khazanah informatika

Jurnal Ilmu Komputer dan Informatika

Combination of Graph-based Approach and Sequential Pattern Mining for Extractive Text Summarization with Indonesian Language

Dian Sa'adillah Maylawati [1*], Yogan Jaya Kumar [2], Fauziah Binti Kasmin [3]
*diansm@uinsgd.ac.id
[1,2,3]Centre for Advanced Computing Technology, Faculty of Information and Communication Technology
Universiti Teknikal Malaysia Melaka
Melaka, Malaysia
[1]Department of Informatics, Faculty of Science and Technology
UIN Sunan Gunung Djati Bandung
Bandung, Indonesia

Abstract-The great challenge in Indonesian automatic text summarization research is producing readable summaries. The quality of text summary can be reached if the meaning of the text can be maintained properly. As a result, the purpose of this study is to improve the quality of extractive Indonesian automatic text summarization by considering the quality of structured text representation. This study employs Sequential Pattern Mining (SPM) to generate a sequence of words as a structured representation of text and a graph-based approach to generate automatic text summarization. The SPM algorithm used is PrefixSpan, and the graph-based approach uses the Bellman-Ford algorithm. The results of an experiment using the IndoSum dataset show that combining SPM and Bellman-Ford can improve the precision, recall, and f-measure of ROUGE-1, ROUGE-2, and ROUGE-L. When Bellman-Ford is combined with SPM, the F-measure of ROUGE-1 increases from 0.2299 to 0.3342. The ROUGE-2 f-measure increases from 0.1342 to 0.2191, and the ROUGE-L f-measure increases from 0.1904 to 0.2878. This result demonstrates that SPM can improve the performance of the Bellman-Ford algorithm in producing Indonesian text summaries.

Keywords: automatic text summarization, Bellman-Ford algorithm, graph-based approach, Indonesian language, prefixspan, sequence of words, sequential pattern mining

Article info: *submitted January, 19, 2023, revised March, 17, 2023, accepted July, 29, 2023*

1. Introduction

Automatic text summarization is a part of Natural Language Processing (NLP) that growing rapidly today. The challenge in NLP research is a language because every language is unique, including Indonesian language. In general, there are two types of automatic text summarization: extractive and abstractive [1]. Extractive summarization generates a sequential summary based on the source document and without changing the sentence word structure [2], [3]. The final summary is made up of sentences from the source document. Extractive summaries are created by detecting and directly selecting key sentences in the source content. While abstractive summarization yields a revised summary [4], for example paraphrase. As a result, the abstracted summary lacks the same phrases and structure as the original document, but it conveys the same message.

Graph-based approach is one of common extractive text summarization method. The previous related works that used graphs for text summarization are as follows: (1) Inductive method in the Indonesian language, and the Shortest Path Algorithm (Dijkstra) to provide a summary path from the first sentence to the last sentence of each paragraph within an article [5]; (2) The TextRank algorithm is used to automatically summarize text documents, where sentences in the text are represented in a graph, and the value of each sentence is calculated using the similarity between sentences to determine the summary results [6]–[8]; (3) Graph Convolutional Networks are used to summarize Indonesian news material using a word embedding sequence and a sentence relationship graph [9]graph construction, sentence scoring, and sentence selection components. Sentence scoring component is a neural network that uses Recurrent Neural Network and GCN to produce scores for all sentences. This study used three different representation types for the sentence relationship graph. The sentence selection component then generates a summary with two different techniques: by greedily choosing sentences with the highest scores and by using the Maximum Marginal Relevance (MMR; (4) heterogeneous graph neural networks for extractive text summarization [10]; (5) LexRank algorithm for Indonesian text summarization [11], [12]; and so on. Several studies of automatic text summarization also have found that graph theory like the Bellman-Ford algorithm remains useful. Bellman-Ford algorithm already used for Indonesian text summarization with an f-measure score of ROUGE-L is 0.495 for the *Hadith* dataset [13] and 0.72 for multiple article journal as a dataset [14]. Based on the results of previous research, Bellman-Ford is quite good at automatically summarizing Indonesian texts. Therefore, this research decides to use the Bellman-Ford algorithm as a graph-based method to produce an Indonesian text summary.

However, the great challenge in automatic text summarization is how to produce a readable summary [15]–[17]. We believe that a good-structured text representation will better preserve the meaning of the text, to improve the quality of the summary results, including the readability aspect. Therefore, to resolve the readability issue from the summary results, it can be started from preparing a quality text representation. Text representation that used must be able to maintain the meaning of the text data. Sequence of words is one of structured text representation that has been proven can maintain the meaning of text well in the document classification and clustering studies [18]–[20]Set of Frequent Word Sequence (SFWS. Sequential Pattern Mining (SPM) is a method to produce sequence of words. Therefore, this research proposes to integrate graph-based approach using Bellman-Ford algorithm and SPM using PrefixSpan algorithm in generating Indonesian text summary, automatically. This research will investigate the performance of SPM when it is combined with graph-based in producing Indonesian text summary. Moreover, this research contributes to NLP technology, especially graph-based automatic text summarization by preparing a structured text representation using SPM to maintain the readability of the results of Indonesian text summaries.

2. Methods

1. Research Overview

Figure 1 presents the research overview that combine SPM and graph-based approach to produce Indonesian automatic text summary. This research has several activities, begin from data gathering and data preparation, producing Indonesian automatic text summary using Bellman-Ford algorithm, producing Indonesian automatic text summary using combination of SPM and Bellman-Ford algorithm, evaluate the ROUGE (Recall-Oriented Understudy for Gisting Evaluation) value of summary result from Bellman-Ford with and without SPM, then comparative analysis whether SPM can enhance the quality of Indonesian automatic text summary result.

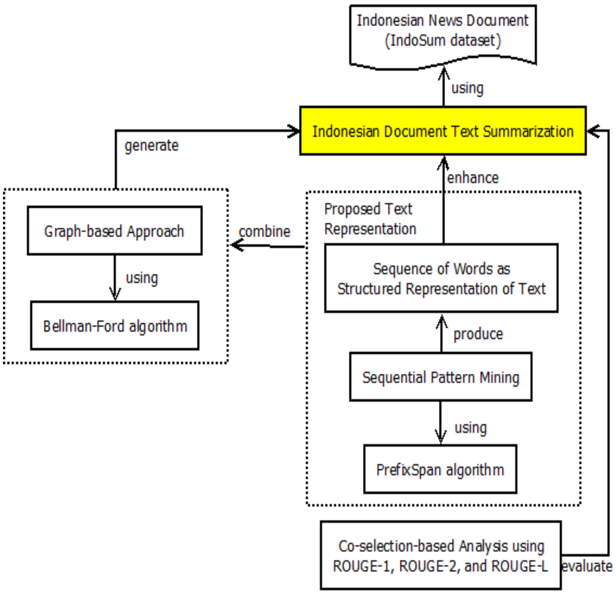

Indonesian News Document (IndoSum dataset)
using
Indonesian Document Text Summarization
generate
enhance
Graph-based Approach
using
Bellman-Ford algorithm
combine
Proposed Text Representation
Sequence of Words as Structured Representation of Text
produce
Sequential Pattern Mining
using
PrefixSpan algorithm
Co-selection-based Analysis using ROUGE-1, ROUGE-2, and ROUGE-L
evaluate

Figure 1. Research overview

This research uses IndoSum dataset that contain news articles from CNN Indonesia, Kumparan, Merdeka.com, etc [21]. This dataset is widely used as a benchmark in Indonesian research on automatic text summarization. In addition, the dataset includes a manual summary created by two Indonesian native speakers as experts. The dataset is open source and can be found at https://github.com/kata-ai/indosum. The dataset of news articles is divided into six categories: entertainment, inspiration, sport, showbiz, headline, and technology. Actually, the total number of news articles from those sources is 18.774. JSON-formatted collections of Indonesian news documents are provided.

Data preparation or data pre-processing that significant steps in most computational linguistics investigations [22]. Text data pre-processing ensures that they have a suitable representation and can be used effectively in experiments. In this study, several text preprocessing procedures are used, including sentence separation, case-folding, tokenizing, removing regular expressions (non-letter characters), removing Indonesian topwords, and stemming using the Nazief-Adriani algorithm for the Indonesian language. Furthermore, the Sastrawi Python library can be used to prepare Indonesian text data during the text pre-processing phase [23]. Sastrawi provides the Indonesian stop words list and Nazief-Adriani as a popular stemming process for Indonesian text [23], [24]. However, sometimes text pre-processing activity is not the same for all cases, depending on the need in each case. Sentence separation aims to distinguish one sentence from another according to the needs of the feature approach using graph-based approach.

b. Experimental Setting and Evaluation

The experiment covers the text pre-processing, including separating sentences, tokenizing, lowering case, removing character non-letter and regular expression, removing Indonesian stopwords, and stemming process using Nazief and Adriani algorithm for the Indonesian language. Stop-words removal and stemming process used Sastrawi library is commonly used for NLP research with Indonesian text [23]. In addition, the PrefixSpan algorithm as a part of the SPM method, used the prefixspan library for Python.

Then, there are two experiment scenarios: (1) produce Indonesian text summary with graph-based approach using Bellman-Ford algorithm and Term-Frequency and Inverse Document Frequency (TF-IDF) as structured representation of text; and (2) produce Indonesian text summary with combination of graph-based approach using Bellman-Ford and PrefixSpan algorithm as a part of SPM which produces sequence of words as structured representation of text. All results of experiments were evaluated using ROUGE evaluation metrics. ROUGE evaluation in this experiment uses the rouge-score library for Python. Lastly, the result of the experiments will be interpreted and analyzed in the discussion section.

The performance of Bellman-Ford with and without SPM are evaluated using co-selection-based analysis with ROUGE-1, ROUGE-2, and ROUGE-L. ROUGE also evaluate the Precision, Recall, and F-1 score. ROUGE-1 and ROUGE-2 is a part of ROUGE-N, where N=1, 2, 3, and 4. ROUGE-N score evaluation use n-gram overlaps between the candidate document and the reference documents. ROUGE-N formula is available in formula (1) [25]word sequences, and word pairs between the computer-generated sum- mary to be evaluated and the ideal summaries cre- ated by humans. This paper introduces four different ROUGE measures: ROUGE-N, ROUGE-L, ROUGE-W, and ROUGE-S included in the ROUGE summariza- tion evaluation package and their evaluatio ns. Three of them have been used in the Document Under- standing\\tConference\\t(DUC.

$$ROUGE - N = \frac{\sum_{S\in\{Reference\ Summaries\}}\sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S\in\{Reference\ Summaries\}}\sum_{gram_n \in S} Count(gram_n)} \qquad (1)$$

In which *n* is the length of an *n-gram, gram$_n$* is the maximum number of *n-grams* found in a candidate summary and a set of reference summaries, and *count$_{match}$(gram$_n$)* is the maximum number of *n-grams* found in a candidate summary and a set of reference summaries.

While ROUGE-L is statistics based on the Longest Common Subsequence (LCS). The longest common subsequence problem automatically discovers the longest co-occurring in sequence n-grams by taking into consideration sentence level structure similarities. LCS has the advantage of not requiring consecutive matches but rather in-sequence matches that reflect sentence level word order. It does not require a predetermined n-gram length because it automatically includes the longest in-sequence common n-grams. ROUGE-N formula is available in formula (2).

$$ROUGE - L = \frac{LCS(gram_n)}{Count(gram_n)} \qquad (2)$$

Where *LCS(gram$_n$)* is longest common subsequence between reference and model output. Then, *gram$_n$* is the maximum number of *n-grams* found in a candidate summary and a set of reference summaries.

3. Result

This section presents the outcome of enhancing the sequence of words as a text representation to improve the Indonesian text summary. To achieve a readable summary result, it is critical to properly prepare the text representation. This section presents the process of combining SPM and Bellman-Ford in sequential order, the result of Bellman-Ford in producing Indonesian text summary, and the result of combining SPM and Bellman-Ford.

a. PrefixSpan algorithm in producing sequence of words

PrefixSpan, also known as Prefix-projected Sequential Pattern Mining, is an SPM algorithm that uses divide and conquer principles and pattern building to generate efficient sequence patterns from large sequence databases [26]time-series databases, and many other kinds of databases has been studied popularly in data mining research. Most of the previous studies adopt an Apriori-like candidate set generation-and-test approach. However, candidate set generation is still costly, especially when there exist a large number of patterns and/or long patterns. In this study, we propose a novel frequent-pattern tree (FP-tree. Here is the PrefixSpan algorithm:

```
Algorithm: PrefixSpan

Input: A sequence database S and the minimum support (threshold min_support)

Output: The complete set of sequential patterns

Method: Call PrefixSpan({},0,S)

Subroutine: PrefixSpan(α,l,S|α)
```

```
The parameters are 1) $\alpha$ is a sequential pattern; 2) l is the length of $\alpha$; and 3) $S|_{\alpha}$ is the
$\alpha$-projected database if $\alpha \neq \{\}$, otherwise, it is the sequence database S.

Method:

1. Scan $S|_{\alpha}$ once, find each frequent item, b, such that
(a) b can be assembled to the last element of $\alpha$ to form a sequential pattern; or
(b) {b} can be appended to $\alpha$ to form a sequential pattern.
2. For each frequent item b, append it to $\alpha$ to form a sequential pattern $\alpha'$ and output $\alpha'$.
3. For each $\alpha'$, construct $\alpha'$-projected database $S|_{\alpha}$ and call PrefixSpan($\alpha'$,l+1,$S|_{\alpha'}$)
```

According to the algorithm, PrefixSpan will first find a length-1 sequential pattern before dividing the search space. Then, for each prefix, find subsets of sequential patterns. The collection of patterns discovered during the recursive mining process is known as the set of sequential patterns. PrefixSpan on text data is used to create a structured text representation in the form of a SoW where the pattern of words that appear considers the order in which the words appear. For example, the following news documents in Indonesian text:

Jakarta, CNN Indonesia -- Gubernur Jawa Barat Ridwan Kamil meminta kepolisian untuk memperketat pintu masuk di wilayah perbatasan, termasuk jalan-jalan tikus. Menurut Ridwan Kamil, Pembatasan Sosial Berskala Besar (PSBB) di wilayah Bandung raya yang sudah diberlakukan sejak Rabu (22/4), masih ditemui sejumlah pelanggaran. Salah satu titik yang harus diperbaiki adalah wilayah perbatasan."Mulai sekarang kita perketat penjagaan di perbatasan, tidak boleh ada warga yang masuk maupun keluar dari wilayahnya, kecuali dengan alasan yang jelas," kata pria yang karib disapa Emil itu saat menggelar pertemuan video conference dari Gedung Pakuan, Kota Bandung, Sabtu (25/4).

Jakarta, CNN Indonesia - West Java Governor Ridwan Kamil has asked the police to tighten entrances in the border area, including rat roads. According to Ridwan Kamil, Large-scale Social Restrictions (PSBB) in the Bandung area, which had been imposed since Wednesday (22/4), a few violations were still encountered. One of the points that need to be improved is the border area. "From now on, we tighten security at the border, no residents may enter or leave the area, except for obvious reasons," said the close friend Emil when holding a video conference meeting from Pakuan Building, Bandung City, Saturday (25/4).

Before the news text is processed in a sequential pattern layer, the news text is prepared and cleaned at the pre-processing stage, including case folding, removing regular expression, removing stop words, stemming, as well as separating sentence for the needs of the sentence classification to be selected in the result of summary. From the news text example above, the following results are obtained for text pre-processing and sequential patterns, with the minimum support is 25% (it means the minimum appearance frequency of SoW is two), the SoW that can be produced:

<jakarta cnn indonesia>
<gubernur jawa barat ridwan kamil pinta polisi ketat pintu masuk wilayah batas masuk jalan jalan tikus>
<turut ridwan kamil batas sosial skala besar psbb wilayah bandung raya sudah laku sejak rabu masih temu jumlah langgar>
<salah satu titik harus baik wilayah batas>
<mulai ketat jaga batas boleh ada warga masuk mau keluar wilayah alas jelas kata pria karib sapa emil gelar temu video conference gedung pakuan kota bandung sabtu>

The sequence of words that produced:
<{ridwan}>: 2
<{kamil}>: 2
<{ketat}>: 2
<{temu}>: 2
<{masuk}>: 3
<{wilayah}>: 4
<{batas}>: 4
<{jalan}>: 2
<{ridwan, kamil}>: 2
<{ketat, masuk}>: 2
<{ketat, wilayah}>: 2

<{masuk, wilayah}>: 2
<{batas, wilayah}>: 2
<{wilayah, batas}>: 2
<{ketat, masuk, wilayah}>: 2

b. Producing Indonesian text summary using Bellman-Ford algorithm

As the shortest path-finding technique built on various previous studies, the Bellman-Ford algorithm provides great accuracy and efficiency [27]; [28]; [29]. After the pre-processing text, then a calculation of sentence weight is conducted using TF-IDF with equations (3), (4), and (5). Where *d* is the document and *t* is the words in the document.

$$tf(t,d) = \frac{count\ of\ t\ in\ d}{number\ of\ words\ in\ d} \tag{3}$$

$$idf(t) = \log\left(\frac{number\ of\ documents}{occurance\ of\ t\ in\ documents+1}\right) \tag{4}$$

$$tfidf(t,d) = tf(t,d) * idf(t) \tag{5}$$

Then, after conducting text pre-processing and computing TF-IDF for sentence weight, the next step is to create a graph that will be used to calculate the similarities between sentences. The following equation (6) calculates the sentence's similarity: *i* is the first sentence, *j* is the second sentence, *overlap* is the number of similar words between the first and second sentences, and *weight* is the sentence's weight.

$$Cost_{i,j} = \frac{(i-j)^2}{overlap_{i,j} \times weight_j} \tag{6}$$

The smaller the cost value between the two sentences, the more similar they are. After forming the graph, the extraction will be summarized by seeking the shortest path between the first and last sentences. In this case, the Bellman-Ford technique is utilized to implement the shortest path search on document graphs. The Bellman-Ford algorithm is implemented to determine the closeness between sentences and generate a summary finding.

With the text example from section 2, the TF-IDF weighting of each sentence is carried out after the text preprocessing step is completed. The number of words in a document (term frequency) and the number of occurrences in a collection of documents are used to determine the weighting (inverse document frequency). The amounts of words in the document impact the weight value. The higher the number of words in the document, the higher the weight value. Table 1 illustrates the example of TF-IDF. Then, the next process is to calculate the cost value between sentences using equation (6), in which, for the example of text above, the overlap between sentences is presented in Table 2 while the cost value between sentences is presented in Table 3.

Table 1. TF-IDF calculation process

Word	Appearance in Sentence					*df*	*idf*	*tf-idf* (using equation (5))				
	S1	S2	S3	S4	S5			S1	S2	S3	S4	S5
jakarta	1	0	0	0	0	1	0.699	0.699	0.000	0.000	0.000	0.000
cnn	1	0	0	0	0	1	0.699	0.699	0.000	0.000	0.000	0.000
indonesia	1	0	0	0	0	1	0.699	0.699	0.000	0.000	0.000	0.000
gubernur	0	1	0	0	0	1	0.699	0.000	0.699	0.000	0.000	0.000
jawa	0	1	0	0	0	1	0.699	0.000	0.699	0.000	0.000	0.000
barat	0	1	0	0	0	1	0.699	0.000	0.699	0.000	0.000	0.000
ridwan	0	1	1	0	0	2	0.398	0.000	0.699	0.699	0.000	0.000
kamil	0	1	1	0	0	2	0.398	0.000	0.699	0.699	0.000	0.000
pinta	0	1	0	0	0	1	0.699	0.000	0.699	0.000	0.000	0.000
polisi	0	1	0	0	0	1	0.699	0.000	0.699	0.000	0.000	0.000
ketat	0	1	0	0	1	2	0.398	0.000	0.699	0.000	0.000	0.699
pintu	0	1	0	0	0	1	0.699	0.000	0.699	0.000	0.000	0.000
wilayah	0	1	1	1	1	4	0.097	0.000	0.699	0.699	0.699	0.699
batas	0	1	1	1	1	4	0.097	0.000	0.699	0.699	0.699	0.699
masuk	0	1	0	0	1	2	0.398	0.000	0.699	0.000	0.000	0.699
jalan	0	1	0	0	0	1	0.699	0.000	0.699	0.000	0.000	0.000
tikus	0	1	0	0	0	1	0.699	0.000	0.699	0.000	0.000	0.000

Word	**Appearance in Sentence**					*df*	*idf*	***tf-idf* (using equation (5))**				
	S1	**S2**	**S3**	**S4**	**S5**			**S1**	**S2**	**S3**	**S4**	**S5**
turut	0	0	1	0	0	1	0.699	0.000	0.000	0.699	0.000	0.000
sosial	0	0	1	0	0	1	0.699	0.000	0.000	0.699	0.000	0.000
skala	0	0	1	0	0	1	0.699	0.000	0.000	0.699	0.000	0.000
besar	0	0	1	0	0	1	0.699	0.000	0.000	0.699	0.000	0.000
psbb	0	0	1	0	0	1	0.699	0.000	0.000	0.699	0.000	0.000
bandung	0	0	1	0	1	2	0.398	0.000	0.000	0.699	0.000	0.699
raya	0	0	1	0	0	1	0.699	0.000	0.000	0.699	0.000	0.000
sudah	0	0	1	0	0	1	0.699	0.000	0.000	0.699	0.000	0.000
laku	0	0	1	0	0	1	0.699	0.000	0.000	0.699	0.000	0.000
sejak	0	0	1	0	0	1	0.699	0.000	0.000	0.699	0.000	0.000
rabu	0	0	1	0	0	1	0.699	0.000	0.000	0.699	0.000	0.000
masih	0	0	1	0	0	1	0.699	0.000	0.000	0.699	0.000	0.000
temu	0	0	1	0	1	2	0.398	0.000	0.000	0.699	0.000	0.699
jumlah	0	0	1	0	0	1	0.699	0.000	0.000	0.699	0.000	0.000
langgar	0	0	1	0	0	1	0.699	0.000	0.000	0.699	0.000	0.000
salah	0	0	0	1	0	1	0.699	0.000	0.000	0.000	0.699	0.000
satu	0	0	0	1	0	1	0.699	0.000	0.000	0.000	0.699	0.000
titik	0	0	0	1	0	1	0.699	0.000	0.000	0.000	0.699	0.000
harus	0	0	0	1	0	1	0.699	0.000	0.000	0.000	0.699	0.000
baik	0	0	0	1	0	1	0.699	0.000	0.000	0.000	0.699	0.000
mulai	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
jaga	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
boleh	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
ada	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
warga	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
mau	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
keluar	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
alas	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
jelas	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
kata	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
pria	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
karib	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
sapa	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
emil	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
gelar	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
video	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
conference	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
gedung	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
pakuan	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
kota	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
sabtu	0	0	0	0	1	1	0.699	0.000	0.000	0.000	0.000	0.699
	Weight of Sentence							2.10	9.79	13.28	4.89	18.87

Table 2. Overlap between sentences

Sentence No.	S1	S2	S3	S4	S5
S1	■	0	0	0	0
S2	0	■	4	2	4
S3	0	4	■	2	4
S4	0	2	2	■	2
S5	0	4	4	2	■

Table 3. Cost value between sentences

Sentence No.	Cost Value (using equation (6))
S1-S2	$\frac{(1-2)^2}{0\times9.79}=\infty$
S1-S3	$\frac{(1-3)^2}{0\times13.28}=\infty$
S1-S4	$\frac{(1-4)^2}{0\times4.89}=\infty$
S1-S5	$\frac{(1-5)^2}{0\times18.87}=\infty$
S2-S3	$\frac{(2-3)^2}{4\times13.28}=0.018825301$
S2-S4	$\frac{(2-4)^2}{2\times4.89}=0.408997955$
S2-S5	$\frac{(2-5)^2}{4\times18.87}=0.119236884$
S3-S4	$\frac{(3-4)^2}{2\times4.89}=0.102249489$
S3-S5	$\frac{(3-5)^2}{4\times18.87}=0.052994171$
S4-S5	$\frac{(4-5)^2}{2\times18.87}=0.026497085$
S2-S1	$\frac{(2-1)^2}{0\times2.10}=\infty$
S3-S1	$\frac{(3-1)^2}{0\times2.10}=\infty$
S4-S1	$\frac{(4-1)^2}{0\times2.10}=\infty$
S5-S1	$\frac{(5-1)^2}{0\times2.10}=\infty$
S3-S2	$\frac{(3-2)^2}{4\times9.79}=0.025536261$
S4-S2	$\frac{(4-2)^2}{2\times9.79}=0.204290092$
S5-S2	$\frac{(5-2)^2}{4\times9.79}=0.229826353$
S4-S3	$\frac{(4-3)^2}{2\times13.28}=0.037650602$
S5-S3	$\frac{(5-3)^2}{4\times13.28}=0.075301205$
S5-S4	$\frac{(5-4)^2}{2\times4.89}=0.102249489$

Cost values with infinite values will be discarded, so there is no connected edge between sentences. After calculating the distance between sentences, the next process is to form a graph based on the number of sentences in the document and the distance between sentences. The graph consists of nodes and edges. In this research, sentences are represented as nodes, and edges represent the distance between sentences. Figure 2 illustrates the application of graph formation to a sample document that has done text preprocessing and TF-IDF weighting to calculate the distance between sentences using the cost value equation, in which the S1 is removed because the cost value between S1 and the others sentence is infinite. The cost value indicates how similar the two sentences are. The more similarity between sentences, the lower the cost value between sentences and vice versa.

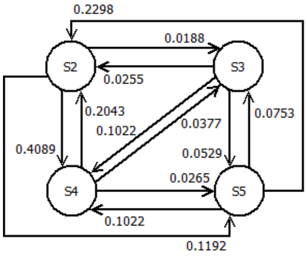
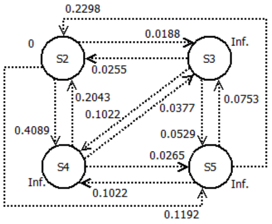
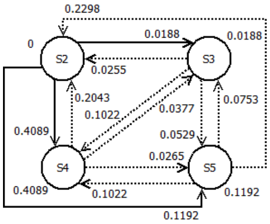


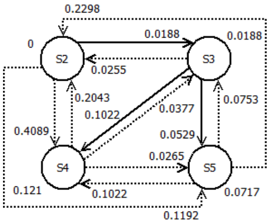
Figure 2. Illustration of sentence graph

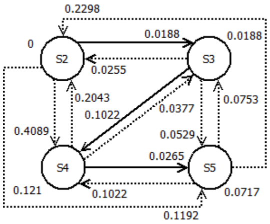
Table 4 provides the iteration of the Bellman-Ford algorithm process to find the sentences for the summary result. In the first iteration, the origin none is assigned zero (0), while the other nodes are assigned infinity (∞). In this example, four sentences are connected to each other, so there are three iterations to form the most optimum path and a summary result. In the iteration process, if the weight of the destination point is greater than the distance between the origin and destination plus the weight of the origin, the destination point will be assigned a value for the sum of the distance from the origin to the destination point and the weight of the origin.

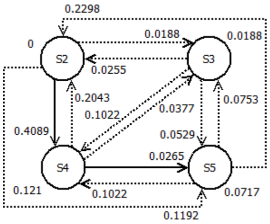
Table 4. Bellman-Ford process

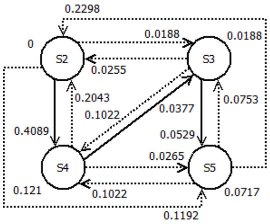
<table>
<tr><th>No</th><th>Bellman-Ford Process</th><th>Graph Illustration</th></tr>
<tr><td>1</td><td><table><tr><td>S2</td><td>S3</td><td>S4</td><td>S5</td></tr><tr><td>0</td><td>∞</td><td>∞</td><td>∞</td></tr></table></td><td>
</td></tr>
</table>

<table>
<tr><th>No</th><th>Bellman-Ford Process</th><th>Graph Illustration</th></tr>
<tr><td>2</td><td>
<table>
<tr><td>S2</td><td>S3</td><td>S4</td><td>S5</td></tr>
<tr><td>0</td><td>0.0188</td><td>0.4089</td><td>0.1192</td></tr>
<tr><td>0</td><td>0.0188</td><td>0.121</td><td>0.0717</td></tr>
<tr><td>0</td><td>0.0188</td><td>0.121</td><td>0.0717</td></tr>
</table>
</td><td>






</td></tr>
<tr><td>3</td><td>
<table>
<tr><td>S2</td><td>S3</td><td>S4</td><td>S5</td></tr>
<tr><td>0</td><td>0.0188</td><td>0.1210</td><td>0.0717</td></tr>
<tr><td>0</td><td>0.0188</td><td>0.1210</td><td>0.0717</td></tr>
</table>
</td><td>




</td></tr>
</table>

Based on the result of Bellman-Ford above, the path that is passed from the starting point S2 to the end point S5 with the minimum number of weights is S3 and S5. Therefore, the summary results obtained are sentence 2 (S2), sentence 3 (S3) and sentence 5 (S5). The result is as follows: "*Gubernur Jawa Barat Ridwan Kamil meminta kepolisian untuk memperketat pintu masuk di wilayah perbatasan, termasuk jalan-jalan tikus. Menurut Ridwan Kamil, Pembatasan Sosial Berskala Besar (PSBB) di wilayah Bandung raya yang sudah diberlakukan sejak Rabu (22/4), masih ditemui sejumlah pelanggaran. "Mulai sekarang kita perketat penjagaan di perbatasan, tidak boleh ada warga yang masuk maupun keluar dari wilayahnya, kecuali dengan alasan yang jelas," kata pria yang karib disapa Emil itu saat menggelar pertemuan video conference dari Gedung Pakuan, Kota Bandung, Sabtu (25/4).*"

c. Producing Indonesian text summary using combination of Bellman-Ford algorithm and PrefixSpan

In the Bellman-Ford process, SPM will replace the TF-IDF process. Table 5 provides a mapping of all the sequence of word which is provided in point 1.

Table 5. Mapping of Sequence of Word in the sentence

Word	Appearance in Sentence					Total Appearance in Sentence
	S1	S2	S3	S4	S5	
<{ridwan}>	0	1	1	0	0	2
<{kamil}>	0	1	1	0	0	2
<{ketat}>	0	1	0	0	1	2
<{temu}>	0	0	1	0	1	2
<{masuk}>	0	2	0	0	1	3
<{wilayah}>	0	1	1	1	1	4

Word	Appearance in Sentence					Total Appearance in Sentence
	S1	S2	S3	S4	S5	
<{batas}>	0	1	1	1	1	4
<{jalan}>	0	2	0	0	0	2
<{ridwan, kamil}>	0	1	1	0	0	2
<{ketat, masuk}>	0	1	0	0	1	2
<{ketat, wilayah}>	0	1	0	0	1	2
<{masuk, wilayah}>	0	1	0	0	1	2
<{batas, wilayah}>	0	0	1	0	1	2
<{wilayah, batas}>	0	1	0	1	0	2
<{ketat, masuk, wilayah}>	0	1	0	0	1	2
Weight of sentence	0	15	7	3	10	

Table 6. Overlap between sentences with SPM

Sentence No.	S1	S2	S3	S4	S5
S1		0	0	0	0
S2	0		5	2	8
S3	0	5		2	4
S4	0	2	2		2
S5	0	8	4	2	

Table 7. Cost value between sentences with SPM

Sentence No.	Cost Value (using equation (6))
S1-S2	$\frac{(1-2)^2}{(0\times15)} = \infty$
S1-S3	$\frac{(1-3)^2}{(0\times7)} = \infty$
S1-S4	$\frac{(1-4)^2}{(0\times3)} = \infty$
S1-S5	$\frac{(1-5)^2}{(0\times10)} = \infty$
S2-S3	$\frac{(2-3)^2}{(5\times7)} = 0.028571429$
S2-S4	$\frac{(2-4)^2}{(2\times3)} = 0.666666667$
S2-S5	$\frac{(2-5)^2}{(8\times10)} = 0.1125$
S3-S4	$\frac{(3-4)^2}{(2\times3)} = 0.166666667$
S3-S5	$\frac{(3-5)^2}{(4\times10)} = 0.1$
S4-S5	$\frac{(4-5)^2}{(2\times10)} = 0.05$
S2-S1	$\frac{(2-1)^2}{(0\times0)} = \infty$
S3-S1	$\frac{(3-1)^2}{(0\times0)} = \infty$
S4-S1	$\frac{(4-1)^2}{(0\times0)} = \infty$
S5-S1	$\frac{(5-1)^2}{(0\times0)} = \infty$
S3-S2	$\frac{(3-2)^2}{(5\times15)} = 0.013333333$
S4-S2	$\frac{(4-2)^2}{(2\times15)} = 0.133333333$
S5-S2	$\frac{(5-2)^2}{(8\times15)} = 0.075$
S4-S3	$\frac{(4-3)^2}{(2\times7)} = 0.071428571$
S5-S3	$\frac{(5-3)^2}{(4\times7)} = 0.142857143$
S5-S4	$\frac{(5-4)^2}{(2\times10)} = 0.05$

Based on the cost value between sentence calculations in Table 6 and 7, the graph that can be formed is available in Figure 3. The process of combining the Bellman-Ford algorithm and SPM to produce the summary is illustrated in Table 8.

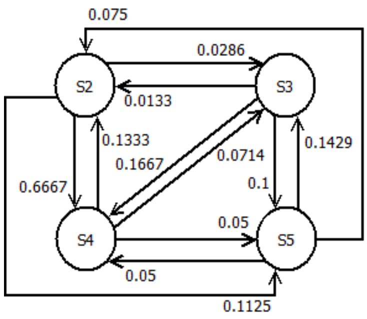
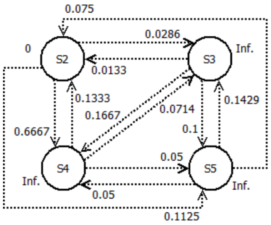
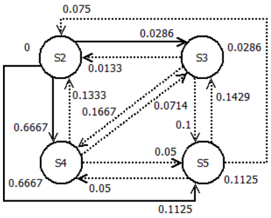
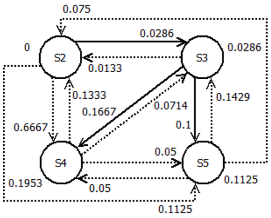
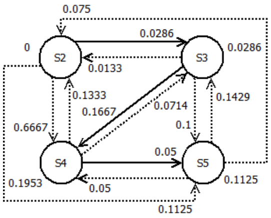
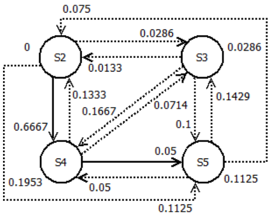
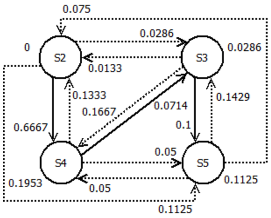
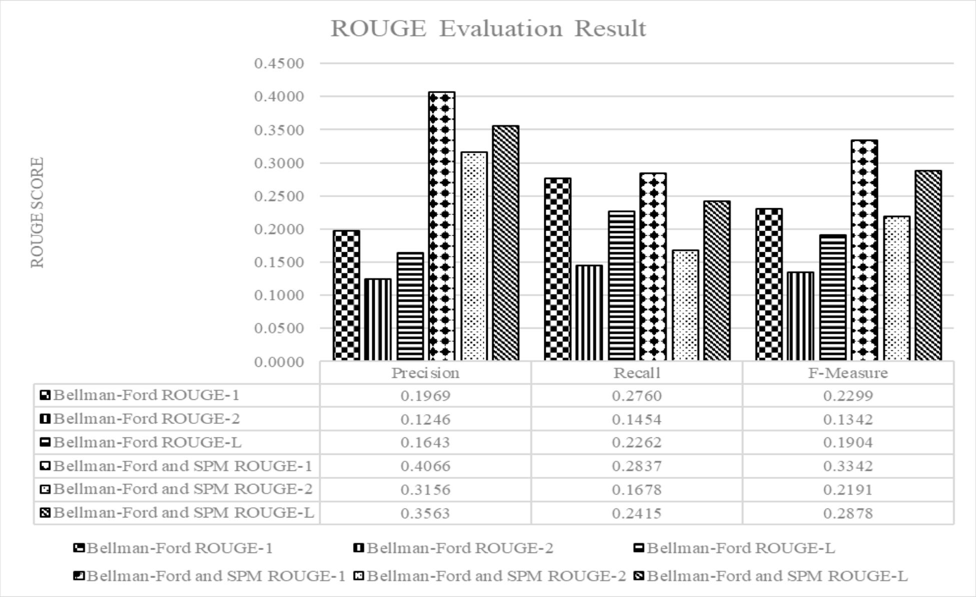


Figure 3. Illustration of sentence graph

Table 7. Bellman-Ford process

No	Bellman-Ford Process				Graph Illustration
	S2	S3	S4	S5	
1	0	∞	∞	∞	 0.075 0 S2 0.0286 0.0133 S3 Inf. 0.1333 0.1667 0.0714 0.1429 0.6667 0.1 0.05 S4 S5 Inf. 0.05 Inf. 0.1125
	S2	S3	S4	S5	
2	0	0.0286	0.6667	0.1125	 0.075 0 S2 0.0286 0.0133 S3 0.0286 0.1333 0.1667 0.0714 0.1429 0.6667 0.1 0.05 S4 S5 0.6667 0.05 0.1125 0.1125
	0	0.0286	0.1953	0.1125	 0.075 0 S2 0.0286 0.0133 S3 0.0286 0.1333 0.1667 0.0714 0.1429 0.6667 0.1 0.05 S4 S5 0.1953 0.05 0.1125 0.1125
	0	0.0286	0.1953	0.1125	 0.075 0 S2 0.0286 0.0133 S3 0.0286 0.1333 0.1667 0.0714 0.1429 0.6667 0.1 0.05 S4 S5 0.1953 0.05 0.1125 0.1125
	S2	S3	S4	S5	
3	0	0.0286	0.1953	0.1125	 0.075 0 S2 0.0286 0.0133 S3 0.0286 0.1333 0.1667 0.0714 0.1429 0.6667 0.1 0.05 S4 S5 0.1953 0.05 0.1125 0.1125
	0	0.0286	0.1953	0.1125	 0.075 0 S2 0.0286 0.0133 S3 0.0286 0.1333 0.1667 0.0714 0.1429 0.6667 0.1 0.05 S4 S5 0.1953 0.05 0.1125 0.1125



	Precision	Recall	F-Measure
Bellman-Ford ROUGE-1	0.1969	0.2760	0.2299
Bellman-Ford ROUGE-2	0.1246	0.1454	0.1342
Bellman-Ford ROUGE-L	0.1643	0.2262	0.1904
Bellman-Ford and SPM ROUGE-1	0.4066	0.2837	0.3342
Bellman-Ford and SPM ROUGE-2	0.3156	0.1678	0.2191
Bellman-Ford and SPM ROUGE-L	0.3563	0.2415	0.2878

Figure 4. ROUGE evaluation result

Based on the result of combination between Bellman-Ford and SPM above, the path that is passed from the starting point S2 to the end point S5 with the minimum number of weights is S3 and S5. Therefore, the summary results obtained are sentence 2 (S2), sentence 3 (S3) and sentence 5 (S5). The result is as follows: "*Gubernur Jawa Barat Ridwan Kamil meminta kepolisian untuk memperketat pintu masuk di wilayah perbatasan, termasuk jalan-jalan tikus. Menurut Ridwan Kamil, Pembatasan Sosial Berskala Besar (PSBB) di wilayah Bandung raya yang sudah diberlakukan sejak Rabu (22/4), masih ditemui sejumlah pelanggaran. "Mulai sekarang kita perketat penjagaan di perbatasan, tidak boleh ada warga yang masuk maupun keluar dari wilayahnya, kecuali dengan alasan yang jelas," kata pria yang karib disapa Emil itu saat menggelar pertemuan video conference dari Gedung Pakuan, Kota Bandung, Sabtu (25/4)*

d. Experimental Result

To evaluate the performance of the proposed methods which combines with SPM (Sentence Scoring and Bellman-Ford), the evaluation process measures the result of the summary using ROUGE-1, ROUGE-2, and ROUGE-L. ROUGE evaluation shows the performance of summary results with precision, recall, and f-measure value. Figure 4 presents the average of ROUGE-1, ROUGE-2, and ROUGE-L evaluation value of each method. The result show that as for Bellman-Ford, the combination of Bellman-Ford and SPM still produced better results.

4. Discussion

The proposed methods for this research integrate SPM with the Bellman-Ford algorithm as a graph-based approach representation. The experiments were conducted separately on each approach to observe the effect before and after integrating with SPM. The result of graph-based representation using the Bellman-Ford algorithm shows that SPM can increase the performance of the summary result. All ROUGE-1, ROUGE-2, and ROUGE-L scores have increased. F-measure of ROUGE-1 increase from 0.2299 to 0.3342 when Bellman-Ford is combined with SPM. The f-measure of ROUGE-2 also increases from 0.1342 to 0.2191, and the result of the f-measure of ROUGE-L increases from 0.1904 to 0.2878. Therefore, this shows that the graph-based approach can be improved with SPM.

The experiment shows the summary results using Bellman-Ford for the IndoSum dataset were not good. Further analysis shows that there are 4,158 or 22.15% summary results with an f-measure of ROUGE value is zero (either ROUGE-1, ROUGE-2, or ROUGE-L). But SPM can reduce the zero value of f-measure of ROUGE, from 22.15%, it reduces to 11.63% or 2184 summary result with zero value of f-measure. This result means many summary results with Bellman-Ford do not match the summary reference. The following document is the example of a summary result with zero value of f-measure of ROUGE (either ROUGE-1, ROUGE-2, or ROUGE-L):

This research also found that the summary results using Bellman-Ford for the IndoSum dataset were not good. Further analysis shows that there are 4,158 or 22.15% summary results with an f-measure of ROUGE value is zero (either ROUGE-1, ROUGE-2, or ROUGE-L). But SPM can reduce the zero value of f-measure of ROUGE, from 22.15%, it reduces to 11.63% or 2184 summary result with zero value of f-measure. This result means many summary results with Bellman-Ford do not match the summary reference. The following document is the example of a summary result with zero value of f-measure of ROUGE (either ROUGE-1, ROUGE-2, or ROUGE-L):

Example of News Article:

jakarta, cnn indonesia - - kurang dari 13 hari lagi bursa transfer bakal ditutup , liverpool dan crystal palace akhirnya mencapai kata sepakat . ***the reds resmi melepas striker mereka, christian benteke, ke crystal palace. pemain timnas belgia itu dijual ke crystal palace dengan harga £ 27 juta dan tambahan £ 5 juta jika palace berada di peringkat bagus musim ini.*** *benteke dibeli liverpool dari aston villa sebesar £ 32,5 juta pada 2015. musim lalu ia hanya mengemas 10 gol di semua kompetisi bersama Liverpool palace sebelumnya sempat mengajukan penawaran sebesar £ 23 juta plus tambahan, tapi ditolak liverpool.* ***bagi the eagles, ini akan menjadi rekor pembelian termahal klub tersebut sepanjang sejarah mereka di liga primer inggris****.skuat arahan alan pardew itu memang kekurangan pilihan para penyerang andalan setelah kepergian emmanuel adebayour, dwight gayle , marouane chamakh , dan yannick bolaise.* ***benteke juga kini sudah berada di london untuk melakukan tes medis sebelum menandatangani kontrak bersama palace****. kesepakatan dicapai liverpool setelah manajer tim, juergen klopp, mengatakan striker timnas inggris di skuatnya membutuhkan penampilan reguler. pada laga perdana menghadapi arsenal minggu (14 / 8), klopp sendiri tidak memasang satu pun striker dan menempatkan roberto firminho sebagai ' false nine ' atau penyerang semu. saat itu liverpool mempermalukan arsenal 4 - 3 di emirates stadium.* ***(bac)***

Reference Summary (from the bold text):

the reds resmi melepas striker mereka, christian benteke, ke crystal palace. pemain timnas belgia itu dijual ke crystal palace dengan harga £ 27 juta dan tambahan £ 5 juta jika palace berada di peringkat bagus musim ini. bagi the eagles, ini akan menjadi rekor pembelian termahal klub tersebut sepanjang sejarah mereka di liga primer inggris. benteke juga kini sudah berada di london untuk melakukan tes medis sebelum menandatangani kontrak bersama palace.

Summary Result of Bellman-Ford (from the bold and underline text):

(bac)

Summary Result of Bellman-Ford and SPM (from the underline text)**:**

pada laga perdana menghadapi arsenal minggu (14 / 8), klopp sendiri tidak memasang satu pun striker dan menempatkan roberto firminho sebagai ' false nine ' atau penyerang semu.

From the example above, Bellman-Ford produces a summary that does not exist in the summary reference. However, the combination of Bellman-Ford and SPM yields a better summary than without SPM. The combination of Bellman-Ford and SPM can produce summaries that cannot be generated with Bellman-Ford only.

4. Conclusion

This research aims to enhance the quality of Indonesian automatic text summary in using sequence of words as the proposed text representation. This research use SPM to produce sequence of words and combine it with Bellman-Ford algorithm. The results of a graph-based representation utilizing the Bellman-Ford algorithm suggest that SPM can improve summary result performance. Scores for ROUGE-1, ROUGE-2, and ROUGE-L have increased. When Bellman-Ford is paired with SPM, the f-measure of ROUGE-1 increases from 0.2299 to 0.3342, ROUGE-2 increases from 0.1342 to 0.2191, and ROUGE-L increased from 0.1904 to 0.2878. SPM can reduce ROUGE's zero value of f-measure when combined with the Bellman-Ford algorithm. This means that SPM can produce a summary when Bellman-Ford produces an unsimilar generated summary with the reference summary. Therefore, the combination of Bellman-Ford and SPM yields a summary that is still readable and meaningful. This research has also revealed a few issues that require further investigation. As a result, a few possible future studies are recommended to improve the current work, such as use another graph-based method (for example TextRank and LexRank) that can be combined with SPM. It is hoped that the contribution of this research has paved the way for future research in this field.

Acknowledgement

The author would like to thank for the support of the Center for Advanced Computing Technology, Universiti Teknikal Malaysia Melaka so that this article can be published. Also thank you for support from the academic community of UIN Sunan Gunung Djati Bandung, especially Prof. Dr. H. Muhammad Ali Ramdhani, S.TP., MT for his advice and knowledge.

References

[1] M. Gambhir and V. Gupta, "Recent automatic text summarization techniques: a survey," *Artif. Intell. Rev.*, vol. 47, pp. 1–66, 2017, doi: 10.1007/s10462-016-9475-9.

[2] V. Gupta and G. S. Lehal, "A Survey of Text Summarization Extractive techniques," in *Journal of Emerging Technologies in Web Intelligence*, 2010, vol. 2, no. 3, pp. 258–268, doi: 10.4304/jetwi.2.3.258-268.

[3] M. Rajangam and C. Annamalai, "Extractive document summarization using an adaptive, knowledge based cognitive model," *Cogn. Syst. Res.*, vol. 56, pp. 56–71, 2019, doi: 10.1016/j.cogsys.2018.11.005.

[4] N. R. Kasture, N. Yargal, N. N. Singh, N. Kulkarni, and V. Mathur, "A Survey on Methods of Abstractive Text Summarization," *Int. J. Res. Emerg. Sci. Technol.*, vol. 1, no. 6, 2014.

[5] G. S. Budhi, R. Intan, R. Silvia, and R. R. Stevanus, "Indonesian Automated Text Summarization," in *Proceeding ICSIIT*, 2007, pp. 26–27.

[6] D. Gunawan and R. F. Rahmat, "Evaluasi Algoritma Textrank Pada Peringkasan Teks Berbahasa Indonesia," Universitas Sumatera Utara, 2018.

[7] P. Wongchaisuwat, "Automatic Keyword Extraction Using TextRank," in *2019 IEEE 6th International Conference on Industrial Engineering and Applications, ICIEA 2019*, 2019, pp. 377–381, doi: 10.1109/IEA.2019.8714976.

[8] C. Mallick, A. K. Das, M. Dutta, A. K. Das, and A. Sarkar, "Graph-based text summarization using modified TextRank," in *Soft computing in data analytics*, Springer, 2019, pp. 137–146.

[9] G. Garmastewira and M. L. Khodra, "Summarizing Indonesian news articles using Graph Convolutional Network," *J. Inf. Commun. Technol.*, vol. 18, no. 3, pp. 345–365, 2019, doi: 10.32890/jict2019.18.3.6.

[10] D. Wang, P. Liu, Y. Zheng, X. Qiu, and X. Huang, "Heterogeneous graph neural networks for extractive document summarization," *arXiv Prepr. arXiv2004.12393*, 2020.

[11] S. Tuhpatussania, E. Utami, and A. D. Hartanto, "Comparison Of Lexrank Algorithm And Maximum Marginal Relevance In Summary Of Indonesian News Text In Online News Portals," *J. Pilar Nusa Mandiri*, vol. 18, no. 2, pp. 187–192, 2022.

[12] S. Agustian and S. Ramadhani, "Peringkasan teks otomatis (automated text summarization) pada artikel berbahasa indonesia menggunakan algoritma lexrank," *J. CoSciTech (Computer Sci. Inf. Technol.*, vol. 3, no. 3, pp. 371–381, 2022.

[13] W. W. Adytoma *et al.*, "Automatic Text Summarization for Hadith with Indonesian Text using Bellman-Ford Algorithm," in *2020 6th International Conference on Computing Engineering and Design (ICCED)*, 2020, pp. 1–6.

[14] M. F. Muharram, C. N. Alam, D. S. Maylawati, W. B. Zulfikar, N. Lukman, and M. A. Ramdhani, "Automatic Text Summarization for Multiple Scientific Indonesian Journal Article using Bellman-Ford Algorithm," in *The 3rd International Conference on Intelligent and Interactive Computing 2021*, 2021, vol. 2021.

[15] Y. J. Kumar, O. S. Goh, H. Basiron, N. H. Choon, and P. C. Suppiah, "A review on automatic text summarization approaches," *Journal of Computer Science*. 2016, doi: 10.3844/jcssp.2016.178.190.

[16] J. ge Yao, X. Wan, and J. Xiao, "Recent advances in document summarization," *Knowl. Inf. Syst.*, vol. 53, pp. 297–336, 2017, doi: 10.1007/s10115-017-1042-4.

[17] K. Nandhini and S. R. Balasundaram, "Improving readability through extractive summarization for learners with reading difficulties," *Egypt. Informatics J.*, vol. 14, no. 3, pp. 195–204, 2013, doi: 10.1016/j.eij.2013.09.001.

[18] D. Rahmawati, G. A. P. Saptawati, and Y. Widyani, "Document clustering using sequential pattern (SP): Maximal frequent sequences (MFS) as SP representation," in *2015 International Conference on Data and Software Engineering (ICoDSE 2015)*, 2015, pp. 98–102.

[19] G. A. P. Saptawati, "Set of frequent word sequence (SFWS) as document model for feature based document clustering," *Int. J. Electr. Eng. Informatics*, vol. 11, no. 4, pp. 822–832, 2019, doi: 10.15676/ijeei.2019.11.4.13.

[20] S. Alias, S. K. Mohammad, G. K. Hoon, and T. T. Ping, "A text representation model using Sequential Pattern-Growth method," *Pattern Anal. Appl.*, vol. 21, no. 1, pp. 233–247, 2018, doi: 10.1007/s10044-017-0624-9.

[21] K. Kurniawan and S. Louvan, "INDOSUM : A New Benchmark Dataset for Indonesian Text Summarization," *2018 Int. Conf. Asian Lang. Process.*, pp. 215–220, 2018.

[22] S. Vijayarani, J. Ilamathi, and M. Nithya, "Preprocessing Techniques for Text Mining - An Overview," *Int. J. Comput. Sci. Commun. Networks*, vol. 5, no. 1, pp. 7–16, 2015.

[23] H. A. Robbani, "Sastrawi," *MIT*, 2016. .

[24] M. Adriani, J. Asian, B. Nazief, S. M. M. Tahaghoghi, and H. E. Williams, "Stemming Indonesian : A confix-stripping approach," *ACM Trans. Asian Lang. Inf. Process.*, vol. 6, no. 4, pp. 1–33, 2007, doi: 10.1145/1316457.1316459.

[25] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Proceedings of the workshop on text summarization branches out (WAS 2004)*, 2004, pp. 74–81, doi: 10.1.1.111.9426.

[26] J. Han, J. Pei, Y. Yin, and R. Mao, "Mining frequent patterns without candidate generation: A frequent-pattern tree approach," *Data Min. Knowl. Discov.*, vol. 8, no. 1, pp. 53–87, 2004, doi:

10.1023/B:DAMI.0000005258.31418.83.

[27] M. Rofiq and R. F. Uzzy, "Penentuan Jalur Terpendek Menuju Cafe Di Kota Malang Menggunakan Metode Bellman-Ford Dengan Location Based Service Berbasis Android," *J. Ilm. Teknol. Inf. Asia*, vol. 8, no. 2, pp. 49–64, 2014.

[28] P. M. Hasugian, "Analisa dan implementasi algoritma bellman ford dalam menentukan jalur terpendek pengantaran barang dalam kota," *J. Mantik Penusa*, vol. 18, no. 2, pp. 118–123, 2015.

[29] R. Pramudita and N. Safitri, "Algoritma Bellman-Ford Untuk Menentukan Jalur Tercepat Dalam Sistem Informasi Geografis," *PIKSEL Penelit. Ilmu Komput. Sist. Embed. Log.*, vol. 6, no. 2, pp. 105–114, 2018, doi: 10.33558/piksel.v6i2.1502.