Improvement of query-based text summarization using word sense disambiguation

Nazreena Rahman¹ · Bhogeswar Borah²

Received: 2 December 2018 / Accepted: 24 June 2019 / Published online: 20 July 2019
© The Author(s) 2019

Abstract

In this paper, a query-based text summarization method is proposed based on common sense knowledge and word sense disambiguation. Common sense knowledge is integrated here by expanding the query terms. It helps in extracting main sentences from text document according to the query. Query-based text summarization finds semantic relatedness score between query and input text document for extracting sentences. The drawback with current methods is that while finding semantic relatedness between input text and query, in general they do not consider the sense of the words present in the input text sentences and the query. However, this particular method can enhance the summary quality as it finds the correct sense of each word of a sentence with respect to the context of the sentence. The correct sense for each word is being used while finding semantic relatedness between input text and query. To remove similar sentences from summary, similarity measure is computed among the selected sentences. Experimental result shows better performance than many baseline systems.

Keywords Common sense knowledge · Expanding the query terms · Query-based text summarization · Semantic relatedness · Word sense disambiguation

Introduction

With the tremendous growth of textual information, text summarization helps in finding the essential information in gist form. Text summarization contributes in retrieval of important information from a large textual data and also reduces the size of the text. Hence, it helps in acquiring information in a short period of time. Additionally, query-based text summarization is used particularly for retrieving information in shorter form for a specific question. It is different from question answering system. Question answering system pinpoints an answer according to the question. Sometimes, users need and appreciate detailed answer to get more information than the exact answer. Therefore, query-based text summarization produces detailed answer which will contain necessary infor-

mation related to the query to acquire knowledge quickly. To find out well-defined relevant summary of a query, it is better to find semantically related words with query words.

Semantic relatedness measure is important to find semantically related words with query from the input text sentences to generate query-based text summarization. Semantic relatedness means sentences which are related with query on the basis of its meaning. While finding semantic relatedness score, it is important to find the senses of a word. A word will have unique sense with respect to a sentence. For example, “Sachin plays with the bat”: The word bat refers to a wooden bat and “the bat is the only mammal that can fly”; here, the word “bat” refers to an animal. This type of word is known as polysemy. Polysemy is a word having multiple meaning. Homograph words also have different senses. Homographs means a word having different meaning but accidentally they have same spelling. Example of homographs are “rose”, this rose can be used for past tense of the word “rise” or it can be used for a flower. Therefore, finding correct sense of a word is important so that appropriate sense can be used, while calculating the semantic relatedness between words.

Adding new and related words to the query give information-rich summary. It helps in finding more related sen-

✉ Nazreena Rahman
nazreena.rehman@gmail.com
Bhogeswar Borah
bgb@tezu.ernet.in

¹ Department of Computer Science and Engineering, Kaziranga University, Jorhat 785006, India

² Department of Computer Science and Engineering, Tezpur University, Tezpur 784028, India

tences with query. To find new concepts and relations of query words, a semantic network of common sense knowledge can be used. Concept means words or phrases. Semantic network consists of a large set of concepts where each concept is related with other concepts in a graphical way. A measure for finding similarity between words is also presented where semantic relatedness and collocation is used in a linear equation. In this paper, the proposed method investigates mainly three issues : (1) expansion of query using common sense knowledge, (2) detection of appropriate sense of a word with respect to the sentence and (3) calculating relatedness between words. Finally, redundancy is checked by finding common words between two sentences so that final text comes with more summarized form.

The remainder of the paper is as follows. Section 2 briefly reviews the literature on works done particularly in query-based text summarization, Sect. 3 describes about the overview of semantic network. Section 4 explains the proposed work. Section 5 discusses the experiments and results of proposed system and finally Sect. 6 includes conclusion and future work.

Literature review

Many supervised and unsupervised techniques are implemented to find out the required sentences or paragraph for creating query-based summary [24]. Here, some recently proposed methods have been explained to know the current works.

Ye et al. [32] proposed a summarization system for identification of highly query related and information density based sentences. Here, natural language processing and information retrieval techniques are being combined together to get improved result. To pick out only the relevant text documents, an existing mapping algorithm is used for extracting senses of terms. Word overlap feature has been adopted to pick up power of correlation with input query. Here, Okapi is used to compute weight of each common word. Their method finds the informativeness power between the sentences by applying a diffusion process. It helps to get a more accurate undirected graph. It has been assumed that inter-document links are more important than intra-document links for finding richer information. Finally, both features are added with a weighted linear combination. Maximal Marginal Relevance (MMR) is used to reduce redundancy among highly scored sentences. For the final step, summary can be generated by selecting that document which contains highest number of summary sentences. The order is decided based on the similarity between them. This method performs better for both retrieval of documents concentrating to one subject and also for many subjects being related to query.

Mohamed et al. [15] used document graphs for query-based summarization purpose. They proposed three different approaches. The method uses document graph (DG) for representation of a text document. A DG graph is a directed graph with relations among them. Two kinds of nodes, concept/entity nodes and relation nodes, are there in DG graph. Here, two kinds of relations “isa” and “related_to” are used as relation nodes. The centric graph is constructed using all the document graphs. This method constructs general, medium and high focus centric graphs. According to their first approach, modification is done at the time of centric graph generation. First approach says that, whenever a relation is added to the centric graph, it must belong to one of the three mentioned centric graph models or should contain at least one concept/entity that is present in the query. This modified DG summarizer can give better result as it includes only those relations which are related to query. In their second approach, DG graphs are generated for each input document and query. Method finds similarity score between sentences and query, and best sentences are being collected and added. Third approach is just similar to the second one by adding restriction on length of summary. Comparison is done among three approaches and finds second approach performs better than the other two approaches.

Ouyang et al. [19] used regression model for ranking and extracting sentences in query-based text summarization. Support Vector Regression (SVR) model has been used here to find the importance of a sentence in the text document. In fact, they use a set of pre-defined features. In this feature-based extractive framework, three features are query dependent and four are query independent. Word matching feature, semantic matching feature and named entity matching feature are considered as under query-dependent feature. Word TF-IDF feature, named entity feature, stop-word penalty feature and sentence position feature are used as query-independent features. To synthesize effects of all the above features, a composite function is used. The method calculates score of the sentences. To calculate the score, regression model is trained by constructing “pseudo” training data. The method assigns a “nearly true” importance score to the training data based on human summaries. Finally, a mapping function is taken from a set of pre-defined sentence features to this “nearly true” sentence important score. For selecting redundant-free sentences, MMR (Maximal Marginal Relevance) technique is applied. This learned function is used for further predicting scores of test data. They examined the effectiveness of this proposed method and gave better result over classification model and learning-to-rank model.

Abdi et al. [1] used linguistic knowledge and content word expansion technique to find important information from text documents on the basis of users’ requirements. The proposed method (QSLK) uses both word order and semantic similarity score for finding query-based text summarization. Nouns,

main verbs, adjectives and adverbs are considered as content words. In this method, word set is created between pair of sentences using distinct words present in both sentences. This word set is being used for semantic and syntactic vector creation. Their method uses Lin similarity for content word expansion. This expansion is used for creation of semantic and syntactic vector. Semantic similarity score is computed using cosine similarity between two semantic vectors. Similarly, syntactic vector is created by placing most semantically similar index position of the corresponding word. Syntactic vector helps in finding word order similarity between sentences. Total similarity between pair of sentences is calculated using a linear equation of semantic and word order similarity score. Ranking of the sentences is done using a combination model. Finally, a graph-based ranking model is used to remove redundancy of selected highly ranked sentences. The performance of QSLK is compared with the best methods participated in Document Understanding Conference (2006) and showed better results.

It is also seen that the presence of ambiguity in a word makes it difficult to find correct semantic relatedness between two sentences. In fact, finding correct sense of content words is quite essential while calculating relatedness between words. Many researchers proposed different methods to disambiguate word sense. Lesk [10] presented a disambiguation method. This method can disambiguate a word present in a short phrases. In this method, definition of each sense of a word in the phrase is compared with the definition of every other word present in the same phrase. Lesk algorithm is used by Plaza et al. [23] to improve text summarization. Further, Lesk method is modified and proposed one improved method named as adapted Lesk method by Banerjee et al. [2]. Here, sense is calculated by finding information about that target word along with the immediate neighboring words present in left and right side of the target word. It is important to mention that semantic similarity and semantic relatedness both are not same. We can say ice cream and spoon both are semantically related but they are not semantically same. Semantic similarity is strictly calculated at a context level.

From the above literature survey, it is found that many researchers are trying to find relevant query-based summary. Though they work on it constantly, but we can see that, semantic conceptual relatedness based on word sense disambiguation can not incorporate efficiently for query-based text summarization purpose. In our method, we try to find conceptual common sense semantic relatedness score by finding its proper sense.

Overview of semantic network

Our query-based text summarization is based on common sense knowledge. This knowledge is based on most general

and widely applicable facts and information that an ordinary person supposed to know. It is better to add common sense knowledge to text summarization. A semantic network of common sense knowledge can be used with the help of ConceptNet [12] to find common sense concepts and relations by expanding the query terms. This semantic network is based on OMCS (Open Mind Common Sense) database. OMCS is developed in 1999. A lot of changes is seen in ConceptNet since 2004. By combining the best of both CyC (framework for common sense knowledge created by Cycorp Company in 1984) like relations and WordNet (a lexical resource) like structure, this ConceptNet is considered as a semi-structured natural language network. The current corpus sources of ConceptNet are OMCS, English Wikipedia, English Wikionary, DBpedia, WordNet, Even Games, etc. It has 1.6 million assertions of common sense knowledge with more than 300,000 concepts.

Semantic network represents semantic relations between concepts in a network where inter-connected nodes and paths are present. Object or concept represents nodes and path gives the different relations between the two concepts and objects. The semantic network carries spatial, physical, social, temporal, and psychological details of every one's day-to-day life [3]. Different relations with different concepts are present in semantic network. We can extract human-like common sense knowledge by simply integrating semantic network with natural language processing. Semantic network contains richer semantic relations where more importance is given to the concepts rather than words. Semantic network contains assertions and concepts. Assertion can be represented by edge. Sometimes, multiple edges specify same assertion. It is represented as a hypergraph where each edge represents a relation between two concepts. These concepts are semi-structured English fragments and can be used for query expansion and semantic similarity measure. Some relation types in semantic network are given in this following Table 1. Semantic network also gives semantically related terms for compound concepts. Table 2 gives different assertions of a semantic relation “Used For” for the compound word “buy food”.

To find the related sentences according to the query, WordNet [14] ontology is used, where different relations are used to find semantic relatedness score between input text and query. WordNet is a lexical dictionary where a number of relations are present between two content words. Semantic relatedness measure defines how much two words are related. Content words (noun, verb, adverb and adjective) are used to find relatedness score as they carry more relevant information. Two words can be said as semantically related if there exists any relation between them. Semantic relatedness uses following Table 3 relations.

Table 1 Example of relation types in semantic network

Relation-type	Example
<i>Capable Of</i>	accident– <i>Capable Of</i> → cause injury
<i>At Location</i>	gym– <i>At Location</i> → school
<i>Derived From</i>	coach– <i>Derived From</i> → train
<i>Has Prerequisite</i>	watch movie– <i>Has Prerequisite</i> → buy ticket
<i>Motivated By Goal</i>	go to school– <i>Motivated By Goal</i> → gain knowledge
<i>Related To</i>	rea– <i>Related To</i> → gain knowledge
<i>Has First Sub event</i>	eat– <i>Has First Sub event</i> → open your mouth
<i>Is A</i>	woman– <i>Is A</i> → human
<i>Has Last Sub event</i>	eat– <i>Has Last Sub event</i> → swallow
<i>Location Of Action</i>	learn– <i>Location Of Action</i> → school
<i>Made Of</i>	house– <i>Made Of</i> → stone
<i>Has Context</i>	civil law– <i>Has Context</i> → law
<i>Has Property</i>	eat– <i>Has Property</i> → satisfy
<i>Causes</i>	eat– <i>Causes</i> → gain weight
<i>Part Of</i>	garden– <i>Part Of</i> → house
<i>Subevent Of</i>	eat– <i>Sub event Of</i> → chew
<i>Used For</i>	house– <i>Used For</i> → shelter

Proposed query-based text summarization method using word sense disambiguation

In query-based text summarization, we find the summarized text according to query. The inputs are the query and text documents, and the output is the summarized text. Summarization picks those sentences which are semantically related with the query sentence. Semantic relatedness score is calculated between a query sentence and an input text sentence. To find the score of each input text sentence, semantic relatedness score is computed for each sentence with the query. Content word always carry important information in a sentence. We consider only the content words present in the query and input text sentences. While picking sentences for summary, there is a high probability that same information might be conveyed by multiple sentences. Redundant sen-

Table 2 Semantic relation “Used For” for the compound word “buy food”

go to market– <i>Used For</i> → buy food
store food– <i>Used For</i> → buy food
delicatessen– <i>Used For</i> → buy food
buy food– <i>Used For</i> → prepare meal
buy food– <i>Used For</i> → consumer
eatery– <i>Used For</i> → buy food
super market– <i>Used For</i> → buy food
buy food– <i>Used For</i> → stock pantry
buy food– <i>Used For</i> → spend money
buy food– <i>Used For</i> → put food on a table
buy food– <i>Used For</i> → feed homeless
buy food– <i>Used For</i> → keep cupboard stock
buy food– <i>Used For</i> → prepare for pot luck dinner
buy food– <i>Used For</i> → satisfy hunger
fast food restaurant– <i>Used For</i> → buy food
buy food– <i>Used For</i> → keep food in refrigerator
buy food– <i>Used For</i> → prepare celebration
go to store– <i>Used For</i> → buy food
buy food– <i>Used For</i> → provide nourishment for person
go to mall– <i>Used For</i> → buy food
buy food– <i>Used For</i> → eat lunch
eat place– <i>Used For</i> → buy food
health food store– <i>Used For</i> → buy foodS
store– <i>Used For</i> → buy food
buy food– <i>Used For</i> → make dinner
buy food– <i>Used For</i> → provide nourishment
earn live– <i>Used For</i> → buy food
buy food– <i>Used For</i> → person with money
buy food– <i>Used For</i> → give to someone else

tence elimination is an important task in text summarization to reduce the size of summary. Before finding the semantic relatedness score between query and input text sentence, extract the important sentences from input text document. In fact, extracting important sentences helps in finding rich information for summary. Important sentences are those sentences which carry more significant information.

Table 3 Different Types of Relations present in WordNet

Relation name	Meaning	Example
Synonymy	Same or identical meaning	Bad and regretful
Hyponymy	More specific meaning	Animal and rhino
Hypernym	Specifies superordinate of hyponymy	Rhino and animal
Meronymy	Part to whole relation	Eye and face
Holonymy	Whole to part relation	Face and eye
Troponymy	Manner relation	Move and walk
Entailment	One follows other	Eat and chew
Antonymy	Opposite meaning	Man and woman

While finding semantic relatedness score between query and input text words, add additional words or phrases with query words or phrases. This is known as query expansion. In query expansion, input is the query words or phrases and output will be the related words or phrases. Expanding query words help in extracting more useful and essential sentences according to user's need. Before finding semantic relatedness score, find the correct sense of the word which will be appropriate for specific context of the sentence. Thus, word sense disambiguation comes here for finding appropriate sense with respect to the context of the sentence.

The overall process for finding query-based text summarization using word sense disambiguation is shown in Fig. 1:

The following steps are needed for finding query-based text summarization using word sense disambiguation (QBTSWSD):

Important sentence selection

Find important sentences on the basis of defined criteria. Find out the content words and do the stemming of content words.

Expansion of query

Find out the word phrases in the query using the method provided in Bird et al. [4].

Find out the expanded terms for each word in query.

Find out the expanded terms for word phrases in query.

Find sense of a word

Find out the exact sense of all query words and expanded words and input text words using Algorithm 1.

Find sentence relatedness score

Find out the semantic relatedness score between the query words and expanded query words with words of the important sentences using the Algorithm 2.

Find redundancy-free sentences

Find and remove the redundant sentences between the input text sentences using the Algorithm 3.

Generate Summary

Extract required number of sentences on the basis of total relatedness score until the desired summary length is reached.

Important sentence selection

To enhance the quality of summary content, the method first finds important sentences from the input text documents. The

Fig. 1 Block diagram of query-based multi-document summarization

method for finding important sentences is proposed in Rahman et al. [25]. The following nine criteria are used to find important sentences from input text document:

1. **Title Matching Words:** If the words present in a sentence are also present in title or heading of the text document, then that sentence is considered as an important one.
2. **Proper Noun:** A sentence containing proper noun or entity name signifies as an important sentence.
3. **Numerical Data:** Numerical data give more information in a text document. A sentence having numerical data is always preferred to be included as an important sentence.
4. **Thematic Word:** Thematic word means appearing more frequently in a text document. Presence of a thematic word in a sentence signifies the sentence as an informative one. In our method, we consider top ten most frequent words from the text file and consider those sentences where these thematic words are present.
5. **Noun Phrase:** Noun phrases always give important information in a sentence. Here, chunk parser [4] is used to find noun phrases.
6. **Font-based Word:** Uppercase, bold, italics or underlined fonts give more meaningful sentence likely to be added as an important one.
7. **Cue Phrase:** Cue phrase (*e.g. this letter, this report, summary, in conclusion, argue, purpose, development*) are considered for incorporation in important sentences.
8. **Sentence Length:** It is found that the presence of more content words in a sentence makes the sentence an informative one. Therefore, we consider long sentences.
9. **Sentence Position:** Sentences present in a first and last of a paragraph are always considered as informative sentences.

Expansion of query words

A query is considered as a set of words. Query expansion adds extra words to the query word set. Query word expansion is done using spreading activation. Spreading activation method is used for finding related query words from seman-

tic networks. Initially, a set of source nodes or concepts are labeled with weight or activation value and repeatedly spreading the activation in to the linked concepts. These weights get decreased as it spread through the semantic network. Activation may start from alternative paths and end when two alternative paths get the same concept.

The node of activation origin is first activated with a initial score as 1.0. Next node will be one node away from the activation origin and so on [8]. Equation 1 is used to find the activation score of a target node b which is connected to the source node a :

$$AS(a, b) = d \sum_{a \in neighbor(b)} \times Score_{Act}(a) \times w(a, b) \quad (1)$$

Here, $AS(a, b)$ stands for Activation Score and d is the distance discount. d helps in getting higher activation score of the node if the node is closer to the activation origin. $Score_{Act}(a)$ represents activation score of source node a . $w(a, b)$ is the weight of path from source node to the target node. Different relation will have different weights. For example, the weight of 'Is A' is 0.9 and the weight of 'Defined As' is 1.0.

Find correct sense of a word

Nouns, verbs, adjectives and adverbs are clustered in to a group of synonyms called as synsets. Each synset is represented as a distinct concept. Some concepts are more general and some are more specific. Each synonym has senses with particular definition. Table 4 gives an example of set of synonyms for a concept 'bank' and the definition of each sense:

In general, semantic relatedness between two words or concepts is calculated using every sense of the first word with all the senses of the second word. Finally among all the scores, highest value is taken. But, it is more appropriate to use the correct sense of a word which will be the most suitable with respect to the context of a sentence. An ontology-based method is presented to find correct sense. Babelfy is used to find out the meaning of a word or a word phrase. Babelfy is based on bablenet ontology [16]. Bablenet is a multilingual

Table 4 Synset and definition of word 'bank'

'bank.n.01'	Sloping land (especially the slope beside a body of water)
'bank.n.02'	A financial institution that accepts deposits and channels the money into lending activities
'bank.n.03'	A long ridge or pile
'bank.n.04'	An arrangement of similar objects in a row or in tiers
'bank.n.05'	A supply or stock held in reserve for future use (especially in emergencies)
'bank.n.06'	The funds held by a gambling house or the dealer in some gambling games

semantic network. It extracts all linked fragments from the text sentence and gives all possible meaning present in the semantic network. Bablenet merges wikipedia with WordNet with an efficient way. It is easy to detect the correct sense of a word. The method also finds the definition of each sense of a content word using WordNet ontology [14].

It is always seen that a word gives more specific meaning with respect to the sentence if we use the definition of other content words present in the same sentence. Words present in a sentence are always semantically related. For example, two sentences are taken: ‘Men sitting by the banks of river Brahmaputra and indulging in conversations all afternoon’, and ‘She is in the bank to deposit money’. In the first sentence, the word ‘river’ specifies the ‘bank’. Hence, the ‘bank’ is related with sloping land besides a river and for the second sentence, ‘deposit’ and ‘money’ both specify that this ‘bank’ is associated with financial institution.

To find the correct sense of a particular word, initially we take the definition of all words except the target word using babelfy. We also get all the definitions of all the senses for the target word using WordNet. Now score of a word sense is calculated based on frequency of occurrence of common words among each target sense definition and the definition of all other content words. Finally, we take the highest score sense as an appropriate one. We use Eq. 2 to find the score of a sense:

$$Score W_{sense} = \log \left(1 + \frac{n(c_t \in S, w_d \in w)}{n(t)} \right), \quad (2)$$

where (c_t) = Definition of all the content words present in the sentence S , (w_d) = Definition of the particular sense of the word w , $n(c_t, w_d)$ = Number of common words between the definition of all content words present in the sentence S and the definition of the particular sense of the word (w_d) , $n(t)$ = Total number of unique words in the definition of all content words present in the sentence S and the definition of the particular sense of the word (w_d)

The steps to find out the correct sense of a word present in a sentence are described in Algorithm 1.

Data: Sentence (S)

Result: Correct senses of words (W_s) in S

Perform the stop word removal and find content words (W_c)

Perform the stemming (S_c) of the W_c

for each word W in S_c do

Find out the synset of W with senses using WordNet

for each each sense (W_s) of W do

Find out the definition (W_d) of W_s using WordNet

Find out the meaning of W_c using Babelfy

Find $Score W_{sense}$ using equation 2

end

Take the sense for which $Score W_{sense}$ is highest

end

Algorithm 1: To find correct sense of a word

This above method can also be used in expanded query terms sense detection. After getting the expanded term, we place the new expanded query term in place of original query term and apply the same method as mentioned in Algorithm 1.

Finding sentence relatedness score

In query-based text summarization, query and input text documents are taken as inputs. Information of a sentence is carried by a content word. Before finding score of semantic relatedness, stop word removal and stemming is done on the input text sentences and query sentence, so that only content words are present for each sentence. The proposed score is based on HSO measure [22] and collocation [13]. HSO [7] is a path-based semantic relatedness measure and collocation means frequency of commonly co-occurring of multiple words. HSO measure classifies the relations as having direction: for example *is-a* relation is considered as upward direction and *has-part* relation can be treated as horizontal direction. From the literature survey it is found that HSO measure uses more relations than the other existing relatedness measures [21]. The relatedness score depends on the path distance and the direction of path. Higher score signifies the minimum distance and less changes of direction between the two words. As we get the correct sense of every word using Algorithm 1, we add the sense at the time of finding semantic relatedness score. This will give the accurate semantic relatedness score as per the context of the sentence. Initially, the method for finding semantic relatedness score between two words is defined in following Eq. 3:

$$hso(w_1, w_2) = 2 \times A - P_L(w_1, w_2) - C \times C_D(w_1, w_2) \quad (3)$$

Here, P_L represents Path Length and C_D represents Change in Direction.

A and C are the constants taking values as 8 and 1, respectively. The maximum relatedness score 16 signifies that two words are identical and minimum score stands for no relation between them and the value will be 0. We assume that two words to be semantically related if the score is equal or higher than the average value. Sometimes, we need to find semantic relatedness score between two word phrases. The method to find semantic relatedness score between two word phrases is given in Eq. 4:

$$hso(W_1, W_2) = \frac{\sum_{w1 \in W_1, w2 \in W_2} (score_{hso}(w1, w2))}{MRS} \quad (4)$$

Here, MRS represents Maximum Relatedness Score.

Now, we consider collocation which gives commonly occurring frequent words. The co-occurrence between two terms is calculated by finding the collocation value of bi-

gram frequency. This bi-gram is counted using Wikipedia Corpus [6]. The method uses Eq. 5 to find bi-gram collocation between two words w_1 and w_2 :

$$bigram(w_1, w_2) = \log_2 \frac{P(w_1, w_2)}{P(w_1) \cdot P(w_2)} \quad (5)$$

$P(w_1, w_2)$ = Probability of occurring two words, w_1 and w_2 together in Wikipedia Corpus, $P(w_1)$ = Probability of occurring word w_1 in Wikipedia Corpus, $P(w_2)$ = Probability of occurring word w_2 in Wikipedia Corpus.

To find collocation between two word phrases, we have to see the occurrence of each word present in the first word phrase with all words of second word phrases and consider the maximum value using Eq. 6:

$$bigram(W_1, W_2) = \max_{w_1 \in W_1, w_2 \in W_2} \sum (bigram(w_1, w_2)) \quad (6)$$

The method to find total relatedness score between two word phrases is given in a following linear Eq. 7:

$$Score(W_1, W_2) = hso(W_1, W_2) + bigram(W_1, W_2) \quad (7)$$

Finally, the method uses following sentence relatedness score 8 to calculate relatedness between two sentences:

$$Score(S_1, S_2) = \max_{W_1 \in S_1, W_2 \in S_2} (Score(W_1, W_2)) \quad (8)$$

To implement this sentence relatedness score in query-based text summarization, we have to calculate relatedness score between query sentence and important text sentences. To apply for expanded query terms we have to place the expanded query terms in place of each of the original query terms. The method for assigning sentence relatedness score to each sentence is described in Algorithm 2:

Data: query sentence (Q_s) and important text sentences (S)
Result: Important text sentences with a score
 Do the stop word removal and stemming of Q_s and S
 Find out the senses of all terms (Q_t) in Q_s and word (Q_w) in S using Algorithm 1
for each sentence in S do
 | Find out the sentence similarity score using equation 8
end

Algorithm 2: Sentence Relatedness Score

Generate redundancy-free summary

Redundancy removal is important while preparing the summary. Redundancy-free text summary decreases the summary size and helps in getting different information related

to query. To remove redundancy, the method finds the common words in both the sentences. The method uses following Eq. 9 to find similarity between two sentences using Jaccard similarity coefficient [17]:

$$Similarity(S_1, S_2) = \frac{S_1 \cap S_2}{S_1 \cup S_2} \quad (9)$$

Here, $S_1 \cap S_2$ = Common words between S_1 and S_2 sentences.
 $S_1 \cup S_2$ = Total unique words in S_1 and S_2 sentences.

Higher similarity score signifies similarity between two sentences are more. Extraction of sentences can be done on the basis of lowest similarity score. If the two sentences have similarity value as 90% (threshold) and above, we consider that both sentences carry the same meaning. To get redundancy-free sentences, the method uses Algorithm 3:

Data: Important sentences (S) with sentence relatedness score
Result: Redundancy-free important sentences selection for summary generation
 Select highest sentence relatedness score sentence (S_h) from (S)
for each S_h do
 | Find out the similarity score (S_r) with other sentences using equation 9
 | Consider sentences where $S_r < \text{threshold}$
 | Choose the highest sentence relatedness score sentence (S_h) from the remaining sentences
end

Algorithm 3: Redundancy-free sentence selection for summary generation

Experiments and results

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

Dataset

DUC 2005 and 2006 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 and each summary length is of 250 words only. A brief description of datasets are shown in Table 5. We compare our proposed method with the existing methods participated in DUC 2005 and DUC 2006.

Table 5 Experimental description of Dataset

DUC Dataset	2005	2006
Number of clusters	50	50
Number of average documents in each cluster	32	25
Data source	TREC	AQUAINT
Length of summary	250 words	250 words

Evaluation metric

Popular and standard intrinsic-based metric ROUGE is adapted by National Institute for Standards and Technology (NIST). ROUGE stands for Recall-Oriented Understudy for Gisting Evaluation. ROUGE is widely used by many researchers for text evaluation purpose [20,26,27]. Candidate and reference summary are compared in ROUGE. Different researchers give their methods to generate candidate summary and expert human made reference summary. A set of metrics are available in ROUGE for comparing different summaries [11]. It measures 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-Bi gram Co-Occurrence Statistics and Extension of Skip-Bi gram Co-Occurrence Statistics are the five standard evaluation metrics available in ROUGE. Here, we use official metrics of ROUGE-1 (uni gram-based), ROUGE-2 (bi gram-based) and ROUGE-SU4 on DUC 2005 and 2006 datasets for our experiment purpose.

Impact of word sense disambiguation in query-based text summarization

We first evaluated our method QBTSWSD on DUC 2005 datasets. Evaluation of the method is done by initially without using word sense disambiguation and then using word sense disambiguation in query-based text summarization. From the Fig. 2 it is seen that incorporating word sense disambiguation gives better results for all three ROUGE values.

Comparison with DUC 2005 and DUC 2006 systems

We compare our results with top-performing DUC 2005 and DUC 2006 systems. Tables 6 and 7 present the different ROUGE scores for DUC 2005 and DUC 2006 datasets. This query-based text summarization with word sense disambiguation is compared with the baseline system along with the top ten systems with maximum ROUGE score. These top ten systems are the participants of DUC 2005 and 2006 tasks. The task is to create query-based text summarization. Baseline system generates summary by selecting the first sentence

Fig. 2 Different ROUGE values for query-based text summarization without and with word sense disambiguation incorporation**Table 6** Experimental results on DUC 2005 datasets

System ID	ROUGE-1	ROUGE-2	ROUGE-SU4
QBTSWSD	0.3924	0.08817	0.15423
Columbia	0.37481	0.06857	0.12772
CCS-NSA-05	0.35983	0.06278	0.11895
FTextST-0	0.35851	0.0674	0.12324
FDUSUM	0.36088	0.06088	0.11878
ISI-Webcl	0.36584	0.06425	0.12251
SFU v2.4	0.36012	0.06323	0.12180
OHSU-DUC05	0.35012	0.06325	0.11896
NUS3	0.37515	0.07251	0.13163
PolyU	0.36977	0.07174	0.12973
isi-bqfs	0.36369	0.06984	0.12526
BASELINE	0.27522	0.04025	0.08716

from the text documents until it gets the required summary length.

Tables 6 and 7 provides the scores for different ROUGE values: ROUGE-1, ROUGE-2 and ROUGE-SU4. From the results, it is found that QBTSWSD method outperforms all the top ten systems along with baseline systems for all ROUGE scores.

Comparison with related systems

To further prove the performance of our method, we compare our QBTSWSD Method with other recognized and current state -of-the-art existing methods. The following methods

Table 7 Experimental results on DUC 2006 datasets

System ID	ROUGE-1	ROUGE-2	ROUGE-SU4
QBTWSWD	0.46745	0.09982	0.17563
Columbia06	0.39862	0.08264	0.14012
OGI.OHSU06	0.38594	0.08408	0.13912
OnModer	0.40488	0.08987	0.14755
ICL_SUM	0.40440	0.08792	0.14486
UMich	0.40206	0.08444	0.14483
TLR	0.39908	0.08576	0.14381
CCS06	0.38991	0.08679	0.14170
JKD	0.38807	0.08707	0.14134
IITH_SUM	0.40980	0.09505	0.15464
LIA_THALES	0.39922	0.08700	0.14522
BASELINE	0.30217	0.04947	0.09788

Table 8 Experimental comparison between QBTWSWD and other methods on DUC 2006 datasets

System ID	ROUGE-1	ROUGE-2	ROUGE-SU4
QBTWSWD	0.46745	0.09982	0.17563
TL-Transum	—	0.10779	0.15854
QSLK	0.4287	0.0968	0.1673
CTMSUM	0.4157	0.0968	0.1548
WAASUM	0.4238	0.0917	0.1671
Topical-N	0.4010	0.0893	0.1459
Qs-MR	0.4012	0.0914	0.1444
SVR	0.4018	0.0926	0.1485
LEX	0.4030	0.0913	0.1449

are selected to compare with our method using DUC 2006 datasets: (1) TL-Transum by Lierde et al. [28], (2) QSLK by Abadi et al. [1], (3) CTMSUM by Yang et al. [30], (4) WAASum by Canhasi et al. [5], (5) Topical-N by Yang et al. [31], (6) Qs-MR by Wei et al. [29], (7) SVR by Ouyang et al. [18], (8) LEX by Huang et al. [9]. We select these methods as they have performed best on DUC 2006 datasets. From the Table 8, it is seen that QBTWSWD Method shows considerably better results for all the three values of ROUGE matrices.

Conclusion and future work

In this paper, a method for query-based text summarization is presented by incorporating word sense disambiguation. The proposed method uses semantic network to expand the query. Here, different relations from semantic networks are used to find relatedness between an input sentence and the query. Our method contributes in finding correct sense of a word present in a sentence by finding the contextual meaning of

all the words. This word sense will contribute in calculating accurate semantic relatedness score between two words with respect to the context of the sentence. Our evaluation and experiments have shown that using word sense, disambiguation in query-based text summarization gives better result than different participating systems in DUC 2005 and DUC 2006 and also for current state-of-the-art systems. In future, we can try to improve the QBTWSWD method for multiple topics. Coherency can also be checked to get fluent and well-organized summary.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

1. Abdi A, Idris N, Alguliyev RM, Aliguliyev RM (2017) Query-based multi-documents summarization using linguistic knowledge and content word expansion. *Soft Comput* 21(7):1785–1801. <https://doi.org/10.1007/s00500-015-1881-4>
2. 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
3. Bhogal J, MacFarlane A, Smith P (2007) A review of ontology based query expansion. *Inf Process Manag* 43(4):866–886
4. Bird S, Klein E, Loper E (2009) Natural language processing with python: analyzing text with the natural language toolkit. O'Reilly Media, Inc, Newton
5. Canhasi E, Kononenko I (2014) Weighted archetypal analysis of the multi-element graph for query-focused multi-document summarization. *Expert Syst Appl* 41(2):535–543
6. 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
7. Hirst G, St-Onge D et al (1998) Lexical chains as representations of context for the detection and correction of malapropisms. *WordNet Electron Lex Database* 305:305–332
8. Hsu MH, Tsai MF, Chen HH (2008) Combining wordnet and conceptnet for automatic query expansion: a learning approach. In: *Information Retrieval Technology*, pp 213–224
9. Huang L, He Y, Wei F, Li W (2010) Modeling document summarization as multi-objective optimization. In: 2010 third international symposium on intelligent information technology and security informatics (IITSI), IEEE, pp 382–386
10. 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. <https://doi.org/10.1145/318723.318728>
11. 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
12. Liu H, Singh P (2004) Conceptnet-a practical commonsense reasoning tool-kit. *BT Technol J* 22(4):211–226. <https://doi.org/10.1023/B:BTJJ.0000047600.45421.6d>

13. McKeown KR, Radev DR (2000) Collocations. In: Handbook of natural language processing Marcel Dekker
14. Miller GA (1995) Wordnet: a lexical database for english. Commun ACM 38(11):39–41
15. Mohamed AA, Rajasekaran S (2006) Improving query-based summarization using document graphs. In: 2006 IEEE international symposium on signal processing and information technology, IEEE, pp 408–410. <https://doi.org/10.1109/ISSPIT.2006.270835>
16. Navigli R, Ponzetto SP (2012) Babelnet: the automatic construction, evaluation and application of a wide-coverage multilingual semantic network. Artif Intell 193:217–250. <https://doi.org/10.1016/j.artint.2012.07.001>
17. Niwattanakul S, Singthongchai J, Naenudorn E, Wanapu S (2013) Using of jaccard coefficient for keywords similarity. Proc Int Multiconf Eng Comput Sci 1:380–384
18. Ouyang Y, Li W, Li S, Lu Q (2010) Intertopic information mining for query-based summarization. J Assoc Inf Sci Technol 61(5):1062–1072
19. Ouyang Y, Li W, Li S, Lu Q (2011) Applying regression models to query-focused multi-document summarization. Inf Process Manag 47(2):227–237. <https://doi.org/10.1016/j.ipm.2010.03.005>
20. Pandit SR, Potey M (2013) A query specific graph based approach to multi-document text summarization: simultaneous cluster and sentence ranking. In: 2013 international conference on machine intelligence and research advancement (ICMIRA). IEEE, pp 213–217
21. Patwardhan S, Banerjee S, Pedersen T (2003) Using measures of semantic relatedness for word sense disambiguation, vol 2588. CICLing, Springer, pp 241–257
22. 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
23. Plaza L, Diaz A (2011) Using semantic graphs and word sense disambiguation techniques to improve text summarization. Procesamiento del lenguaje natural 47:97–105
24. Rahman N, Borah B (2015) A survey on existing extractive techniques for query-based text summarization. In: 2015 international symposium on advanced computing and communication (ISACC). IEEE, pp 98–102
25. Rahman N, Borah B (2017) A method for semantic relatedness based query focused text summarization. In: International conference on pattern recognition and machine intelligence. Springer, pp 387–393
26. Steinberger J, Ježek K (2012) Evaluation measures for text summarization. Comput Inf 28(2):251–275
27. 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
28. Van Lierde H, Chow TW (2019) Query-oriented text summarization based on hypergraph transversals. Inf Process Manag 56(4):1317–1338
29. 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
30. Yang G (2014) A novel contextual topic model for query-focused multi-document summarization. In: 2014 IEEE 26th international conference on tools with artificial intelligence (ICTAI). IEEE, pp 576–583
31. Yang G, Wen D, Sutinen E et al (2013) A contextual query expansion based multi-document summarizer for smart learning. In: 2013 international conference on signal-image technology & internet-based systems (SITIS). IEEE, pp 1010–1016
32. Ye X, Wei H (2008) Query-based summarization for search lists. In: First international workshop on knowledge discovery and data mining, 2008. WKDD 2008. IEEE, pp 330–333. <https://doi.org/10.1109/WKDD.2008.14>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.