Score criteria for evaluation metric by using the equation 15 (Dongsuk et al. (2018)). It is a harmonic mean of precision and recall value shown in equations 13 and 14.

$$Precision = \frac{\text{Number of correctly disambiguated answers}}{K} \quad (13)$$

K=Number Of Words in the test set for which the algorithm made a prediction

$$Recall = \frac{\text{Number of correctly disambiguated answers}}{\text{Number of all test instances}} \quad (14)$$

$$F - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (15)$$

We first evaluate our proposed WSD (Word Sense Disambiguation) on publicly available English WSD corpora Senseval-2, Sensval-3 task1, SemEval-2007 task17, Sem Eval-2013 task 12 and SemEval-2015 task 13 (Raganato et al. (2017a)).

Results in the Table 9 clearly shows that though many supervised methods outperforms existing knowledge-based methods, but our proposed WSD method

Table 9: F-Measure scores of different WSD Methods for all five datasets

Approach	Tr. Corpus	System	Senseval-2	Senseval-3	SemEval-07	SemEval-13	SemEval-15
Supervised	SemCor	IMS	70.9	69.3	61.3	65.3	69.5
		IMS+emb	71.0	69.3	60.9	67.3	71.3
		IMS-S+emb	72.2	70.4	62.6	65.9	71.5
		Context2Vec	71.8	69.1	61.3	65.6	71.9
		MFS	65.6	66.0	54.5	63.8	67.1
	SemCor+OMSTI	IMS	72.8	69.2	60.0	65.0	69.3
		IMS+emb	70.8	68.9	58.5	66.3	69.7
		IMS _{-s} +emb	73.3	69.6	61.1	66.7	70.4
		Context2Vec	72.3	68.2	61.5	67.2	71.7
		MFS	66.5	60.4	52.3	62.6	64.2
Unsupervised (Knowledge-based)		Lesk _{ext}	50.6	44.5	32.0	53.6	51.0
		Lesk _{ext} +emb	63.0	63.7	56.7	66.2	64.6
		UKB	56.0	51.7	39.0	53.6	55.2
		UKB _{-gloss}	60.6	54.1	42.0	59.0	61.2
		Babelify	67.0	63.5	51.6	66.4	70.3
		WN 1 st sense	66.8	66.2	55.2	63.0	67.8
		Proposed WSD	75.4	71.6	63.7	77.8	75.3

has performed much better for different datasets. It is also noticed from the Table that the performance of WSD methods for all datasets are not uniform. A large performance gap is seen between the best and worst performing dataset. For dataset SemEval-07, performance is quite low for all WSD systems as this dataset is the most ambiguous one.

We compare our proposed WSD approach to other BabelNet-based unsupervised and supervised WSD systems (Dongsuk et al. (2018)). BabelNet is a multilingual lexical semantic network. It is automatically created by linking Wikipedia to the WordNet (Navigli and Ponzetto (2012)). Unsupervised systems are listed as: Moro et al. (Moro et al. (2014)), Agirre et al. (Agirre et al. (2014)), Apidianaki et al. (Apidianaki and Gong (2015)), Tripodi et al. (Tripodi and Pelillo (2017)), Dongsuk et al. (Dongsuk et al. (2018)) and supervised systems are: Zhong et al. (Zhong and Ng (2010)), Weissenborn et al. (Weissenborn et al. (2015)), Raganato et al. (Raganato et al. (2017b)), Pasini et al. (Pasini and Navigli (2017)). From experimental results, it is seen that our proposed WSD method outperforms all listed WSD systems for SemEval-2013 dataset. For the SemEval-2015 dataset, our proposed WSD method has similar performance to the supervised Weissenborn et al. method. However, in terms of macro average F-score of both datasets SemEval-13 and SemEval-15, the proposed WSD method shows higher performance for all WSD systems present in the Table 10. A macro-average calculates the metric independently for each class and then takes the average value.

9 Dataset Description for Query-Based Text Summarization Method

Here, we have experimented our proposed algorithm with DUC (Document Understanding Conference) 2005, 2006 and 2007 datasets (<http://duc.nist.gov>). We try to evaluate the effectiveness of proposed method with existing systems that perform experimental evaluation using DUC and also with the current related systems.

All the three DUC 2005, 2006 and 2007 datasets contain real life complex questions, particularly used for query-based text summarization purpose. Each dataset has a query and related text documents. There are 50 queries with 50 different topics for DUC 2005 and DUC 2006 datasets. For DUC 2007 datasets, it has 45 different number of topics. Each summary length is of 250 words only. A brief description of datasets are shown in Table 11. Documents present in each cluster of DUC 2005 are selected from the Los Angeles Times and Financial Times of London. For 2006 and 2007 datasets, documents are taken from the Associated Press and New York Times (1998-2000) and Xinhua News Agency (1996-2000). Our proposed method is compared with existing methods participated in DUC 2005, 2006 and 2007 datasets.

10 Evaluation Metric for Query-Based Text Summarization

DUC produces query-oriented text summarization with 250 words. Pre-processing part is done on the sentences. It includes stop word removal and stemming. The popu-

Table 10: Performance comparison of different BabelNet-based unsupervised and supervised state-of-the-art methods

Approach	System	F-score for SemEval-13	F-score for SemEval-15	Macro Avg F-score
Unsupervised (Knowledge-based)	Moro 14	66.4	70.3	68.4
	Agirre 14	62.9	63.3	63.1
	Apidianaki 15	-	64.7	-
	Tripodi 17	70.8	-	-
	Wordsim_iter _{SRP2vSim} 18	75.0	65.8	70.4
	Proposed WSD	77.8	75.3	76.6
Supervised	Zhong 10	66.3	69.7	68.0
	Weissenborn 15	71.5	75.4	73.5
	Raganato 17	66.9	71.5	69.2
	Pasini 17	65.5	68.6	67.1

Table 11: Experimental Description of Dataset

DUC Dataset	2005	2006	2007
Number of clusters	50	50	45
Number of average documents in each cluster	32	25	25
Data source	TREC	AQUAINT	AQUAINT
Number of sentences	45931	34560	24282
Length of summary	250 words	250 words	250 words

lar and standard ROUGE (Recall-Oriented Understudy for Gisting Evaluation) (Lin (2004)) intrinsic based metric has been adapted by NIST (National Institute for Standards and Technology) and widely used by many researchers for text evaluation purpose. Candidate summary and reference summary are compared in ROUGE. Different systems generate candidate summary and expert human made reference summary. A set of metrics are available in ROUGE for comparing different summaries. It measures the quality of summary in terms of overlapping units such as N-grams, word sequences and word pairs. N-gram Co-Occurrence Statistics, Longest Common Subsequence, Weighted Longest Common Subsequence, Skip-Bigram Co-Occurrence Statistics and Extension of Skip-Bigram Co-Occurrence Statistics are the five standard evaluation metrics available in ROUGE. ROUGE-N compares system generated summary and human summary on the basis of total number of common content words. N-gram recall measure is calculated using following equation:

$$ROUGE - N = \frac{\sum_{Refs} \sum_{N-gram \in S} Count_{match}(N - gram)}{\sum_{S \in Refs} \sum_{N-gram \in S} Count(N - gram)}$$

where

N= length of N-gram

$Count_{match}(N - gram)$ = Number of common

$N - grams$ co-occurring in both system summary and human summary

$Count(N - gram)$ = Number of $N - grams$ present in reference summary

Here, official metrics of ROUGE-1 (unigram-based), ROUGE-2 (bi-gram based) and ROUGE-SU4 are used along with 95 % confidence intervals within the square brackets. We use the average recall scores of ROUGE-1, ROUGE-2 and ROUGE-SU4 on DUC 2005, 2006 and 2007 datasets for our comparison purpose.

11 Parameter Setting for Proposed Query-Based Text Summarization Method

Before tested the proposed method, we first focus on optimizing the all three (α , β and γ) weighting parameters. α parameter signifies the importance of semantic relatedness information, where as β extracts the query relevant sentences sense wise and γ carries word order information. To set these weighting parameters, we select randomly set of documents from DUC 2005 dataset and run our proposed method to find optimal values of three parameters. All documents are considered as sentences. Pre-processing is done on the sentences to filter out unwanted words.

We evaluate our method for each α between 0.1 to 0.9 with β between .1 to .9 and γ between 0 to 1 with a step of 0.1 (e.g. $\alpha=0.5$, $\beta=0.3$ and $\gamma=0.2$). We use a nested loop to estimates the values of α , β and γ where γ is outer most loop, β is a inner loop and α is inner-most loop. In the first pass of outer loop when value of $\gamma=0$, then control enters into next inner loop where β varies from 0 to 1 and finally enters into inner most loop α , which is varied from 0.1 to 0.9. In second pass, again the outer most loop triggers the two inner loops. This repeats until outer most loop finishes. A sample of the results are shown in Table 12:

Table shows different experimental results obtained for different α , β and γ values. We evaluated it in terms of Recall scores obtained through ROUGE-1, ROUGE-2 and ROUGE-SU4 values. After analyzing the results, we have achieved highest ROUGE values for the values of $\alpha=0.5$, $\beta = 0.3$ and $\gamma = 0.2$. Recall stores for all three ROUGE matrices are: 0.39512 for ROUGE-1, 0.07521 for ROUGE-2 and 0.15631 for ROUGE-SU4. Highest contribution is given by semantic relatedness measure while finding query relevant sentences. In the Table 12, the most appropriate values are marked using boldface. As stated by Wiemer Wiemer-Hastings (2000), semantic information carries more relevant information, therefore more weightage is given to semantic relatedness measure. As a result, using current DUC 2005 dataset, we get all ROUGE peak values. Therefore, we can recommend these weightage parameter values for further analysis and comparison.

12 Performance Comparison for Proposed Query-Based Text Summarization Method

We compare the performance of our proposed query-based text summarization method with other well-known or recently proposed methods on three standard datasets on the basis of ROUGE scores. In particular, to evaluate

our proposed method on current state-of-the-art systems we use following methods: (1) Hierarchical clustering based approach (QBUEM) by Mahmud (Mahmud (2020)) (2) A topical approach based Method (MRC) by Lierde et al. (Van Lierde and Chow (2019a)), (3) hypergraph transversals based method (TL-TranSum) by Lierde et al. (Van Lierde and Chow (2019b)) (4) linguistic knowledge and content word expansion based method (QSLK) by Abadi et al. in 2017 (Abdi et al. (2017)) (5) using unsupervised multi-document summarization via deep learning (OQDE) by Zhong et al. (Zhong et al. (2015)), graph-based sentence ranking algorithm by (6) Wan (Wan (2009)), (7) Wan et al. (Wan and Xiao (2009)) and (8) Wei et al. (Wei et al. (2010)). We take following supervised learning based sentence ranking algorithms like: Support Vector Classification by (9) Vapnik (Vapnik (2013)), (10) Ranking SVM by Joachims (Joachims (2002)), (11) Regression by Ouyang et al. (Ouyang et al. (2011)) and also classical relevance and redundancy based selection algorithms like: (12) Greedy Search by Filatova et al. (Filatova and Hatzivassiloglou (2004)), (13) Maximum Marginal Relevance (MMR) by Goldstein et al. (Goldstein et al. (2000)) and (14) integer linear program (ILP) by McDonald (McDonald (2007)). Results listed in Table 13 show that our proposed method outperforms many existing and recent query-based text summarization systems.

To further prove the performance of our method, we add other recognized and current state-of-the-art existing methods particularly on DUC 2006 for comparing with proposed Method. The following methods are selected for comparison (Abdi et al. (2017)): (1) QSLK by Abadi et al. in 2017 (Abdi et al. (2017)), (2) CTMSUM by Yang et al. in 2014 (Yang (2014)), (3) WAASum by Canhasi et al. in 2014 (Canhasi and Kononenko (2014)), (4) Topical-N by Yang et al. in 2013 (Yang et al. (2013)), (4) Qs-MR by Wei et al in 2011 (Wei et al. (2011)), (5) SVR by Ouyang et al. in 2010 (Ouyang et al. (2010)), (6) LEX by Huang et al. in 2010 (Huang et al. (2010)). We select these methods as they have performed best on DUC 2006 datasets.

From the Table 14 and 15 it is seen that proposed Method shows considerably better results while applying on DUC 2006 and DUC 2007. As our method is an unsupervised learning algorithm, the performance is quite comparable to supervised learning based algorithms. In real-world applications, labeled informations are always not sufficient. Manual summaries are created by experienced human experts. Manually created summaries are often difficult, expensive and time consuming. On the contrary, with the fast growing availability of unstructured textual information, huge number of

Table 12: Performance of proposed method against various α , β and γ values on DUC 2005 dataset

Weighting (γ)	Weighting (β)	Weighting (α)	ROUGE-1	ROUGE-2	ROUGE-SU4
$\gamma=(0...0.5)$	$\beta=(0...0.5)$	$\alpha=0$	-	-	-
$\gamma=(0...0.5)$	$\beta=(0...0.5)$	$\alpha=0.1$	-	-	-
$\gamma=(0...0.5)$	$\beta=(0...0.5)$	$\alpha=0.2$	-	-	-
$\gamma=(0...0.5)$	$\beta=(0...0.5)$	$\alpha=0.3$	-	-	-
$\gamma=(0...0.5)$	$\beta=(0...0.5)$	$\alpha=0.4$	-	-	-
$\gamma=0$	$\beta=0.5$	$\alpha=0.5$	-	-	-
$\gamma=0.1$	$\beta=(0.4)$	$\alpha=0.5$	0.38567	0.07457	0.14785
$\gamma=0.2$	$\beta=0.3$	$\alpha=0.5$	0.39512	0.08931	0.15631
$\gamma=0$	$\beta=0.4$	$\alpha=0.6$	-	-	-
$\gamma=0.1$	$\beta=0.3$	$\alpha=0.6$	0.38412	0.07238	0.14276
$\gamma=0.2$	$\beta=0.2$	$\alpha=0.6$	0.38321	0.07332	0.14138
$\gamma=0.3$	$\beta=0.1$	$\alpha=0.6$	0.37927	0.07231	0.13712
$\gamma=0.4$	$\beta=0$	$\alpha=0.6$	-	-	-
$\gamma=0$	$\beta=0.3$	$\alpha=0.7$	-	-	-
$\gamma=0.1$	$\beta=0.2$	$\alpha=0.7$	0.37451	0.07154	0.12963
$\gamma=0.2$	$\beta=0.1$	$\alpha=0.7$	0.37214	0.07119	0.12472
$\gamma=0$	$\beta=0.2$	$\alpha=0.8$	-	-	-
$\gamma=0.1$	$\beta=0.1$	$\alpha=0.8$	0.37012	0.06902	0.11452
$\gamma=0.2$	$\beta=0$	$\alpha=0.8$	-	-	-
$\gamma=(0...0.1)$	$\beta=(0.1...0)$	$\alpha=0.9$	-	-	-
$\gamma=0$	$\beta=0$	$\alpha=1.0$	-	-	-

unlabeled data are available. It is much more practically feasible to construct text summarization under unsupervised learning framework. Therefore, conclusion can be derived from this analysis that our method is able to attain significant performances by providing the comparison with baseline and state-of-the-art systems by applying in different standard datasets.

13 Discussion of Performance of Query Relevance

Existing query-based text summarization methods hold extremely small query-dependent features. For example: Ouyang et al. (2011) proposed a query-based text summarization method (SVR) where both query-dependent and query independent features are present. Support Vector Regression is used here to learn feature

weights. According to the Cao et al. (2016), without query dependent features, SVR drops only for 1%. SVR can make generic summary of text documents but fails to find query relevant sentences. In this section, we do the qualitative analysis to inspect what our proposed method actually catches according to the relevance of query. Both one-sentence query and multiple sentence queries are shown in Table 16 and 17. We also show the sentences obtained from two other query-based text summarization methods: one is TF-IDF cosine similarity and other is AttSum Cao et al. (2016) for comparison.

Table 13: Experimental results on DUC 2005 dataset

System ID	ROUGE-1	ROUGE-2	ROUGE-SU4
Proposed_method	0.3951	0.0893	0.1563
QBUEM (Mahmud (2020))	.3830	.0962	.1441
MRC (Van Lierde and Chow (2019a))	-	0.0786	0.1282
TL-TranSum (Van Lierde and Chow (2019b))	-	0.0774	0.1287
QSLK (Abdi et al. (2017))	0.3862	0.0801	0.1371
OQDE (Zhong et al. (2015))	0.3751 (0.3687-0.3809)	0.0775 (0.07341-0.08136)	0.1341 (0.1303-0.1378)
Graph-based (Wan (2009))	0.3839	0.0737	0.1317
Graph-based (Wan and Xiao (2009))	0.3718	0.0676	0.1293
Graph-based (Wei et al. (2010))	-	0.0771	0.1337
Support Vector Classification (Vapnik (2013))	0.3663 (0.3569-0.3757)	0.0701 (0.0677-0.0736)	0.1243 (0.1202-0.1382)
Ranking SVM (Joachims (2002))	0.3702	0.0711	0.1299
Regression (Ouyang et al. (2011))	0.3770 (0.3713-0.3828)	0.0761 (0.0727-0.0793)	0.1329 (0.1294-0.1363)
Greedy Search (Filatova and Hatzivassiloglou (2004))	0.3560	0.0610	-
MMR (Goldstein et al. (2000))	0.3701	0.0701	0.1289
ILP (McDonald (2007))	0.3580	0.0610	-
Columbia (Evans et al. (2004))	0.37481	0.06857	0.12772
CCS-NSA-05 (Dang (2005))	0.35983	0.06278	0.11895
FTextST-0 (Dang (2005))	0.35851	0.0674	0.12324
FDUSUM (Dang (2005))	0.36088	0.06088	0.11878
ISI-Webcl (Dang (2005))	0.36584	0.06425	0.12251
SFU v2.4 (Dang (2005))	0.36012	0.06323	0.12180
OHSU-DUC05 (Dang (2005))	0.35012	0.06325	0.11896
NUS3 (Dang (2005))	0.37515	0.07251	0.13163
PolyU (Li et al. (2005))	0.36977	0.07174	0.12973
isi-bqfs (Dang (2005))	0.36369	0.06984	0.12526
BASELINE (Dang (2005))	0.27522	0.04025	0.08716

With manual inspection, the extracted sentences in our proposed method have more detailed information rather than just the exact answer. For instance, when the query is asking about benefits of drug legalization, proposed method catches long sentences having rich information regarding how drug legalize can reduce the drug-related violence. Person's name, numerical values makes the sentences more informative. Other two query-based methods have less query relevant ranking. In fact TF-IDF carries short and simple sentences just having same keywords with query.

Table 17 shows when multiple sentences are present in a query, our proposed method responds to all query sentences. Although all the needs present in the query are almost satisfied, the order of respond of each query sentence is not correct. We do not give much importance to the coherency issue as only a small part of DUC dataset contains such complex queries. In fact, it is also hard for human to read attentively when the number of needs are more in a query.

Table 14: Experimental results on DUC 2006 dataset

System ID	ROUGE-1	ROUGE-2	ROUGE-SU4
Proposed method	0.5679	0.1242	0.2181
QBUEM	.3876	.1131	.1583
MRC	-	0.1095	0.1614
TL-TranSum	-	0.1078	0.1585
OQDE	0.4015 (0.3957-0.41076)	0.0928 (0.0884-0.0972)	0.1479 (0.1440-0.1521)
Graph-based (Wan)	0.4101	0.0886	0.1420
Graph-based (Wan & Xiao)	0.4031	0.0851	0.1400
Graph-based (Wei, Li, Lu & He)	-	0.0899	0.1427
Support Vector Classification	- -	0.0834 (0.0793-0.0876)	0.1387 (0.1344-0.1428)
Ranking SVM	-	0.0890 (0.0852-0.0928)	0.1443 (0.1403-0.1477)
Regression	-	0.0926 (0.0833-0.0969)	0.1485 (0.1443-0.1525)
Columbia06 (Hoa (2006))	0.39862	0.08264	0.14012
OGL.OHSU06 (Hoa (2006))	0.38594	0.08408	0.13912
OnModer (Hoa (2006))	0.40488	0.08987	0.14755
ICL.SUM (Hoa (2006))	0.40440	0.08792	0.14486
UMich (Hoa (2006))	0.40206	0.08444	0.14483
TLR (Schilder and McInnes (2006))	0.39908	0.08576	0.14381
CCS06 (Hoa (2006))	0.38991	0.08679	0.14170
JKD (Hoa (2006))	0.38807	0.08707	0.14134
IITH.SUM (Hoa (2006))	0.40980	0.09505	0.15464
LIA.THALES (Favre et al. (2006))	0.39922	0.08700	0.14522
QSLK	0.4287	0.0968	0.1673
CTMSUM (Yang (2014))	0.4157	0.0968	0.1548
WAASUM (Canhasi and Kononenko (2014))	0.4238	0.0917	0.1671
Topical-N (Yang et al. (2013))	0.4010	0.0893	0.1459
Qs-MR (Wei et al. (2011))	0.4012	0.0914	0.1444
SVR (Ouyang et al. (2010))	0.4018	0.0926	0.1485
LEX (Huang et al. (2010))	0.4030	0.0913	0.1449
BASELINE (Hoa (2006))	0.30217	0.04947	0.09788

Table 15: Experimental results on DUC 2007 dataset

System ID	ROUGE-1	ROUGE-2	ROUGE-SU4
Proposed method	0.5735	0.1367	0.2371
QBUEM	.4125	.1211	.1692
MRC	-	0.1275	0.1792
TL-TranSum	0.4239	0.1298	0.1796
OQDE	0.4295 (0.4233-0.4355)	0.1163 (0.1119-0.1205)	0.1685 (0.1645-0.1723)
Graph-based (Wan)	0.4204	0.1030	0.1460
Graph-based (Wan & Xiao)	-	0.1123	0.1682
Graph-based (Wei, Li, Lu & He)	0.4211 (0.4152-0.4274)	0.1103 (0.1063-0.1144)	0.1628 (0.1588-0.1670)
Support Vector Classification	- -	0.1075 (0.1032-0.1120)	0.1616 (0.1573-0.1659)
Ranking SVM	0.4301 (0.4237-0.4365)	0.1175 (0.1134-0.1219)	0.1682 (0.1642-0.1725)
NIST baseline (Dang (2007))	0.3091	0.0599	0.1036

14 Error Analysis of Proposed System

It is not always seen that our system has significant performance for each of the query. For example, we take this query from DUC 2005 Dataset, “Identify and describe types of organized crime that crosses borders or involves more than one country. Name the countries involved. Also identify the perpetrators involved with each type of crime, including both individuals and organizations if possible.” Here, we have to find the name of involved countries. Our system is an extractive text summarization system and there is no such direct information present in the input text documents which can give the names of involved countries. Sometimes, we need to make our own sentence based on the information. Abstractive summary will help us in such situation. If the query is an indirect question, we have to rephrase the sentences. In that case, our system fails to achieve good performance. Another drawback of our system is that there is a lack of coherency among the selected sentences. We can find out the mentioned disadvantages in the following summary generated from our system for the above query: *In the 1980s the international spread of drugs (cocaine from Latin America, heroin from Turkey, central Asia and the south-east*

Asia), advances in telecommunications, liberalisation of banking, freer trade and the opening of eastern Europe provided an enormous stimulus to the growth of Italian organised crime. A report released earlier this year by Censis, the social research institute, said the annual turnover of organised crimes’ illicit and ‘licit’ activities was L19, 389bn. Other guesstimates have put the figure at over L35, 000bn. As Liliano Ferraro, a senior official with Italy’s ministry of justice, commented last week: ‘The organised crime syndicates have already held this kind of meeting, they just meet in a hotel in eastern Europe and divide the profits. Mr Freeh made it clear that he saw the burgeoning organised crime wave in Russia as being as much a threat to the US and the west as to Russia itself. ‘We have to go where the crime is and not do what we did with respect to Italian organised crime and that was wait for 50 years before we got involved,’ Mr Freeh said. Though his speech was a model of diplomatic balance, he stressed the many mistakes made by US law enforcement agencies in the original response to the growth of organised crime which ‘permitted the expansion of a powerful, well-organised crime syndicate’. Much of the trade, which is centred on the St Lawrence River and Indian reserves which straddle the US-Canada border, has fallen into the hands of

Table 16: Extracted sentences obtained for the query ‘What are the benefits drug legalization?’

Method Name	Extracted sentences for summary
Attsum (Cao et al. (2016))	It acknowledges that illegal drugs cannot be kept out of the country by tougher border control and interdiction measures.
	Much greater resources, derived from taxation of the drugs that are now illegal and untaxed and from the billions saved by not wasting money on more criminal- justice measures, must be devoted to drug treatment and drug prevention.
	As is the case with tobacco, legalizing marijuana, cocaine and heroin would not signify an endorsement of their use.
	The consumption and production of marijuana in the United States is on the decrease, and that criminalization costs society more in terms of increased law-enforcement-related costs and deprived revenues from taxes on pot than legalization would.
TF-IDF (Cao et al. (2016))	Drug prices have soared.
	Drug addicts are not welcome.
	How refreshing to have so much discourse on drugs and legalization.
	The only solution now is a controlled policy of drug legalization.
Proposed_Method	The panel, appointed by state agencies and the governor, found that anti-drug laws have been “manifestly unsuccessful in that we are now using more and a greater variety of drugs, legal and illegal.” Created by the Legislature in 1969 to study drug abuse and possible remedies, the panel recommended that the Legislature legalize cultivation of marijuana for personal use, consider decriminalizing other drugs on an individual basis and legalize possession of hypodermic needles.
	Ralph Lee White, a former Stockton councilman who was thrown out of office for bribing a voter, announced his candidacy for state controller on a platform of legalizing drugs.” We should legalize drugs because it would stop the killing between gangs,” White said at a Stockton news conference announcing his candidacy for the Democratic nomination, if incumbent Gray Davis runs for governor as expected.
	One notable aberration from the campaign’s homogeneous tone involves Democrat Abeles’ proposal that the use of drugs such as cocaine and methamphetamines be legalized, with the state regulating their production and distribution, as well as that of “legal drugs” such as liquor.
	Originally, drugs were legal, and there was no drug-related violence.

organised crime. THE ITALIAN authorities yesterday achieved a breakthrough in the fight against organised crime with the capture of Mr Salvatore ‘Toto’ Riina, acknowledged to be the boss of Cosa Nostra, the umbrella organisation of the Sicilian Mafia. Mr Riina’s arrest exemplifies the determination of the government of Mr Giuliano Amato to combat organised crime and break the long-standing protection given to the Mafia by elements of the Italian state and among the political parties. The son of a poor smallholder, Mr Riina has been involved in organised crime since his early teens and married into one of the traditional Mafia families of Corleone. Organised crime has traditionally been a ‘southern’ phenomenon, associated with three groups, each distinguished by clear geographical and historical

characteristics - Cosa Nostra, the Sicilian Mafia; the Camorra, based round Naples and the ‘ndrangheta of Calabria.

15 Statistical Significance Test

To statistically compare the performance of our proposed redundancy free query-based extractive text summary method with other summarization methods, we use a non-parametric Wilcoxon’s matched-pairs signed rank based statistical significance test to determine the significance of our results. The statistical significance test for independent samples has been conducted at the 5% significance level of the summarization results. Nine groups, corresponding to the nine widely used and cur-

Table 17: Extracted sentences obtained for the query ‘Why are wetlands important? Where are they threatened? What steps are being taken to preserve them? What frustrations and setbacks have there been?’

Method Name	Extracted sentences for summary
Attsum	Boparai also said that wetlands in many developing countries were vital to the sustenance of human beings, not just flora and fauna.
	EPA says that all water conservation projects, and agriculture and forestry development along China’s major rivers must be assessed in accordance with environmental protection standards, and that no projects will be allowed if they pose a threat to the environment.
	Finland has agreed to help central Chinas Hunan Province improve biodiversity protection, environmental education, sub-tropical forestry and wetlands protection, according to provincial officials.
	The EPA had sought as early 1993 to subject all development on wetlands to strict environmental review, but that approach was rejected by the courts, which ruled in favor of arguments made by developers and by the National Mining Association.
TF-IDF	Statistics on wetlands loss vary widely.
	Mitigation of any impact on wetlands by creating or enhancing other wetlands.
	The new regulations would cover about one-fourth of all wetlands.
	Now more and more people have recognized wetlands great ecological and economic potential and the conservation and utilization of wetlands has become an urgent task.
Proposed_Method	Xu said his Ministry will launch a drive to enhance the awareness of wetland protection among the public, establish a dynamic monitoring network and information system for China’s wetlands, conduct scientific research in wetland protection as well as proper utilization and management of wetlands resources.
	The new organization links the Asian Wetland Bureau, Wetlands for the Americas and the International Waterfowl and Wetlands Research Bureau.
	Zhen said, for quite a long time, many people believed that the wetlands were useless mud. Now more and more people have recognized wetlands’ great ecological and economic potential and the conservation and utilization of wetlands has become an urgent task.
	The four-day International Workshop on Wetland, sponsored by China’s Ministry of Forestry, Japan’s Environment Agency and Wetlands International will also arrange investigation of wetlands and water birds in Beidaihe.

rent methods: (1) LEX, (2) TMR, (3) SVR, (4) Topical-N, (5) QEMD, (6) Qs-MR, (7) CTMSUM, (8) WAA-SUM, (9) QSLK, have been created for data set. Two groups are compared at a time one corresponding to our proposed method and the other corresponding to some other method considered here. Each group consists of the ROUGE-1 and ROUGE-2 scores for the data set produced by each corresponding method. The median values of ROUGE-1 and ROUGE-2 scores of

each method for the data set are presented in Table 18. Table 18 shows that the median values of ROUGE-1 and ROUGE-2 for our proposed method on data set are better than that for the other methods. To establish that this goodness is statistically significant, Table 19 reports the P values produced by Wilcoxon’s matched-pairs signed rank test for comparison of two groups (one group corresponding to proposed method and another group corresponding to some other method) at

a time. As a null hypothesis, it is assumed that there are no significant differences between the median values of two groups. Whereas the alternative hypothesis is that there is significant difference in the median values of the two groups. It is clear from Table 8 that P values are much less than 0.05 (5% significance level). For example, the Wilcoxon’s matched-pairs signed rank test between our proposed method and WAASum for DUC 2006 provides a P value of 0.037 (ROUGE-1), which is very small. This is strong evidence against the null hypothesis, indicating that the better median values of the performance metrics produced by our proposed method is statistically significant and has not occurred by chance. Similar results are obtained for all other methods compared to our proposed method, establishing the significant superiority of the proposed method. From the statistical results, we observe that our proposed query-based extractive text summarization method significantly outperforms the other baseline summarization methods.

16 Complexity Analysis of the Query-based Extractive Text Summarization Using Sense-oriented Semantic Relatedness Measure technique

The time complexity of the proposed query-based text summarization method depends upon total number of sentences in the input text to be summarized and the number of words present in an individual sentence. Both of the counts will not be so high. The complete method for the proposed query-based text summarization method is presented in Algorithm 1. The complexity analysis is shown below.

Let, Total number of sentences in the text to be summarized is N , average number of words in a sentence is A , and each word has S number of senses.

The processing steps consumes $\mathcal{O}(\mathcal{N}\mathcal{A})$ time as each word of the text need to be visited.

Word sense disambiguation is done based on the words present in the sentence. If there are S senses for a word, then time requirement will be $\mathcal{O}(\mathcal{N}\mathcal{S}\mathcal{A})$.

To find query relevant sentences each sentence of the text is to be examined. Each sentence has on an average A word, and each word has S senses. Time complexity will be $\mathcal{O}(\mathcal{N}\mathcal{A}^2\mathcal{S}^2)$.

The k-means clustering step will require $\mathcal{O}(\mathcal{N})$ time, neglecting value of k , number of dimensions D , number of iterations I which are small.

The redundancy removal step will be of the order of $\mathcal{O}(\mathcal{N}^2\mathcal{A}^2)$ assuming that word sense disambiguation step stores the correct sense of each word.

Thus, overall time requirement is $\mathcal{O}(\mathcal{N}\mathcal{A}) + \mathcal{O}(\mathcal{N}\mathcal{S}\mathcal{A}) + \mathcal{O}(\mathcal{N}\mathcal{A}^2\mathcal{S}^2) + \mathcal{O}(\mathcal{N}^2\mathcal{A}^2)$. Values of A , Q and S can be neglected as they will be small constants. Thus, overall complexity becomes $\mathcal{O}(\mathcal{N}^2)$, where N is the number of sentences in the text.

17 Conclusion and Future Work

In this paper, we proposed a redundancy free query-based extractive text summarization method. Under unsupervised learning methods, the proposed method provides excellent extraction ability and better query-based summary quality even compared with some supervised methods. We also discuss query relevance performance with other query-based text summarization systems. We have also introduced a word sense disambiguation method for query-based text summarization. This method helps in finding the sense-oriented query relevance sentences on the basis of its meaning.

We have conducted extensive experiments on DUC query-based text summarization datasets. Proposed algorithm achieves competitive performance for all the best performing systems on DUC datasets and as well as with current state-of-the-art query-based text summarization systems. Method also finds those sentences which are relevance in terms of query sense. Literature survey states that existing query-based text summarization systems rarely provide query relevant sentences but our proposed method can find query-based sentences on the basis of word’s sense. The proposed WSD technique is also compared with other existing methods and finds its better performance. Our method can also find semantic relatedness score between those words which are not present in WordNet ontology.

Although the proposed method works better than many existing query-oriented text summarization methods, we can still improve its performance. In fact, a lot of potential for future work is seen based upon this research. As future work, we plan to identify between active and passive sentences which will help in minimizing redundant information and it will help in finding correct summary.

The over all multiple text document query-based text summarization method is affected by the evaluation process. DUC datasets only generate the summary with 250 words. That is one major drawback of text summarization evaluation system as it is much difficult to generate the summary for only 250 words which should match with human generated summary. For the redundancy and coherency issues, we have not found proper evaluation method through which importance of redundancy and coherency can be justified. Evaluation process is also affected by reference summary. Although

Table 18: Median values of methods on DUC 2006 dataset

System ID	ROUGE-1	ROUGE-2
Proposed method	0.5679	0.1242
TMR Tang et al. (2009)	0.6005	0.0695
QEMD Zhao et al. (2009)	0.5650	0.0691
QSLK Abdi et al. (2017)	0.6515	0.0769
CTMSUM Yang (2014)	0.5635	0.0709
WAASUM Canhasi and Kononenko (2014)	0.6170	0.0711
Topical-N Yang et al. (2013)	0.4370	0.0521
Qs-MR Wei et al. (2011)	0.5225	0.0608
SVR Ouyang et al. (2010)	0.5405	0.0668
LEX Huang et al. (2010)	0.5870	0.0649

Table 19: P values produced by Wilcoxon’s matched-pairs signed rank test by comparing proposed method with other methods in DUC 2006 dataset

LEX	TMR	SVR	Topical-N	QEMD	Qs-MR	CTMSUM	WAASum	QSLK
0.003	0.040	0.001	0.000	0.001	0.000	0.001	0.037	0.00
0.003	0.040	0.001	0.000	0.001	0.000	0.001	0.037	0.00

reference summary is produced by expert human, still accuracy can not expect fully from it.

In addition, we can implement our method to generic multi-document summarization in where abstractive summary can also be applied. Finally, to improve the linguistic quality of summaries, we need to do more research on how to process natural language on machine most effectively.

18 Compliance with Ethical Standards:

1. (In case of Funding) Funding: Not Applicable
2. Conflict of Interest: First Author Nazreena Rahman declares that she has no conflict of interest. Second Author Dr. Bhogeswar Borah declares that he has no conflict of interest.

3. (In case animals were involved) Ethical approval : This article does not contain any studies with animals performed by any of the authors.
4. (And/or in case humans were involved) Ethical approval: This article does not contain any studies with human participants performed by any of the authors.
5. (In case humans are involved) Informed consent: This article does not contain any studies with human participants performed by any of the authors.

References

- Abdi A, Idris N, Alguliyev RM, Aliguliyev RM (2017) Query-based multi-documents summarization using linguistic knowledge and content word expansion. *Soft Computing* 21(7):1785–1801
- Abu Nada AM, Alajrami E, Al-Saqqa AA, Abu-Naser SS (2020) Arabic text summarization using arabert model using extractive text summarization approach
- Afsharizadeh M, Ebrahimpour-Komleh H, Bagheri A (2018) Query-oriented text summarization using sentence extraction technique. In: 2018 4th international conference on web research (ICWR), IEEE, pp 128–132
- Afuan L, Ashari A, Suyanto Y (2019) A study: query expansion methods in information retrieval. In: *Journal of Physics: Conference Series*, IOP Publishing, vol 1367, p 012001
- Agirre E, Soroa A (2009) Personalizing pagerank for word sense disambiguation. In: *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, Association for Computational Linguistics, pp 33–41
- Agirre E, López de Lacalle O, Soroa A (2014) Random walks for knowledge-based word sense disambiguation. *Computational Linguistics* 40(1):57–84
- Akhtar N, Beg MS, Javed H (2019) Textrank enhanced topic model for query focussed text summarization. In: 2019 Twelfth International Conference on Contemporary Computing (IC3), IEEE, pp 1–6
- Allahyari M, Pouriyeh S, Assefi M, Safaei S, Trippe ED, Gutierrez JB, Kochut K (2017) Text summarization techniques: a brief survey. *arXiv preprint arXiv:170702268*
- Anderson JA (1995) *An introduction to neural networks*. MIT press
- Andrews NO, Fox EA (2007) Recent developments in document clustering. Tech. rep., Department of Computer Science, Virginia Polytechnic Institute & State
- Apidianaki M, Gong L (2015) Limsi: translations as source of indirect supervision for multilingual all-words sense disambiguation and entity linking. In: *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pp 298–302
- Bahdanau D, Cho K, Bengio Y (2014) Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:14090473*
- Balachandran PV, Xue D, Theiler J, Hogden J, Gubernatis JE, Lookman T (2018) Importance of feature selection in machine learning and adaptive design for materials. In: *Materials discovery and design*, Springer, pp 59–79
- Banerjee S, Pedersen T (2002) An adapted lesk algorithm for word sense disambiguation using wordnet. In: *International Conference on Intelligent Text Processing and Computational Linguistics*, Springer, pp 136–145
- Banerjee S, Pedersen T (2003) Extended gloss overlaps as a measure of semantic relatedness. In: *Ijcai*, vol 3, pp 805–810
- Basile P, Caputo A, Semeraro G (2014) An enhanced lesk word sense disambiguation algorithm through a distributional semantic model. In: *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pp 1591–1600
- Baxendale PB (1958) Machine-made index for technical literature: an experiment. *IBM Journal of Research and Development* 2(4):354–361
- Bayatmakou F, Mohebi A, Ahmadi A (2021) An interactive query-based approach for summarizing scientific documents. *Information Discovery and Delivery*
- Berger A, Mittal VO (2000) Query-relevant summarization using faqs. In: *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, Association for Computational Linguistics, pp 294–301
- Bird S, Klein E, Loper E (2009) *Natural language processing with Python: analyzing text with the natural language toolkit*. O'Reilly Media, Inc.
- Brin S, Page L (1998) The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems* 30(1-7):107–117
- Budanitsky A, Hirst G (2001) Semantic distance in wordnet: An experimental, application-oriented evaluation of five measures. In: *Workshop on WordNet and other lexical resources*, vol 2, pp 2–2
- Burges C, Shaked T, Renshaw E, Lazier A, Deeds M, Hamilton N, Hullender G (2005) Learning to rank using gradient descent. In: *Proceedings of the 22nd international conference on Machine learning*, ACM, pp 89–96
- Canhasi E, Kononenko I (2014) Weighted archetypal analysis of the multi-element graph for query-focused multi-document summarization. *Expert Systems with Applications* 41(2):535–543
- Cao Z, Wei F, Dong L, Li S, Zhou M (2015) Ranking with recursive neural networks and its application to multi-document summarization. In: *AAAI*, pp 2153–2159
- Cao Z, Li W, Li S, Wei F, Li Y (2016) Attsum: Joint learning of focusing and summarization with neural attention. *arXiv preprint arXiv:160400125*
- Carbonell JG, Goldstein J (1998) The use of mmr and diversity-based reranking for reordering documents

- and producing summaries
- Carpineto C, Romano G (2012) A survey of automatic query expansion in information retrieval. *Acm Computing Surveys (CSUR)* 44(1):1–50
- Chali Y, Joty SR (2008) Unsupervised approach for selecting sentences in query-based summarization. In: *FLAIRS Conference*, pp 47–52
- Charles WG (1988) The categorization of sentential contexts. *Journal of Psycholinguistic Research* 17(5):403–411
- Cheng J, Lapata M (2016) Neural summarization by extracting sentences and words. *arXiv preprint arXiv:160307252*
- Damova M, Koychev I (2010) Query-based summarization: A survey
- Dang HT (2005) Overview of duc 2005. In: *Proceedings of the document understanding conference*, vol 2005, pp 1–12
- Dang HT (2007) Document understanding conference duc 2007. In: *Document Understanding Conference*
- Denoyer L, Gallinari P (2006) The wikipedia xml corpus. In: *International Workshop of the Initiative for the Evaluation of XML Retrieval*, Springer, pp 12–19
- Devlin J, Chang MW, Lee K, Toutanova K (2018) Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:181004805*
- Dongsuk O, Kwon S, Kim K, Ko Y (2018) Word sense disambiguation based on word similarity calculation using word vector representation from a knowledge-based graph. In: *Proceedings of the 27th International Conference on Computational Linguistics*, pp 2704–2714
- Duraiswamy K (2014) An approach for text summarization using deep learning algorithm
- Edmundson HP (1969) New methods in automatic extracting. *Journal of the ACM (JACM)* 16(2):264–285
- Erkan G, Radev DR (2004) Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research* 22:457–479
- Evans DK, Klavans JL, McKeown KR (2004) Columbia newsblaster: Multilingual news summarization on the web. In: *Demonstration Papers at HLT-NAACL 2004*, pp 1–4
- Favre B, Bechet F, Bellot P, Boudin F, El-Beze M, Gillard L, Lapalme G, Torres-Moreno JM (2006) The lia-thales summarization system at duc-2006. In: *Proceedings of document understanding conference (DUC-2006)*, New York, USA
- Filatova E, Hatzivassiloglou V (2004) A formal model for information selection in multi-sentence text extraction. In: *Proceedings of the 20th international conference on Computational Linguistics, Association for Computational Linguistics*, p 397
- Filippova K, Alfonseca E, Colmenares CA, Kaiser L, Vinyals O (2015) Sentence compression by deletion with lstms. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp 360–368
- Goldstein J, Mittal V, Carbonell J, Kantrowitz M (2000) Multi-document summarization by sentence extraction. In: *Proceedings of the 2000 NAACL-ANLP Workshop on Automatic summarization, Association for Computational Linguistics*, pp 40–48
- Gupta M, Agrawal P (2020) Compression of deep learning models for text: A survey. *arXiv preprint arXiv:200805221*
- Hadiwinoto C, Ng HT, Gan WC (2019) Improved word sense disambiguation using pre-trained contextualized word representations. *arXiv preprint arXiv:191000194*
- Han KS, Baek DH, Rim HC (2000) Automatic text summarization based on relevance feedback with query splitting. In: *Proceedings of the fifth international workshop on on Information retrieval with Asian languages*, pp 201–202
- Haveliwala TH (2002) Topic-sensitive pagerank. In: *Proceedings of the 11th international conference on World Wide Web, ACM*, pp 517–526
- Hirst G, St-Onge D, et al. (1998) Lexical chains as representations of context for the detection and correction of malapropisms. *WordNet: An electronic lexical database* 305:305–332
- Hoa T (2006) Overview of duc 2006. In: *Document Understanding Conference*
- Hovy E, Lin CY (1998) Automated text summarization and the summarist system. In: *Proceedings of a workshop on held at Baltimore, Maryland: October 13-15, 1998, Association for Computational Linguistics*, pp 197–214
- Huang L, He Y, Wei F, Li W (2010) Modeling document summarization as multi-objective optimization. In: *Intelligent Information Technology and Security Informatics (IITSI), 2010 Third International Symposium on, IEEE*, pp 382–386
- Iacobacci I, Pilehvar MT, Navigli R (2016) Embeddings for word sense disambiguation: An evaluation study. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol 1, pp 897–907
- Jiang JJ, Conrath DW (1997) Semantic similarity based on corpus statistics and lexical taxonomy. *arXiv preprint cmp-lg/9709008*
- Joachims T (2002) Optimizing search engines using clickthrough data. In: *Proceedings of the eighth ACM SIGKDD international conference on Knowledge dis-*

- covery and data mining, ACM, pp 133–142
- Jurafsky D, Martin JH (2014) Speech and language processing, vol 3. Pearson London
- Kågebäck M, Salomonsson H (2016) Word sense disambiguation using a bidirectional lstm. arXiv preprint arXiv:160603568
- Kutuzov A, Kuzmenko E (2019) To lemmatize or not to lemmatize: How word normalisation affects elmo performance in word sense disambiguation. arXiv preprint arXiv:190903135
- Lamsiyah S, Mahdaouy AE, Ouatik SEA, Espinasse B (2021) Unsupervised extractive multi-document summarization method based on transfer learning from bert multi-task fine-tuning. *Journal of Information Science* p 0165551521990616
- Leacock C, Chodorow M (1998) Combining local context and wordnet similarity for word sense identification. *WordNet: An electronic lexical database* 49(2):265–283
- Lepikhin D, Lee H, Xu Y, Chen D, Firat O, Huang Y, Krikun M, Shazeer N, Chen Z (2020) Gshard: Scaling giant models with conditional computation and automatic sharding. arXiv preprint arXiv:200616668
- Lesk M (1986) Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In: *Proceedings of the 5th annual international conference on Systems documentation*, ACM, pp 24–26
- Li J, Li S (2013) A novel feature-based bayesian model for query focused multi-document summarization. *Transactions of the Association for Computational Linguistics* 1:89–98
- Li Q, Feng L, Pei J, Wang XS, Zhou X, Zhu Q (2009) *Advances in Data and Web Management: Joint International Conferences, APWeb/WAIM 2009*, Suzhou, China, April 2-4, 2009, *Proceedings*, vol 5446. Springer
- Li W, Li W, Li B, Chen Q, Wu M (2005) The hong kong polytechnic university at duc 2005. In: *Proceedings of Document Understanding Conferences*, Citeseer
- Li Y, Bandar ZA, McLean D (2003) An approach for measuring semantic similarity between words using multiple information sources. *IEEE Transactions on knowledge and data engineering* 15(4):871–882
- Lin CY (2004) Rouge: A package for automatic evaluation of summaries. In: *Text summarization branches out: Proceedings of the ACL-04 workshop*, Barcelona, Spain, vol 8
- Liu X, He P, Chen W, Gao J (2019) Multi-task deep neural networks for natural language understanding. arXiv preprint arXiv:190111504
- Lloret E, Palomar M (2013) Tackling redundancy in text summarization through different levels of language analysis. *Computer Standards & Interfaces* 35(5):507–518
- Lopyrev K (2015) Generating news headlines with recurrent neural networks. arXiv preprint arXiv:151201712
- Luhn HP (1958) The automatic creation of literature abstracts. *IBM Journal of research and development* 2(2):159–165
- Mahmud A (2020) Query-based summarization using reinforcement learning and transformer model. University of Lethbridge (Canada)
- Maulud DH, Zeebaree SR, Jacksi K, Sadeeq MAM, Sharif KH (2021) State of art for semantic analysis of natural language processing. *Qubahan Academic Journal* 1(2):21–28
- McDonald R (2007) A study of global inference algorithms in multi-document summarization. In: *European Conference on Information Retrieval*, Springer, pp 557–564
- McKeown KR, Radev DR (2000) Collocations. *Handbook of Natural Language Processing* Marcel Dekker
- McRae K, Khalkhali S, Hare M (2012) Semantic and associative relations in adolescents and young adults: Examining a tenuous dichotomy
- Mei JP, Chen L (2012) Sumcr: A new subtopic-based extractive approach for text summarization. *Knowledge and information systems* 31(3):527–545
- Melamud O, Goldberger J, Dagan I (2016) context2vec: Learning generic context embedding with bidirectional lstm. In: *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pp 51–61
- Meng L, Huang R, Gu J (2013) A review of semantic similarity measures in wordnet. *International Journal of Hybrid Information Technology* 6(1):1–12
- Mihalcea R, Tarau P (2004) Textrank: Bringing order into text. In: *Proceedings of the 2004 conference on empirical methods in natural language processing*, pp 404–411
- Miller D (2019) Leveraging bert for extractive text summarization on lectures. arXiv preprint arXiv:190604165
- Mohamed AA, Rajasekaran S (2006) Improving query-based summarization using document graphs. In: *Signal Processing and Information Technology, 2006 IEEE International Symposium on*, IEEE, pp 408–410
- Moro A, Raganato A, Navigli R (2014) Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics* 2:231–244
- Murarka S, Singhal A (2020) Query-based single document summarization using hybrid semantic and

- graph-based approach. In: 2020 International Conference on Advances in Computing, Communication & Materials (ICACCM), IEEE, pp 330–335
- Nallapati R, Xiang B, Zhou B (2016a) Sequence-to-sequence rnns for text summarization
- Nallapati R, Zhou B, Gulcehre C, Xiang B, et al. (2016b) Abstractive text summarization using sequence-to-sequence rnns and beyond. arXiv preprint arXiv:160206023
- Nallapati R, Zhai F, Zhou B (2017) Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In: Thirty-First AAAI Conference on Artificial Intelligence
- Navigli R, Ponzetto SP (2012) Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence* 193:217–250
- Oufaida H, Nouali O, Blache P (2014) Minimum redundancy and maximum relevance for single and multi-document arabic text summarization. *Journal of King Saud University-Computer and Information Sciences* 26(4):450–461
- Ouyang Y, Li W, Li S, Lu Q (2010) Intertopic information mining for query-based summarization. *Journal of the Association for Information Science and Technology* 61(5):1062–1072
- Ouyang Y, Li W, Li S, Lu Q (2011) Applying regression models to query-focused multi-document summarization. *Information Processing & Management* 47(2):227–237
- Park S, Lee JH, Ahn CM, Park TS, Song JW, Kim DH (2006) Query-based text summarization using cosine similarity and nmf. In: Proceedings of the Korea Information Processing Society Conference, Korea Information Processing Society, pp 473–476
- Pasini T, Navigli R (2017) Train-o-matic: Large-scale supervised word sense disambiguation in multiple languages without manual training data. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp 78–88
- Patil K, Brazdil P (2007) Text summarization: Using centrality in the pathfinder network. *Int J Comput Sci Inform Syst [online]* 2(18-32)
- Patwardhan S, Banerjee S, Pedersen T (2003) Using measures of semantic relatedness for word sense disambiguation. In: *CICLing*, Springer, vol 2588, pp 241–257
- Pedersen T, Patwardhan S, Michelizzi J (2004) Wordnet:: Similarity: measuring the relatedness of concepts. In: *Demonstration papers at HLT-NAACL 2004*, Association for Computational Linguistics, pp 38–41
- Peng M, Gao B, Zhu J, Huang J, Yuan M, Li F (2016) High quality information extraction and query-oriented summarization for automatic query-reply in social network. *Expert Systems with Applications* 44:92–101
- Perkins J (2014) Python 3 text processing with NLTK 3 cookbook. Packt Publishing Ltd
- Plaut DC (1995) Semantic and associative priming in a distributed attractor network. In: *Proceedings of the 17th annual conference of the cognitive science society*, Pittsburgh, PA, vol 17, pp 37–42
- Prasad RS, Kulkarni U, Prasad JR (2009) Connectionist approach to generic text summarization. *World Academy of Science, Engineering and Technology* 55
- Rada R, Mili H, Bicknell E, Blettner M (1989) Development and application of a metric on semantic nets. *IEEE transactions on systems, man, and cybernetics* 19(1):17–30
- Radford A, Wu J, Child R, Luan D, Amodei D, Sutskever I, et al. (2019) Language models are unsupervised multitask learners. *OpenAI blog* 1(8):9
- Raffel C, Shazeer N, Roberts A, Lee K, Narang S, Matena M, Zhou Y, Li W, Liu PJ (2019) Exploring the limits of transfer learning with a unified text-to-text transformer. arXiv preprint arXiv:1910.10683
- Raganato A, Bovi CD, Navigli R (2017a) Neural sequence learning models for word sense disambiguation. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp 1156–1167
- Raganato A, Camacho-Collados J, Navigli R (2017b) Word sense disambiguation: A unified evaluation framework and empirical comparison. In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, vol 1, pp 99–110
- Ramos J, et al. (2003) Using tf-idf to determine word relevance in document queries. In: *Proceedings of the first instructional conference on machine learning*, Citeseer, vol 242, pp 29–48
- Rastogi H (2020) Automatic text summarization using soft-cosine similarity and centrality measures. PhD thesis
- Resnik P (1995) Using information content to evaluate semantic similarity in a taxonomy. arXiv preprint cmp-lg/9511007
- RISNE V, SIITOVA A (2019) Text summarization using transfer learnin: Extractive and abstractive summarization using bert and gpt-2 on news and podcast data
- Rosset C (2020) Turing-nlg: A 17-billion-parameter language model by microsoft. *Microsoft Blog* 1:2

- Rothe S, Schütze H (2015) Autoextend: Extending word embeddings to embeddings for synsets and lexemes. arXiv preprint arXiv:150701127
- Rush AM, Chopra S, Weston J (2015) A neural attention model for abstractive sentence summarization. arXiv preprint arXiv:150900685
- Schilder F, Kondadadi R (2008) Fastsum: fast and accurate query-based multi-document summarization. In: Proceedings of the 46th annual meeting of the association for computational linguistics on human language technologies: Short papers, Association for Computational Linguistics, pp 205–208
- Schilder F, McInnes BT (2006) Tlr at duc 2006: approximate tree similarity and a new evaluation regime. In: Proc. of DUC
- Shardan R, Kulkarni U (2010) Implementation and evaluation of evolutionary connectionist approaches to automated text summarization
- Sparck Jones K (1972) A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation* 28(1):11–21
- Srikanth A, Umasankar AS, Thanu S, Nirmala SJ (2020) Extractive text summarization using dynamic clustering and co-reference on bert. In: 2020 5th International Conference on Computing, Communication and Security (ICCCS), IEEE, pp 1–5
- Strubell E, Ganesh A, McCallum A (2019) Energy and policy considerations for deep learning in nlp. arXiv preprint arXiv:190602243
- Strzalkowski T, Wang J, Wise GB (1998) Summarization-based query expansion in information retrieval. In: COLING 1998 Volume 2: The 17th International Conference on Computational Linguistics
- Svore K, Vanderwende L, Burges C (2007) Enhancing single-document summarization by combining ranknet and third-party sources. In: Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)
- Taghipour K, Ng HT (2015) One million sense-tagged instances for word sense disambiguation and induction. In: Proceedings of the nineteenth conference on computational natural language learning, pp 338–344
- Tang J, Yao L, Chen D (2009) Multi-topic based query-oriented summarization. In: Proceedings of the 2009 SIAM International Conference on Data Mining, SIAM, pp 1148–1159
- Thakkar KS, Dharaskar RV, Chandak M (2010) Graph-based algorithms for text summarization. In: 2010 3rd International Conference on Emerging Trends in Engineering and Technology, IEEE, pp 516–519
- Tong H, Faloutsos C, Pan JY (2006) Fast random walk with restart and its applications. In: Sixth International Conference on Data Mining (ICDM'06), IEEE, pp 613–622
- Tripodi R, Pelillo M (2017) A game-theoretic approach to word sense disambiguation. *Computational Linguistics* 43(1):31–70
- Ulrich J, Murray G, Carenini G (2008) A publicly available annotated corpus for supervised email summarization. In: Proc. of aaai email-2008 workshop, chicago, usa
- Van Lierde H, Chow TW (2019a) Learning with fuzzy hypergraphs: A topical approach to query-oriented text summarization. *Information Sciences* 496:212–224
- Van Lierde H, Chow TW (2019b) Query-oriented text summarization based on hypergraph transversals. *Information Processing & Management* 56(4):1317–1338
- Vapnik V (2013) The nature of statistical learning theory. Springer science & business media
- Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. In: Advances in neural information processing systems, pp 5998–6008
- Verma S, Nidhi V (2017) Extractive summarization using deep learning. arXiv preprint arXiv:170804439
- Wan X (2009) Topic analysis for topic-focused multi-document summarization. In: Proceedings of the 18th ACM conference on Information and knowledge management, ACM, pp 1609–1612
- Wan X, Xiao J (2009) Graph-based multi-modality learning for topic-focused multi-document summarization. In: Twenty-First International Joint Conference on Artificial Intelligence
- Wan X, Yang J, Xiao J (2007) Manifold-ranking based topic-focused multi-document summarization. In: IJ-CAI, vol 7, pp 2903–2908
- Wang S, Li W, Wang F, Deng H (2010) A survey on automatic summarization. In: 2010 International Forum on Information Technology and Applications, IEEE, vol 1, pp 193–196
- Wei F, Li W, Lu Q, He Y (2010) A document-sensitive graph model for multi-document summarization. *Knowledge and information systems* 22(2):245–259
- Wei F, Li W, He Y (2011) Document-aware graph models for query-oriented multi-document summarization. In: Multimedia Analysis, Processing and Communications, Springer, pp 655–678
- Weissenborn D, Hennig L, Xu F, Uszkoreit H (2015) Multi-objective optimization for the joint disambiguation of nouns and named entities. In: Proceedings of the 53rd Annual Meeting of the Association

- for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), vol 1, pp 596–605
- Widyassari AP, Rustad S, Shidik GF, Noersasongko E, Syukur A, Affandy A, et al. (2020) Review of automatic text summarization techniques & methods. *Journal of King Saud University-Computer and Information Sciences*
- Wiedemann G, Remus S, Chawla A, Biemann C (2019) Does bert make any sense? interpretable word sense disambiguation with contextualized embeddings. *arXiv preprint arXiv:190910430*
- Wiemer-Hastings P (2000) Adding syntactic information to lsa. In: *Proceedings of the Annual Meeting of the Cognitive Science Society*, vol 22
- Wu HC, Luk RWP, Wong KF, Kwok KL (2008) Interpreting tf-idf term weights as making relevance decisions. *ACM Transactions on Information Systems (TOIS)* 26(3):1–37
- Wu Z, Palmer M (1994) Verbs semantics and lexical selection. In: *Proceedings of the 32nd annual meeting on Association for Computational Linguistics, Association for Computational Linguistics*, pp 133–138
- Yadav A, Patel A, Shah M (2021) A comprehensive review on resolving ambiguities in natural language processing. *AI Open* 2:85–92
- Yang G (2014) A novel contextual topic model for query-focused multi-document summarization. In: *Tools with Artificial Intelligence (ICTAI), 2014 IEEE 26th International Conference on, IEEE*, pp 576–583
- Yang G, Wen D, Sutinen E, et al. (2013) A contextual query expansion based multi-document summarizer for smart learning. In: *Signal-Image Technology & Internet-Based Systems (SITIS), 2013 International Conference on, IEEE*, pp 1010–1016
- Yang Z, Dai Z, Yang Y, Carbonell J, Salakhutdinov RR, Le QV (2019) Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems* 32
- Ye X, Wei H (2008) Query-based summarization for search lists. In: *Knowledge Discovery and Data Mining, 2008. WKDD 2008. First International Workshop on, IEEE*, pp 330–333
- Yeh JY, Ke HR, Yang WP, Meng IH (2005) Text summarization using a trainable summarizer and latent semantic analysis. *Information processing & management* 41(1):75–95
- Yousefi-Azar M, Hamey L (2017) Text summarization using unsupervised deep learning. *Expert Systems with Applications* 68:93–105
- Yu S, Su J, Li P, Wang H (2016) Towards high performance text mining: a textrank-based method for automatic text summarization. *International Journal of Grid and High Performance Computing (IJGHPC)* 8(2):58–75
- Yuan D, Richardson J, Doherty R, Evans C, Altendorf E (2016) Semi-supervised word sense disambiguation with neural models. *arXiv preprint arXiv:160307012*
- Zamanifar A, Minaei-Bidgoli B, Sharifi M (2008) A new hybrid farsi text summarization technique based on term co-occurrence and conceptual property of the text. In: *2008 Ninth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, IEEE*, pp 635–639
- Zhang B, Li H, Liu Y, Ji L, Xi W, Fan W, Chen Z, Ma WY (2005) Improving web search results using affinity graph. In: *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, ACM*, pp 504–511
- Zhao L, Wu L, Huang X (2009) Using query expansion in graph-based approach for query-focused multi-document summarization. *Information processing & management* 45(1):35–41
- Zhao X, Tang J (2010) Query-focused summarization based on genetic algorithm. In: *2010 International Conference on Measuring Technology and Mechatronics Automation, IEEE*, vol 2, pp 968–971
- Zhong Sh, Liu Y, Li B, Long J (2015) Query-oriented unsupervised multi-document summarization via deep learning model. *Expert Systems with Applications* 42(21):8146–8155
- Zhong Z, Ng HT (2010) It makes sense: A wide-coverage word sense disambiguation system for free text. *Proceedings of the ACL 2010 system demonstrations* pp 78–83