A Topic-driven Summarization using K-mean Clustering and Tf-Isf Sentence Ranking

Rajesh Wadhvani
Computer Science Department
National Institute of Technology
Bhopal, India

R. K. Pateriya
Computer Science Department
National Institute of Technology
Bhopal, India

Devshri Roy
Computer Science Department
National Institute of Technology
Bhopal, India

ABSTRACT

Enormous online information is available due to the World Wide Web. This needed efficient and accurate summarization systems to extract significant information. Text summarization system automatically generates a summary of a given document and helps people to make effective decisions in less time. In this paper two methods have been proposed for query-focused multi-document summarization that uses k-mean clustering, term-frequency and inverse-sentence-frequency method for sentence weighting to rank the sentences of the documents with respect to a given query. The proposed method finds the proximity of documents and query, and later uses this proximity to rank sentences of each document. It is assumed that the document which is nearer to a query might contain more meaning full sentences with respect to the information need expressed by user's query further if a sentence contains rare query term than it is more informative than the sentences that contains frequent query term. Both methods first gives weights to documents according to their proximity and use these document weights to rank each of their sentences with tf-idf ranking function. A relative study for proposed methods has been done and experimental results shows that both methods are comparable because of a slight difference in performance. DUC 2007 test dataset and ROUGH-1.5.5 summarization evaluation package is used for evaluation purpose.

General Terms:

Text summarization, Query-based summaries

Keywords:

Sentence Extraction, Document Clustering, F-score

1. INTRODUCTION

Summarizing the documents manually requires lots of efforts and time and thus it is very difficult. A technique is required where a computer program make a reduced version of a text while preserving the content of the source, i.e, summarize document automatically. Automatic text summarization is not new. Work in this area originated in the 1950s, when the first breakthroughs were achieved by Luhn (1958). Despite this, most of the significant research in this area has been carried out in the last few years. The objective of automatic summarization is to take an information source, extract content from it and supply the most important content to the user in a condensed form, in a manner sensible to user's or application's need, see (Mani, 2001) for details. According to the function, a summary can be classified as generic or user-oriented. Generic summary present the authors viewpoint on the document. It considers all the information in the document to create summary. Moreover user-oriented summary consider only that information which is relevant to user query. Based on relationship that a summary has to the source document, a summary can be either abstract or extract[1,2]. An extract involve selection of informaive sentences or paragraphs from source text in the summary. Moreover abstract involve the identification of salient concepts in the source document and rewrite them through natural language generation.

The notion of information retrieval is to locate documents that might contain the relevant information. Generally, when a user fires a query his desire is to locate relevant information rather than locate a ranked list of documents. The retrieved documents might contain the desire information and it leaves user with a massive amount of text. There is a requirement for a tool to shrink the amount of text in order to comprehend the complete text. The query focused summarization track at DUC aims at doing this. People often have questions in their mind and they expect answers, as opposed to a set of documents as output. This motivated to focus on query based summarization. This paper proposed two methods for query-focused summarization based on tf-isf sentence weighting and also including the document weight to rank sentences. The thinking behind the document weighting is "more closer the document to query the probability of its sentences close to query is high". A relative for proposed methods has been done. The aim of document summarization is to find out the salient units of the document and rank these units according to their significance before giving the final summary to user. Automated system comprises different modules which are independent of each-other. Researcher and experts have categorized document summarization into three different stages [3]. The stages are

—Topic identification

—Topic fusion

—Summary generation

In **topic identification** stage system identifies the significant units of document; A unit may be a word, a sentence or a paragraph. To weight these significant units of document,independent models can be used. The scores for each unit are then combined in order to provide a single score. The n top-scoring units, depending on

the summary-length is used to form a summary. The topic identification stage generates the simplest summary. On the lowest-level significant words are identified, weighted according to significance. On sentence-level, sentences are weighted according to occurrence of significant words. The words in sentence can be weighted by different means such as word frequency, positional importance, cue words existence, words overlapping with heading, text connectivity etc. In **Topic fusion** extracted coherent units are merged and insignificant content of text are removed. **Summary generation phase** produces the final summary based on earlier two phases in human readable form. The aim of all these phases are to conserve the salient features of original document before producing and presenting the relevant information to user.
This paper is organized as follows. Literature review is presented in Section 2, this chapter gives an account of previous works done in the area of text summarization. Description of the proposed work is given in Section 3, this chapter provides the detail of system architecture. It also explains the proposed work in detail. Section 4 provides the description of tools used for evaluation and also gives the detail of evaluation metrics used for evaluation purpose. Section 5 is Result and Analysis which provides the results obtained by the proposed methods. The experimental result shows the accuracy of the methods using precision, recall and F-score measures. Section 6 is the conclusion and section 7 is limitation.

2. RELATED WORK

Text comprises of paragraphs, sentences and the smallest unit is word. In literature, researchers have broken the text into number of paragraphs and sentences to achieve better performance of summarization system. The performance of these systems have been studied in number of papers using various techniques such as statistical, graph-based, machine learning, cluster based etc. In the literature of automated text summarization varieties of approaches, either extractive [1,2] or abstractive, have been proposed. Understanding of contents, reformulation and sentence-compression is done while abstraction [4,5] where as the sentences of text are ranked and important ones are picked-up in extraction. In extractive summarization, sentence/paragraph ranking [6,7,8] is the centre of attention; various methods are used to rank sentences/paragraphs. In [6], compare the effectiveness of paragraph, word, and coherence based sentence ranking approaches. The best performance was accomplished by coherence based approaches. In query-based summarization most sentence ranking methods are based on a usual matching between query and sentences. The job of the query words and the named entities seemed in the query were exclusively emphasized in [9,10]. In [11], a topic-sensitive version of PageRank was proposed to incorporate the relevance of a sentence to the query into LexRank to get a biased PageRank ranking. The sentence-ranking score obtained by this process pointed the query biased informativeness of the sentence and sentences with high ranks are selected to form the summary. In [12], regression models are employed to focus query in multi-document summarization.

3. PROPOSED MODEL

Query-based multi-document summarization generates the summary by extracting a proper set of informative sentences from multiple text units based on the information need presented in query. This chapter proposes two methods for query-based multi-document summarization that uses k-mean clustering for document weighting and term frequency and inverse sentence frequency for weighting the query-term to rank the sentences of the documents with respect to a given query. These two methods are:

—**ISF-D Method**: **I**nverse-**S**entence-**F**requency at **D**ocument level.

—**ISF-C Method**: **I**nverse-**S**entence-**F**requency at **C**orpus level.

The ISF-D method provide tf-isf weight to query-terms at document level. In this query terms get different weights for each document associated with query where as ISF-C method provide the tf-isf weight at corpus level. These methods finds the proximity of documents and query, and provide a normalize weight to each document associated with a single query and later uses this weight to rank sentences of each document. Proposed methods takes advantage of the fact that the more closer the document to query, the likelihood of its sentences to be close to query is higher, both methods first give weights to query-terms, then documents according to their proximity with the query and later use both weights to rank each sentence from the multi-document set.
Both methods have five phases, which are as following:

(1) Preprocessing phase
(2) Query-terms weighting phase
(3) Document weighting phase
(4) Sentence weighting phase
(5) Summary generation phase

Phases first, third and five are common where as there is a slight difference in second and fourth phase in case of two methods.

3.1 Preprocessing phase

Preprocessing phase is divided into two phases; a cleaning phase and document presentation phase. The preprocessing phase is common for both the proposed methods.

(1) **Cleaning phase:** This phase is for both query and documents. In this phase we remove stop words and special symbols like punctuation marks. To remove stop words we use two methods; first method removes stop-words by setting the minimum word length to three and the second method uses a precompiled list of stop words to remove them.

(2) **Document presentation phase:** After the raw files are prepared, these are passed through a structuring phase, where in all the files are structured into one unique term verses paragraph matrix. Each document Dm consist of different sentences named as $s_1, s_2, ..., s_n$ where each paragraph is collection of m unique terms $t_1, t_2, t_3, ...t_m$.

3.2 Query-terms weighting

The two proposed methods differ in this phase. Before calculating the weight for each method, term frequency is calculated, It measures the importance of a term within a document. A document or zone that mentions a query term more often has more to do with that query. Therefore weight of index term should consider term frequency. Number of occurrences of term within document is known as raw term frequency, but it is not what we want because relevance does not increase proportionally with raw term frequency.

Inverse-sentence frequency: If a query word occurs in every single sentence it means that query-term is a frequent term and can't distinguish sentences therefore a low weight is assigned to it. Moreover if a query is rare a higher weight is assigned to it.

Some set notations are used are following:

$Q = \{q_1, q_2, q_3 ... q_m\}$ is set of unique query-terms.

$D = \{d_1, d_2, d_3 ... d_n\}$ is set of documents associated with query.

$S = \{s_{1,1}, s_{2,1}, ..., s_{j,1}\}, \{s_{1,2}, s_{2,2}, ..., s_{k,2}\}$,.......... ,$\{s_{1,n}, s_{2,n}, ..., s_{l,n}\}$ is the set of sets, and each set is the sentence collection per document that is, $s_{l,n}$ is the l^{th} sentence of document n.

Based on the term frequency and inverse sentence frequency query terms are weighted using ISF-C and ISF-D methods as follows.

(1) **I**nverse-**S**entence-**F**requency at **C**orpus level (ISF-C)

The ISF-C method provide tf-isf weight to query terms at corpus level.

(a) For each sentence $s_{l,n}$,do,

(b) For each query-term, q_m , do,

i. Calculate log(tf) for q_m ,get term-frequency (tf) of q_m from, $s_{l,n}$ sentence-count vector.

$$\log(tf_{q_m}) = 1 + \log_{10}(count\ of\ q_m\ from\ s_{l,n}\ ) \quad (1)$$

ii. Calculate inverse-sentence-frequency for q_m using :

$$isf - C_{q_m} = log_{10}(N/sf_{q_m}) \quad (2)$$

where N is total number of sentences in corpus and sf_{q_m} is total number of sentences in which q_m occur within corpus.

iii. Calculate final query-term weight, $w_{(q_m)}$

$$ISFCW_{q_m} = \log(tf_{q_m}) * isf - C_{q_m} \quad (3)$$

where, $ISFCW_{q_m}$ is weight of q_m using ISF-C method.

(2) **I**nverse-**S**entence-**F**requency at **D**ocument level (ISF-D)

The ISF-D method provide tf-isf weight to query terms at document level. In this query terms get different weights for each document associated with query.

(a) For each sentence $s_{l,n}$, do,

(b) For each query-term, q_m, do,

i. Calculate log(tf) for q_m, get term-frequency (tf) of q_m from, $s_{l,n}$ sentence-count vector.

$$\log(tf_{q_m}) = 1 + \log_{10}(count\ of\ q_m\ from\ s_{l,n}\ ) \quad (4)$$

ii. Calculate inverse-sentence-frequency for q_m using :

$$isf - D_{q_m} = log_{10}(N/sf_{q_m}) \quad (5)$$

where N is total number of sentences in document d_n and sf_{q_m} is total number of sentences in which q_m occur within d_n.

iii. Calculate final query-term weight, $w_{(q_m)}$

$$ISFDW_{q_m} = \log(tf_{q_m}) * isf - D_{q_m} \quad (6)$$

where, $ISFDW_{q_m}$ is weight of q_m using ISF-D method.

3.3 Document weighting phase

Author writes a document to represent an idea, an event, and a concept moreover a theme. Directly or indirectly all the sentences of documents circumnavigates around author's viewpoint. Therefore it is assumed that document is a cluster of similar sentences. Henceforth the document shall be interchangeable used as cluster especially in document weighting phase. For document weighting k-mean clustering is used; k-means clustering is a technique of cluster analysis with the aim of partition n objects into k clusters in which each object belongs to the cluster with the closest centroid (mean). For document-weighting, previously calculated, sentence-weight vector is converted into normalized sentence weight vector that is a unit-vector. Similarly query-count-vector is also converted into unit-vector. so form now $s_{l,n}$ is sentence unit vector of l^{th} sentence of document d_n.

Further we use vector space model and map each document and a query into that vector-space. The number of dimensions for vector space model is the cardinality of set Q ($|Q|$) that is number of unique query terms and query terms are dimensions. Each document d_i from D is mapped, by mapping the sentence-unit-vector of each sentence, in $|Q|$-dimension vector space and in the same way query is mapped by mapping query unit vector. A specific document weight is calculated by following steps:

(1) Find centroid for each document d_i, which is a mean of each sentence unit vector.

$$C(d_i) = \sum \frac{s_{l,i}}{n} \quad (7)$$

Where, $C(d_i)$ is the centroid of document d_i and $S_{l,i}$ is the sentence-unit-vector of l^{th} sentence of document d_i and, n is the total number of sentences representing d_i in $|Q|$-dimension vector space.

(2) Find out the Euclidean distance (ed) between centroid of each document d_i and the query to measure the proximity between document d_i and the query by:

$$ed_i(C(d_i), q) = \sqrt{(C(d_1) - q_1)^2 + ... + (C(d_i) - q_i)^2} \quad (8)$$

The one more option here, we have that calculate proximity between d_i and a query by using cosine-similarity between the centroid of d_i and the query. The result of this step is the Euclidean distance (ed_i) between centroid of each document d_i and the query.

(3) Calculate a normalized weight for each document d_i by,

$$w(d_i) = \frac{ed_i}{\sum ed_i} \quad (9)$$

Where,w(d_i) is the normalized weight for document d_i, ed_i is the euclidean distance between d_i and the query. The result of this phase is the document-weight w(d_i) for each document d_i.

3.4 Sentence weighting phase

Weight calculated from the second phase and third phase, for each query-term using ISF-D , ISF-C method and document-weight (d_i). To calculate sentence weight for each query-corpus sentence following steps are used:

(1) Sum all the query-terms weight found in sentence $S_{l,n}$.
(2) Multiply corresponding document weight w(d_i).
 (a) Calculate sentence-weight using query-terms weight calculated by **ISF-D** and document-weight (d_i):

$$ISFDW(s_{l,n}) = (\sum_{q_{m\epsilon s_{l,n}}} ISFDW_{(q_m)})*w(d_n) \quad (10)$$

Where, $ISFDW_{s_{l,n}}$ is the weight of l^{th} sentence of document n using ISF-D method. $S_{l,n}$ is the l^{th} sentence of document (d_i).
 (b) Calculate sentence-weight using query-terms weight calculated by **ISF-C** and document-weight (d_i):

$$ISFCW(s_{l,n}) = (\sum_{q_{m\epsilon s_{l,n}}} ISFCW_{(q_m)})*w(d_n) \quad (11)$$

Where,$ISFCW_{s_{l,n}}$ is the weight of l^{th} sentence of document n using ISF-C method. $S_{l,n}$ is the l^{th} sentence of document (d_i).

Every sentence of the corpus now have some for both ISF-D and ISF-C method. List them into decreasing order.

3.5 Summary generation phase

From the decreasing rank list of sentences, choose a top-rank sentence and add to the summary for both ISF-D and ISF-C, repeat the step till summary-length reach up to 250 words. Before adding a new sentence in summary we also take care of redundancy by not including already added sentence. We have two summaries one for ISF-D method and other is for ISF-C method.

4. EXPERIMENTAL SETUP

(i) Test Dataset: For evaluation DUC-2007 dataset is used[13]. There are 45 topics in the dataset and for each topic a set of 25 relevant documents are given. Each DUC topic comprises of four part; document set number, title of topic, narration and the list of document associated with topic. In this paper, the narration part of topic is used to frame the query. Table 1 shows the description of DUC-2007 dataset.

Table 1. Dataset description

Dataset description	DUC-2007 dataset
Number of topics	45
Number of collections	45
Number of documents per collections	25
Total number of documents in dataset	$45 * 25$
Summary Length	250 words

(ii) Evaluation Metrics: The standard practice in the field of summarization is to have a standard reference summary based on the queries. The summaries are manually generated by human experts. The automated summaries are then compared with the human generated summaries evaluation results are normally obtained by the ROUGE (Recall-Oriented Understudy for Gisting Evaluation). It is a summary evaluation package for judging the performance of the summarization system. The ROUGE summary evaluation package[14] is written in Perl.To evaluate the accuracy and relevance of the automated summary with respect to the expert summaries, three metrics are used :

(1) Recall
(2) Precision
(3) F-score

F-measure [3] is a measure of a system's summary accuracy. It considers both the precision p and the recall r of the system's summary to compute the score. Precision reflects how many of the system's extracted sentences were relevant, and Recall reflects how many relevant sentences the system missed.
Given an input text, a expert's summary, and a automated summary, these scores inform us by quantifying that how closely the system's summary corresponds to the human one. For each unit, we let correct = the number of sentences extracted by the system and the human; wrong = the number of sentences extracted by the system but not by the human; and missed = the number of sentences extracted by the human but not by the system. Then

$$Precision = correct/(correct + wrong)$$

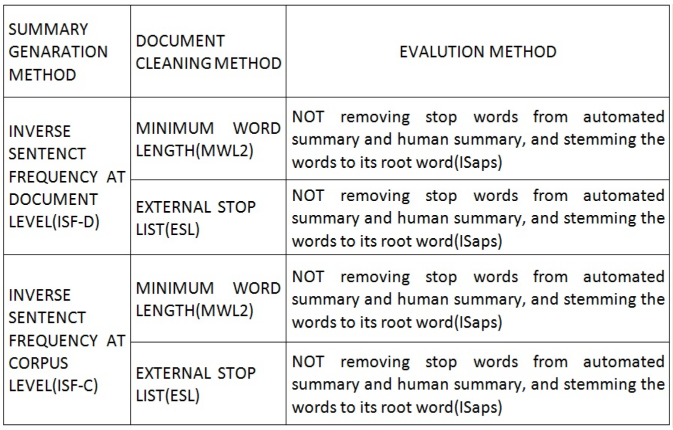
$$Recall = correct/(correct + missed)$$

$$F - Score = \frac{(1+\beta^2)Recall*Precision}{Recall+\beta^2 Preicision}$$

Where, F_β "measures the effectiveness of system's summary with respect to a user who attaches β times as much importance to recall as precision".

5. RESULT AND ANALYSIS

The DUC topic and a set of 25 relevant documents are used for performing the experiment. The two proposed methods are used to create a brief, well-ordered, summary to answer the need for information expressed in the topic statement, actually narration part of topic is used as query for summarization purpose. The sentences are ranked using proposed sentence ranking techniques and top ranked sentences are collected before finally delivering the summary based on query. The summary contains 250 words only.



SUMMARY GENARATION METHOD	DOCUMENT CLEANING METHOD	EVALUTION METHOD
INVERSE SENTENCT FREQUENCY AT DOCUMENT LEVEL(ISF-D)	MINIMUM WORD LENGTH(MWL2)	NOT removing stop words from automated summary and human summary, and stemming the words to its root word(ISaps)
	EXTERNAL STOP LIST(ESL)	NOT removing stop words from automated summary and human summary, and stemming the words to its root word(ISaps)
INVERSE SENTENCT FREQUENCY AT CORPUS LEVEL(ISF-C)	MINIMUM WORD LENGTH(MWL2)	NOT removing stop words from automated summary and human summary, and stemming the words to its root word(ISaps)
	EXTERNAL STOP LIST(ESL)	NOT removing stop words from automated summary and human summary, and stemming the words to its root word(ISaps)

Fig. 1. *Summary evaluation method used by ROUGE.*

Figure 1 shows description of evaluation methods used for evaluation purpose. By default ROUGE include stopwords in (Avg_Recall, Avg_Precision and Avg_F-score) score calculation and stemming of words not perform, to remove stopwords from automated summary and human summary (reference summary), use ROUGE parameter -s and to perform word-stemming use ROUGE parameter -m. ROUGE uses Porter stemmer for word stemming.

Experiment 1: Generate summary for each available topic in the test dataset using ISF-C with MWL2 (Minimum word length 2) cleaning method, Pseudo code as:

```
Data: A DUC topic.
Result: 250 words summary.
(1) Preprocess each document and query by setting Minimum
    Word Length 2 (MWL2).
(2) Weigth each query term using ISF-C method.
(3) Weight each document using k-mean clustering.
(4) Rank sentences of each document using weight calculated
    in Step 2 and Step 3, in decreasing weight order.
(5) Select one by one top sentences till summary-length does
    not exceed 250 words.
```

The automated summaries are compared with the available reference summaries and evaluation results are obtained by the ROUGE (Recall-Oriented Understudy for Gisting Evaluation) summary evaluation package for judging the performance of the summarization system. To compute ROUGE-Scores, ROUGE-1.5.5 will be run with the following parameters : ROUGE-1.5.5.pl -n 4 -2 -1 -U -c 95 -r 1000 -f A -p0.5 -t 0 -s settings.xml Where, settings.xml is a xml file for specifying system summaries and corresponding reference summaries locations.

Table 2.

Evaluation Method	Average_R	Average_P	Average_F
ROUGE-1	0.36986	0.40334	0.38520
ROUGE-2	0.07493	0.07923	0.07690
ROUGE-3	0.02377	0.02507	0.02571
ROUGE-4	0.01179	0.01245	0.01210
ROUGE-L	0.33662	0.35651	0.34576
ROUGE-W-1.2	0.09671	0.19048	0.12809
ROUGE-S*	0.12908	0.15326	0.13921
ROUGE-SU*	0.13098	0.15542	0.14123

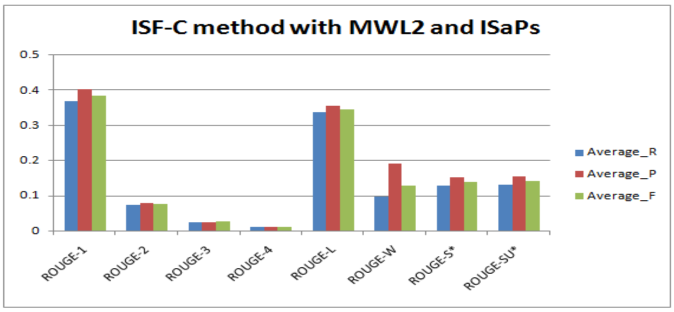


Fig. 2. *Evaluation results of ISF-C method with MWL2 and ISaPs.*

The performance of method with respect to the ROUGE-1 evaluation- in case of Average_R, Average_P, Average_F has gone up 0.36986, 0.40334, 0.38520 respectively.

Experiment 2: Generate summary for each available topic in the test dataset using ISF-C with ESL (External Stopword List) cleaning method, Pseudo code as:

```
Data: A DUC topic.
Result: 250 words summary.
(1) Preprocess each document and query by setting External
    Stopword List (ESL).
(2) Weigth each query term using ISF-C method.
(3) Weight each document using k-mean clustering.
(4) Rank sentences of each document using weight calculated
    in Step 2 and Step 3, in decreasing weight order.
(5) Select one by one top sentences till summary-length does
    not exceed 250 words.
```

Table 3.

Evaluation Method	Average_R	Average_P	Average_F
ROUGE-1	0.37801	0.39925	0.38774
ROUGE-2	0.08005	0.08442	0.08205
ROUGE-3	0.02526	0.02659	0.02587
ROUGE-4	0.01243	0.01309	0.01273
ROUGE-L	0.35717	0.37727	0.36638
ROUGE-W-1.2	0.10244	0.20112	0.13552
ROUGE-S*	0.13588	0.15148	0.14240
ROUGE-SU*	0.13779	0.15356	0.14439

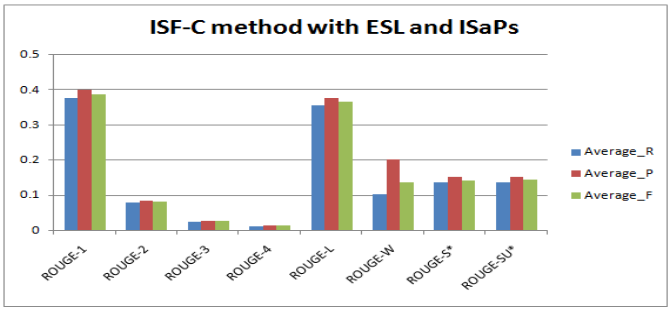


Fig. 3. *Evaluation results of ISF-C method with ESL and ISaPs.*

The performance of method with respect to the ROUGE-1 evaluation- in case of Average_R, Average_P, Average_F has gone up 0.37801, 0.39925, 0.38774 respectively.

Experiment 3: Generate summary for each available topic in the test dataset using ISF-D with MWL2 (Minimum word length 2) cleaning method, Pseudo code as:

```
Data: A DUC topic.
Result: 250 words summary.
(1) Preprocess each document and query by setting Minimum Word Length 2 (MWL2).
(2) Weigth each query term using ISF-D method.
(3) Weight each document using k-mean clustering.
(4) Rank sentences of each document using weight calculated in Step 2 and Step 3, in decreasing weight order.
(5) Select one by one top sentences till summary-length does not exceed 250 words.
```

Table 4.

Evaluation Method	Average_R	Average_P	Average_F
ROUGE-1	0.36850	0.40840	0.38693
ROUGE-2	0.08242	0.09118	0.08646
ROUGE-3	0.02791	0.03084	0.02926
ROUGE-4	0.01484	0.01641	0.01556
ROUGE-L	0.34929	0.38709	0.36675
ROUGE-W-1.2	0.10062	0.20750	0.13535
ROUGE-S*	0.12848	0.15668	0.14044
ROUGE-SU*	0.13038	0.15888	0.14248

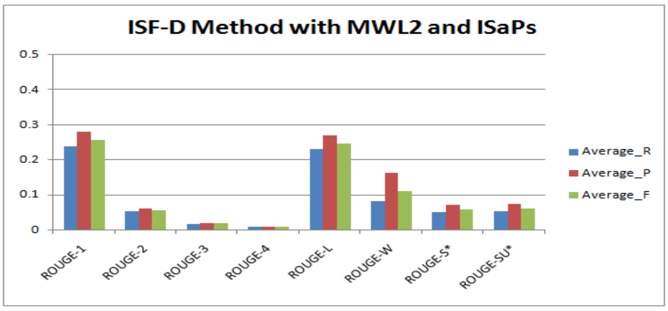


Fig. 4. *Evaluation results of ISF-D Method With MWL2 and ISaPs.*

The performance of method with respect to the ROUGE-1 evaluation- in case of Average_R, Average_P, Average_F has gone up 0.36850, 0.40840, 0.38693 respectively.

Experiment 4: Generate summary for each available topic in the test dataset using ISF-D with ESL (External Stopword List) cleaning method, Pseudo code as:

```
Data: A DUC topic.
Result: 250 words summary.
(1) Preprocess each document and query by setting External Stopword List (ESL).
(2) Weigth each query term using ISF-D method.
(3) Weight each document using k-mean clustering.
(4) Rank sentences of each document using weight calculated in Step 2 and Step 3, in decreasing weight order.
(5) Select one by one top sentences till summary-length does not exceed 250 words.
```

Table 5.

Evaluation Method	Average_R	Average_P	Average_F
ROUGE-1	0.37451	0.40104	0.38663
ROUGE-2.	08085	0.08654	0.08344
ROUGE-3	0.02494	0.02671	0.02575
ROUGE-4	0.01191	0.01282	0.01233
ROUGE-L	0.35369	0.37876	0.36515
ROUGE-W-1.2	0.10138	0.20184	0.13471
ROUGE-S*	0.13426	0.15386	0.14237
ROUGE-SU*	0.13615	0.15596	0.14436

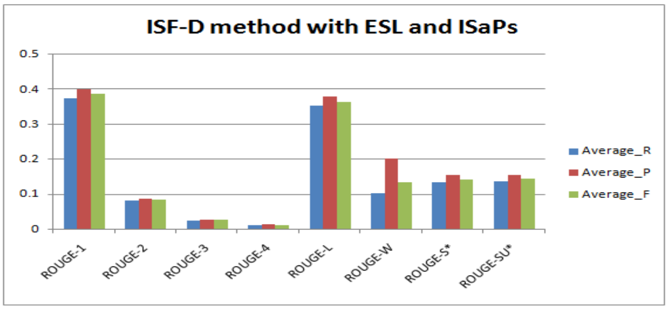


Fig. 5. *Evaluation results of ISF-D Method With ESL and WoSaWoSt.*

The performance of method with respect to the ROUGE-1 evaluation- in case of Average_R, Average_P, Average_F has gone up 0.37451, 0.40104, 0.38663 respectively.

6. CONCLUSION

Rresearch on summarization started about 60 years ago, there is still a long trail to walk in this field. Summarization is a challenging task as it is difficult to automate the process that provides the perfect summary of the document as per the user need it becomes further complicated multiple documents are considered for summarization. In this paper multi-document summarization based on query is studied. Here have tried to highlight the importance of document. The assumption was that, "more nearer a document to a query might contain more meaning full sentences with respect to the need expressed by user's query". The performance of proposed methods that is ISF-C and ISF-D is 0.38774 and 0.38663 respectively. This was an initial attempt and the result shows that there is a need to further improve performance of existing system by incorporating few other techniques. Initial system performance is fairly well.

7. LIMITATION:

If query terms are abbreviated in the document's sentences then the summarizer shall not be able to extract it even though it might be quite relevant in the context of summary.

8. REFERENCES

[1] Vishal Gupta, Gurpreet Singh Lehal, ”A Survey of Text Summarization Extractive Techniques”, Journal of Emerging Technologies in web Intelligence, Vol. 2, No. 3, 2010.

[2] K. Knight and D.Marcu, ”Summarization beyond sentence extraction: a probablistic approach to sentence compression”, Artefcial Intelligence, pages 91-107, 2002 Elsevier Science.

[3] Eduard Hovy, ”Text Summarization”, In R. Mitkov Ed.The Oxford Hand-book of Computational Linguistics, chapter 32 (2005) 583-598.

[4] D. Zajic, B. J. Dorr, J. Lin, and R. Schwartz, ”Multi-candidate reduction: Sentence compression as a tool for document summarization tasks” , Inf. Process. Manage, Volume 43, pp. 1549-1570, November 2007.

[5] H. Daume, D. Marcu, ”A noisy-channel model for document-compression”, In proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Ser. ACL 02. Stroudsburg, PA, USA:Association for Computational Linguistics, pp. 449-456, 2002.

[6] Florian Wolf, Edward Gibson, ”Paragraph-, word-, and coherence-based approaches to sentence ranking: a comparison of algorithm and human performance”, In proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, ACL ’04, Article No. 383, 2004.

[7] Heng-Hui Liu, Yi-Ting Huang , Jung-Hsien Chiang, ”A study on paragraph ranking and recommendation by topic information retrieval from biomedical literature”, In proceeding of the International Conference on Computer Symposium (ICS), 2010, pp. 859-864, Dec. 2010.

[8] Laszlo Grunfeld, Kui-Lam Kwok, ”Sentence Ranking Using Keywords And Meta-Keywords”, Publisher Springer Netherlands, Volume 32, pp 229-258,2006.

[9] H. Saggion, K. Bontcheva, and H. Cunningham, ”Robust generic and query based summarization”, In proceedings EACL Conf., pp. 235-238, 2003.

[10] J. Ge, X. Huang, and L.Wu, ”Approaches to event-focused summarization based on named entities and query words”, In proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 281-288, 2004.

[11] J. M. Conroy and J. D. Schlesinger, ”CLASSY query-based multi-document summarization” In proceedings of the Document Understanding Conf. Wksp. 2005 (DUC 2005) at the Human Language Technology Conf./Conf. on Empirical Methods in Natural Language Processing (HLT/EMNLP).

[12] You Ouyanga, Wenjie Lia, Qin Lua, ”Applying regression models to query-focused multi-document summarization”, In Information Processing and Management volume 47, issue 2, pp 227237, March 2011.

[13] DUC. Document understanding conference 2007 (2007), http://www-nlpir.nist.gov/projects/duc.

[14] Chin-Yew Lin, ”ROUGE: A Package for Automatic Evaluation of Summaries”, In Proceedings of Workshop on Text Summarization of ACL, Barcelona, Spain(2004).