Regression model focused on query for multi documents summarization based on significance of the sentence position

Aris Fanani*[1], Yuniar Farida[2], Putra Prima Arhandi[3], M Mahaputra Hidayat[4], Abdul Muhid[5], Billy Montolalu[6]
[1,2,5]UIN Sunan Ampel Surabaya, 117, A Yani St., Surabaya, Indonesia
[3]State Polytechnic of Malang, 9, Soekarno Hatta St., Malang, Indonesia
[4]Bhayangkara University, 114, A Yani St., Surabaya, Indonesia
[6]IT Telkom, Surabaya, Indonesia
*Corresponding author, e-mail: arisfa@uinsby.ac.id[1], yuniar_farida@uinsby.ac.id[2], putraprima@polinema.ac.id[3], mahaputra@ubhara.ac.id[4], abdulmuhid@uinsby.ac.id[5], billy@ittelkom-sby.ac.id[6]

Abstract

Document summarization is needed to get the information effectively and efficiently. One method used to obtain the document summarization by applying machine learning techniques. This paper proposes the application of regression models to query-focused multi-document summarization based on the significance of the sentence position. The method used is the Support Vector Regression (SVR) which estimates the weight of the sentence on a set of documents to be made as a summary based on sentence feature which has been defined previously. A series of evaluations performed on a data set of DUC 2005. From the test results obtained summary which has an average precision and recall values of 0.0580 and 0.0590 for measurements using ROUGE-2, ROUGE 0.0997 and 0.1019 for measurements using the proposed regression-SU4. Model can perform measurements of the significance of the position of the sentence in the document well.

Keywords: *multi-document summarization, sentence position, support vector regresion*

1. Introduction

As internet usage increases, all information becomes easier to obtain and in abundant amounts. For just one topic, so many information documents are displayed with various different narratives even though the core information is the same. Document summarization is needed to get the information effectively and efficiently. In the process of searching documents on web pages, Keyword searches for collections of documents are generally carried out on the entire contents of the document. So the process of information retrieval takes a long time. whereas users expect the right results with a short time in the process of information retrieval. Therefore, it is recommended that the keyword matching process for document collections be carried out at the core of documents that have shorter content. Summarization is needed to get the contents of the article in summary. Summary is a strict expression of the main content of an article, which aims to tell the reader the core of a main thought [1-4]. The simple concept of summary is taking an important part of the entire contents of the article which then presents it again in a more concise form for its users [5]. A good summary should retain the most important contents of the original document or a cluster of related documents, while being coherent, non-redundant and grammatically readable [6].

Basically a summary can be done on one document or several documents. There are different characteristics in making multi-document summarization compared to summarizing single documents, in which multi-document summarization involves many sources of information that overlap and complement each other on several occasions. So, the main task is not only to identify and overcome redundancy in all documents, but also to ensure that the final summary is coherent and complete [3, 5, 7]. This is the background to the need for an automatic summarization system in a document. An Automatic Text Summarization is a computer-based device to produce text that is shorter than the original text but still holds the main points of the summarized text [8-11].

Received February 6, 2019; Revised June 18, 2019; Accepted July 2, 2019

Automatic summarization techniques are divided into two groups: extractive summarization and abstractive summarization [4]. Extractive summary is produced by arranging a few sentences. These sentences are selected exactly as it appears in the original document. On the other hand, abstractive summarization is a more difficult task because it is carried out by paraphrasing source documents. In the research conducted by V. Tohalino and D. R. Amancio [12], using dynamic measurement methods based on complex networks for extractive multi-document summarization methods, which extracts the most central sentences from several textual sources. Meanwhile, research conducted by G. D. Fabbrizio, A. J. Stent and R. Gaizauskas [13] presents the STARLET-H hybrid method as an abstract/extractive summarizer to produce a summary of opinion reviews by combining natural language document with prominent sentence selection techniques.

In another study it was stated that document summarization methods can also be differentiated into generic summarization and query-based summarization [9, 14]. In this study also explained that generic summarization is divided into two parts, namely supervised and unsupervised methods. In the supervised method, training data from a group of people is needed to produce a summary of a document, so that when there are different documents, different training data is needed. This supervised method can only be applied to certain data models. Whereas, in the unsupervised method, summarization does not require training data as like carried out in the supervised method. The research conducted by T. Nayeem, T. A. Fuad and Y. Chali [15] developed an unsupervised abstractive summarization system in multi-document settings. They designed a paraphrastic sentence fusion model which jointly performs sentence fusion and paraphrasing using skip-gram word embedding model at the sentence level. The results showed that this method provides a significant increase in multi-document abstractive summarization.

Several other research related to multi document summarization was conducted by Lin Zhao, et al. [16] who presented about multi-document summarization using extractive summarization methods on query. They propose a query expansion algorithm in a graph-based ranking approach. In addition, Ercan Canhasi et al. [17] also studied the summarization of multi-document that focuses on query using graphical representation based on weighted archetypal analysis. Research conducted by Amini [18] investigate how to use a ranking learning model for single document summarization that focuses on queries and compares the ranking algorithms proposed with the logistic classifier. The ranking algorithm outperforms the logistic classifier.

Another research conducted by You Ouyang [19] successfully developed a regression model to make a summary of many documents that consider queries from users. This study concludes that in making a summary of many documents, the regression model has a better performance than the classification or ranking model. The sentence position feature in this study is assessed based on its global position in a document, so that the sentence at the beginning of the document always has a greater weight than the next sentence. This is considered inappropriate because not all documents have important sentences at the beginning of the document. To overcome this, it is assumed that the sentence in the document that has a high level of significance is the sentence located at the beginning and at the end of the document.

Another study that apply regression in summarize multi document were conducted by [20-22]. Researchers [20] present a fast query-based multi-document summarizer called FastSum based solely on word-frequency features of clusters, documents and topics. Researchers [21] use Integer Linear Programming to jointly maximize the importance of sentences included in the summary and diversity, without exceeding the maximum summary length allowed. To get an important score for each sentence, they use the Support Vector Regression (SVM) model which is trained on summaries written by humans. Researchers [22] use SVM as a supervised learning algorithm for ranking sentences based on score similarities between candidate sentences and benchmark summaries. From several methods used by several researchers above, the authors are interested in applying a regression model in summarizing multi documents because of their simplicity but having a reliable ability to summarize multiple documents. So this paper proposes a regression model to rank sentences in a multi-document summarization that focuses on queries based on the significance of sentence positions.

2. Research Method

The summarization approach proposed is based on feature-based extractive framework, in which ranking and sentence extraction are based on a set of pre-defined sentence features and a combination of assessment functions.

2.1. Feature Design

The sentence in the document is assessed based on the value of its features, so that features have an important role in the assessment and ranking of sentences. The features used in this paper are as follows:

a. Word matching feature

Compare similarities between queries with sentences in documents.

$$f_{word}(s) = \sum_{w_j \in s} \sum_{w_i \in q} same(w_i, w_j) \quad (1)$$

where f is the feature value, q is the query. If the word in the query is the same as the sentence it will be given a value of 1, while if not the same is given a value of 0.

b. Semantic matching feature

Compare similar words between queries and sentences in a document:

$$f_{wordnet}(s) = \sum_{w_j \in s} \sum_{w_i \in q} similarity(w_i, w_j) \quad (2)$$

where f is the value of the similarity between the query and the sentence, q is a query. If the word in the query is the same as the sentence it will be given a value of 1, while if not the same is given a value of 0.

c. Named entity matching feature (query-dependent)

The sliced result of named entity is queried with named entity in the sentence in the document:

$$f_{entity}(s) = |enentity(s) \cap entity(q)| \quad (3)$$

d. Named entity feature

$$f_{entityno} = |entity(s)| \quad (4)$$

where $f_{entityno}$ is number of entity names in sentences.

e. Stop-word penalty feature

Assuming that sentences with many stop-words as less informative sentences:

$$f_{stopword} = |stopword(s)| \quad (5)$$

where $|stopword|$ is number of stop-word in sentences.

f. Sentence position feature

Assuming that the sentence at the beginning and end of the document has more important information, the sentence at the beginning and end of the document has a higher weight than the other sentences.

$$f_{pos} = \begin{cases} 1 - \left(\frac{i-1}{n}\right), & 1 < ip < \frac{ip+j}{2} \\ \left(\frac{i-1}{n}\right), & \frac{ip+j}{2} \le ip \le j \end{cases} \quad (6)$$

where *i* is sentence position on the document, *n* is number of sentences in the document, and *ip* is sentence index.

2.2. Support Vector Regression

SVR is the application of Support Vector Machine (SVM) for regression cases. In the case of regression, the output is a real or continuous number. SVR is a method that can overcome overfitting, so it will produce good performance [23-25]. For example we have *λ* set of training data (*x_j*,*y_j*) where *j = 1,2,...λ* with input $x = \{x_1, x_2, x_3\} \ldots \subseteq \Re^N$ and output $y = \{y_i, \ldots, y_\lambda\} \subseteq \Re$. With SVR, we want to find the function of *f(x)* that have the biggest deviation *ε* of actual target y_i for all of training data. When *ε* is equal to zero (0) then we get perfect regression [23]. For example we have the following function as a regression line:

$$f(x) = W^T\varphi(x) + b \tag{7}$$

where *φ(x)* shows a point in feature space *F* mapping results *x* in input space. Coefficient of *w* and *b* estimated by minimizing the risk function defined in the (8):

$$min\frac{1}{2}||w|^2 + C\frac{1}{\lambda}\sum_{i=1}^{\lambda} L_\in \left(y_i, f(x_i)\right) \tag{8}$$

2.3. Sentence Ranking Method with Regression

The defined feature is used as a combined function to calculate the importance score of a sentence. In this paper, Support Vector Regression (SVR) was adopted to study the assessment function using previously defined features. Regression models are trained from a set of topic D which gives importance score for each sentence. Topics derived from the DUC dataset, each containing query and a set of relevant documents. A sentence in document D is given a score that shows the importance score (s) and a vector of the corresponding F (s) feature. Training data is built by connecting the scores of sentences and features together, that is $\{(score(s), F(s)) \mid S \in D\}$. The target is to predict the score of a new sentence s' in topic D' which is unknown through its vector feature F(s'). This task can be considered as a typical linear regression problem, such as the use of training data $\{(score(s), F(s)) \mid S \in D\}$ to learn the optimal regression function $f: F(s) \to R$ from a set of candidate functions $\{f\ (x) = w\,.x + b \mid w \in Rn, b \in R\}$. For regression problems, linear SVR selects the optimum function $f_0(x) = w_0.x + b_0$ by minimizing the risk function structure.

$$\Phi(w,b) = \frac{1}{2}||w||^2 + C(\frac{1}{|D|}\sum_{s_i \in D} L\big(score(s_i) - (w.F(s_i) + b)\big) \tag{9}$$

where *L(x)* is a loss function, *C* indicates weights to balance factors and |*D*| indicates the number of sentences in *D*. After the regression function f_0 is learned, the results are used to provide an estimate of the importance of the new sentence s

$$score(s') = f_0(F(s')) = w_0.F(s') + b_0 \tag{10}$$

2.4. Establishment of Training Data

To establish training data, a DUC (Document Understanding Conference) 2005 dataset is used where in this dataset there are 50 documents with 25 topics, each topic has a query that is specific to the topic and has 4 summaries of human experts depending on the query given. The initial hypothesis we proposed is: it is increasingly similar between sentences in the human expert summary with the sentence in the document, the better the weight given by the N-gram in the training data formation process. For the D document set and set of human expert summary *H={H1,…,Hm},* each time in D will be given an importance score *(s*|*H).* The score is calculated by probabilistic unigram of *s* to be recognized as a summary sentence given a human summary. By using a bag-of-word model, the probabilistic of unigram in the *i* human summary of *Hi* can be calculated by:

$$p(t|H_i) = freq(t)/|H_i| \tag{11}$$

where *freq(t)* is frequence of *t* in *Hi* and |*Hi*| is number of words on *Hi*. To get the probability of *t* in all human summaries is using the maximum strategy of:

$$p_{\max}(t|H) \ = \max_{\mathrm{H_i \in H}}(\frac{p(t)}{|H_i|}). \tag{12}$$

The overall score of sentence *s* is calculated by summing the probability of unigram:

$$score(s|H) = \sum_{t_j \in s} p(t_j|H) \tag{13}$$

or by analogy, the scoring method is based on unigram as follows:

$$score_{max}(s|H) = \sum_{t_j \in s} \max_{\mathrm{H_i \in H}}(\frac{t_j}{|H_i|}) \tag{14}$$

to calculate the score of a sentence, a combined function is used. It uses the features as mentioned above. In this study used Support Vector Regression (SVR) as a learning tool. The general process of this system can be shown in Figure 1.

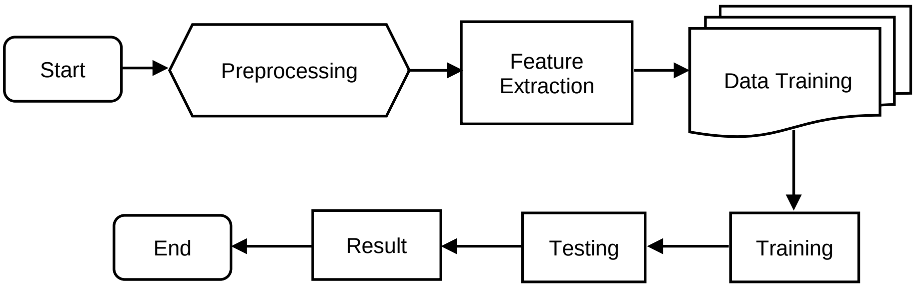

Figure 1. General system diagram

3. Results and Analysis

A series of trials were conducted to obtain a multi-document summarization document that focuses on queries based on the significance of sentence position. The dataset used is the DUC (Document Understanding Conference) 2005. This dataset is used because it consists of 10 topics, with each topic consisting of 30-50 news documents and 4 kinds of human summary results. This dataset can be downloaded at the link http://www-nlpir.nist.gov/projects/duc/duc2005/.

In all trials, queries and documents are preprocessed by eliminating stopword and stemming. The system created will be limited to produce a summary with a word length of 250 words. After ranking the sentence, the sentence with the highest score will be chosen from the original document to be used as a summary until the limit of the summary sentence is reached which are 250 words.

In this paper, two DUC automatic evaluation criteria, ROUGE-2 and ROUGE-SU4, are used to compare the summary results obtained from a system built with a summary made by humans. ROUGE-2 and ROUGE-SU4 are used because these two criteria are the official evaluation values of ROUGE. ROUGE (Recall Oriented Understudy for Gisting Evaluationa) [26] is an automatic summarization evaluation method that utilizes the N-gram ratio. For example, ROUGE-2 evaluates the summary results of the system by matching Bi-gram with a human summary, i.e.:

$$R_n(s) = \frac{\sum_{j=1}^{h} \sum_{t_i \in s} Count(\ t_i|S, H_j)}{\sum_{j=1}^{h} \sum_{t_i \in s} Count(\ t_i|H_i)} \tag{14}$$

where *S* is the summary that will be evaluated, H_j *(j = 1, 2, ..., h)* is a human summary which is considered as a standard summary, *ti* shows Bi-gram in summary *S*, *Count* $(t_i|H_j)$ is number of occurrences Bi-gram t_i that happens in the human summary of the *j* in H_j and *Count (ti |S, Hj)* is the number of occurrences of t_i that occur in *S* and H_j. ROUGE-SU4 is the same as ROUGE-2. ROUGE-SU4 matches Uni-grams and ignores the Bi-gram summary of human summaries.

In this study two experiments were conducted to measure the reliability of regression models in multi document summarization based on the significance of sentence position. The first experiment was carried out by using all the features that were defined in section 2.1, while the second experiment was carried out without entering the sentence position feature. The two experiments above were carried out to find out how effective the summarization system was by paying attention to the significance of the position of the important sentences in the document. Table 1 shows the results of average ROUGE-2 and ROUGE-SU4 with the 95% Confidential Interval (CI) suitability level:

Table 1. The Results of the Evaluation of the Application of Different Features in the Dataset DUC 2005 (*CI* = 95%)

Evaluation	Fiture	*Precision* (*CI*)	*Recall* (*CI*)
Rouge-2	All	0.0580 (0.0347-0.1005)	0.0590 (0.0344-0.1034)
	Without f_{pos}	0.0576 (0.0328-0.1005)	0.0585 (0.0344-0.1034)
Rouge-SU4	All	0.0997 (0.0636-0.1414)	0.1019 (0.0684-0.1384)
	Without f_{pos}	0.0994 (0.0683-0.1414)	0.1015 (0.0689-0.1384)

4. Conclusion

In this paper, we design the application of regression models to query-focused multi-document summarization based on the significance of the sentence position. This method using Support Vector Regression (SVR) which estimates the weight of the sentence on a set of documents to be made as a summary based on sentence feature which has been defined previously. A series of evaluations performed on a data set of DUC 2005. From the test results obtained summary which has an average precision and recall values of 0.0580 and 0.0590 for measurements using ROUGE-2, ROUGE 0.0997 and 0.1019 for measurements using the proposed regression-SU4. Model can perform measurements of the significance of the position of the sentence in the document well. This also shows the proposed summarization system has better precision and recall values.

References

[1] Sartuni, Rasjid, et al. Indonesian for Higher Education (in Indonesia: Bahasa Indonesia untuk Perguruan Tinggi). Jakarta: Nina Dinamika. 1984.

[2] Wang L, Raghavan H, Castelli V, Florian R, Cardie C. A Sentence Comparession Based Framework to Query-Focused Multi-Document Summarization. 2016.

[3] Kumar YJ, Salim N. Automatic Multi Document Summarization Approaches. *Journal of Computer Sciences*. 2012; 8(1): 133-140.

[4] Haghighi A, Vanderwende L. *Exploring Content Models for Multi-Document Summarization*. Human Language Technologies: The 2019 Annual Conference of the North American Chapter of the ACL. 2009: 362-370.

[5] Mani I, Maybury MT. Advance in Automatic Text Summarization. Cambridge: The MIT Press.

[6] Nayeem MT, Fuad TA, Chali Y. *Abstractive Unsupervised Multi-Document Summarization using Paraphrastic Sentence Fusion*. Proceedings of the 27th International Conference on Computational Linguistics. Santa Fe. 2018.

[7] Cao Z, Li W, Li S, Wei F. *Improving Multi-Document Summarization via Text Classification*. Proceedings of the Thirty-First AAAI Conference on Artifical Intelegence. 2017: AAAI-17.

[8] Dallianis H. GSLT: Natural Language Generation Spring. 2005.

[9] Lukmana I, Swanjaya D, Kurniawardhani A, Arifin AZ, Purwitasari D. Multi-Document Summarization Based on Sentence Clustering Improved Using Topic Words. *JUTI*: *Jurnal Ilmiah Teknologi Informasi*. 2014; 12(2) :1-8.

[10] Yih WT, Goodman J, Vanderwende L, Suzuki H. *Multi-Document Summarization by Maximizing Informative Content-Words*. Proceedings of The 20th International Joint Conference on Artificial Intelligents. 2007: 1776-1782.

[11] Bysani P, Reddy VB, Varma V. *Modeling Novelty and Feature Combination using Support Vector Regression for Update Summarization*. Proceedings of ICON-2009: 7th International Conference on Natural Language Processing. 2009: 41.

[12] Tohalino JV, Amancio DR. *Extractive Multi Document Summarization using Dynamical Measurements of Complex Networks*. 2017 Brazilian Conference on Intelligent Systems (BRACIS). 2017: 366-371.

[13] Di Fabbrizio G, Stent A, Gaizauskas R. *A Hybrid Approach to Multi-document Summarization of Opinions in Reviews*. Proceedings of the 8th International Natural Language Generation Conference. 2014: 54-63.

[14] Lee JH, Park S, Ahn CM, Kim D. Automatic Generic Document Summarization based on Non-Negative Matrix Factorization. *Information Processing and Management*. 2009; 45(1): 20-34.

[15] Nayeem MT, Fuad TA, Chali Y. *Abstractive Unsupervised Multi-Document Summarization using Paraphrastic Sentence Fusion*. Proceeding of the 27th International Conference on Computational Linguistics. 2018: 1191-1204.

[16] Lin CY. *ROUGE: A package for automatic evaluation of summaries*. Text Summarization Branches Out-Proceedings of the ACL Workshop. 2004: 74-81.

[17] Canhasi E, Kononenko I. Weighted archetypal analysis of the multi-element graph for query-focused multi-document summarization. *Expert Systems with Applications*. 2014; 41(2): 535-43.

[18] Amini MR, Usunier N, Gallinari P. *Automatic Text Summarization based on Word-Clusters and Ranking Algorithms*. ECIR. In D. E. Losada & J.M. 2005; 3408: 142-156.

[19] Ouyang Y, et al. Applying Regression Models to Query-Focused Multi-Document Summarization. *Information Processing and Management*. 2011; 47(2): 227-37.

[20] Schilder F, Kondadadi R. *Fast and accurate query-based multi-document summarization*. Proceedings of ACL-08: HLT, Short Papers (Companion Volume). 2008: 205–208.

[21] Galanis D, Lampouras G, Androutsopoulos I. *Extractive Multi-Document Summarization with Integer Linear Programming and Support Vector Regression*. Proceedings of COLING 2012: Technical Papers. 2012: 911–926.

[22] Dlikman A, Last M. Last. *Using Machine Learning Methods and Linguistic Features in Single-Document Extractive Summarization*. Proceedings of DMNLP, Workshop at ECML/PKDD. Riva del Garda. 2016: 1-8.

[23] Santosa B. Applied Data Mining using Matlab (in Indonesia: Data Mining Terapan dengan Matlab). Yogyakarta: Graha Ilmu. 2011.

[24] Alkaff M, Khatimi H, Puspita W, Sari Y. Modelling and predicting wetland rice production using support vector regression. *TELKOMNIKA Telecommunication Computing Electronics and Control*. 2019; 17(6): 819-825.

[25] Harabagiu S, Lacatusu F. Using Topic Themes for Multi-Document Summarization. *ACM Transactions on Information Systems*. 2010; 28(3): 13.

[26] Lin CY, Hovy E. *Manual and Automatic Evaluation of Summaries*. Proceedings of the ACL-02 Workshop on Automatic Summarization. 2002; 4: 45-51.