Available online at: http://proceeding.rsfpress.com/index.php/ess/index
LPPM UPN "Veteran" Yogyakarta Conference Series
Proceeding on Engineering and Science Series (ESS)
Volume 1 Number 1 (2020): 747-754

Description Processing Of Criminal Cases Using Latent Semantic Analysis Method

Hidayatulah Himawan, Dessyanto Boedi Prasetyo, Wilis Kaswidjanti

Universitas Pembangunan Nasional Veteran Yogyakarta
E-mail Address :if.iwan@upnyk.ac.id, E-mail Address : dess@upnyk.ac.id, E-mail Address : wilisk@upnyk.ac.id

Abstract

Identification of a criminal case requires a complete and thorough analysis. The description and sequence of events require attention that is not only seen from the side of the report, but the actual events accompanied by the evidence found. This process requires a lot of time, accuracy in analyzing requires a separate technique and method, for this reason, Latent Semantic Analysis (LSA) is used to summarize and draw conclusions about the relationship between one information and other related information. This description processing design uses similarities or semantics between words and sentences. Analyzing the relationships that are formed so that it is hoped that this design can be implemented into a system that helps the process of investigating a criminal case.

Keywords: Design, Latent Semantic Analysis, Semantic

I. INTRODUCTION

The use of information technology, especially for the efficiency and effectiveness of work within an institution or institution, is a must in itself. Including in law enforcement institutions which are expected to always provide the best, fast, and precise service and there is no mistake in providing an assessment and analysis of a criminal case or crime. In article 2 of Law No.2 of 2002 concerning the Indonesian National Police, one of the functions of the police is protection, protection, and community service. In this case, police services are not only in the form of traffic activities, but other types of services include serving the receipt of reports from the public. These reports can be in the form of incident reports, reports of loss, reports of crimes and violence, as well as reports requiring help and security services. Received reports will be recorded for data and further action. The report contains various information, one of which is a description of the events reported.

In a study conducted by Chen, Chang, & Chen (Chen, Chang, & Chen, 2013), they developed a summary to summarize the sound of broadcast news. In this study, they used speech recognition built by themselves. The accuracy of the speech recognition results is still not accurate enough with an error rate of 35%. And in that study also compared the KLM and LSA methods. From the test results, it was found that KLM was better than LSA. However, the LSA used in this study is the LSA developed first by Gong & Liu (Gong & Liu, 2001), in automatic text summarization, the accuracy

DOI: https://doi.org/10.31098/ess.v1i1.172

produced by Gong & Liu (Gong & Liu, 2001) can still be improved by adding a cross method developed by Winata & Rainarli (Winata & Rainarli, 2016). Likewise, research conducted by Ribeiro & Matos (Ribeiro R., 2007), comparing the summary of the LSA method Gong & Liu (Gong & Liu, 2001) with the Maximal Marginal Relevance (MMR) with test results obtained better MMR results by 10%. In addition, from the results of this study, it is said that increasing speech recognition and sentence segmentation will increase the accuracy of the summarization.

The conclusion is a text that is generated from document files or more which states important information from the original file which has a relatively shorter size (Liu, Chen, Chen, Wang, & Yen, 2015) (Bharti, Babu, & Jena, 2017). One method that can be used to summarize documents is Natural Language Processing (NLP), which uses the automatic summarization method (Rott & Cerva, 2016) (Tixier, Hallowell, Rajagopalan, & Bowman, 2020). The summarization is based on the topic obtained from the analysis based on the similarity of meaning (semantic) between words and sentences. The method used is Latent Semantic Analysis (LSA) which is implemented using Singular Value Decomposition (SVD). To improve the accuracy of the LSA by using a cross method for the extraction of sentences after SVD. The result of adding sentence extraction using the cross method itself shows better results than the previous LSA method (Savanti, Gotami, & Dewi, 2018). In research conducted by Akbar, Husodo, & Zubaidi (Akbar et al., 2019), it is said that the accuracy results of the Google Speech API are used for correcting the memorization of the Qur'an produce very accurate accuracy. So that using the Google Speech API can improve the accuracy of voice summarization with LSA.

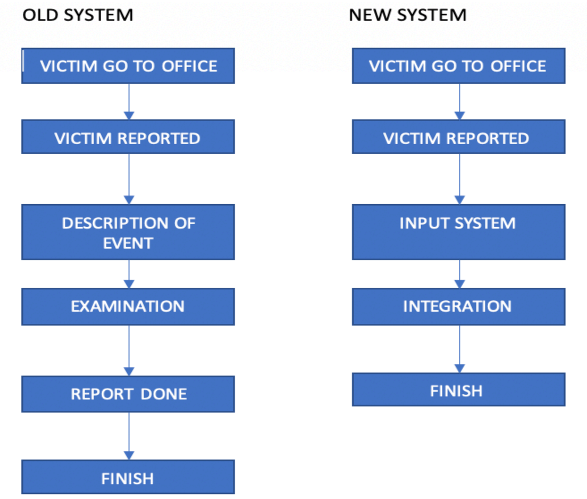


Figure 1.Different Design Workflow Systems.

II. LITERATURE REVIEW

This study uses two ways to analyze existing problems, namely:

1. Literature study, where at this stage it is used to obtain and deepen theory or knowledge as well as references related to the research theme. Sources of literature study used by the author are online scientific journals, e-books, and books.

2. Problem Analysis. This stage is to identify the problem and the content or content that will be used in the research.

The process of making a summary can be done in 2 ways (Mani et al., 2012)(Nada, 2004), namely:

a. Extraction (extractive summary). In the extraction approach, shallower approaches are used by the system of copying text units that are considered the most important or most informative from the source text into a summary. The text units to be copied can be the main clause, the main sentence, or the main paragraph.

b. Abstraction (abstractive summary). In the abstraction approach, deeper approaches or deep approaches are used by involving paraphrases from the source text. The abstraction technique using the essence of the source text then makes a summary by creating new sentences that represent the essence of the source text in different data from the sentences in the source text. In general, abstraction can summarize text more strongly than extraction, but the system is more difficult to develop because it applies natural language generation technology, which is a separately developed discussion.

III. RESEARCH METHODOLOGY

The process begins with recording the voice of the complaint as data input. After the data is obtained, the speech segmentation process will be carried out. This aims to break down the results of the voice data obtained into several voice data which are useful for making sentences. After that, a speech to text or speech recognition process is carried out to convert voice data into text data. Then the final step is the summarization process.

After the sound recording is done, the next step is to segment the sound (speech segmentation) of the recorded results into several sound pieces. The first process is to load a file from the recorded sound, then check the audio chunks with a minimum silence duration of 400ms and a maximum silence threshold of -16. If these conditions are true, the audio is trimmed (trim) to be an audio chunk. Furthermore, silence padding is added before and after the chunk and normalizes the chunk. And the last process is to export the audio pieces so that you will get the audio pieces that will be useful in the voice to the text conversion process and also the summarization process.

III.1. Preprocessing System

Preprocessing is the initial stage of text processing. Preprocessing has the aim of producing index terms from text documents that are carried out so that they can be processed to the next stage, namely the TF-IDF processing stage and the LSA method which uses SVD to be clean from the noise that can interfere with the next process stage. The stages in preprocessing are carried out with the stages of parsing, tokenization, filtering, and stemming (Savanti et al., 2018).

In the preprocessing system, there is a sentence solving process which is a stage where the strings or structures are separated in each sentence by dividing the dot symbol into separate components. Each document that has been split will be included in the sentence list. The output of the segmentation results is a collection of sentences that will be used in the next process (Savanti et al., 2018).

Table 1. Examples of Sentence Truncation

No.	Document	Tokenizing Process
1	Saya Sedang Berjalan Santai. Budi dan Tono Bermain Bola	{[Saya Sedang Berjalan Santai],[Budi dan Tono Bermain Bola.][Ibu dan Bapak Berjalan Santai, Kakak Bermain Bola]}

In addition, there is also a tokenizing process or word splitting, which is the way of cutting the input string into words, phrases, symbols, or other meaningful elements called tokens. A textual data initially contains only a set of words. In an information search process requires a collection of words from a dataset. Therefore, a condition is needed to split the document in the form of tokenization so that the text can be stored in a format that can be read by machines. (Gurusamy & Kannan, 2014).

Table 2. Examples of Word Breakers

No.	Document	Tokenizing Process
1	Saya Sedang Berjalan Santai	[Saya,Sedang,Berjalan,Santai]
2	Budi dan Tono Bermain Bola	[Budi,dan,Tono,Bermain,Bola]
3	Ibu dan Bapak Berjalan Santai, Kakak Bermain Bola	[Ibu,dan,Bapak,Berjalan,Santai,Kakak,Bermain,Bola]

III.2. Stemming Process

The process of stemming is the process of taking a word base for each word from a document. This process is carried out by decomposing all word forms, be it a prefix, suffix, or a combination of prefix and suffix (confix) into a root word (stem) (Kannan & Gurusamy, 2014). The stemmer algorithm introduced by Nazief and Adriani is defined as follows:

1. At the beginning of the stemming process and each subsequent stage, check the outcome of the stemming process of words that are input in that step into the basic word dictionary. If a word is found, it means that the word is already in the form of a root word and the stemming process is stopped. If not found, then the next stage is carried out.

2. Eliminate Inflection Suffixes ("-lah", "-kah", "-ku", "-mu", or "-nya"). If it is particles ("-lah", "-kah", "-tah " Or " -pun ") then this step is repeated again to delete the Possesive Pronouns (" -ku "," -mu ", or "-nya"), if any.

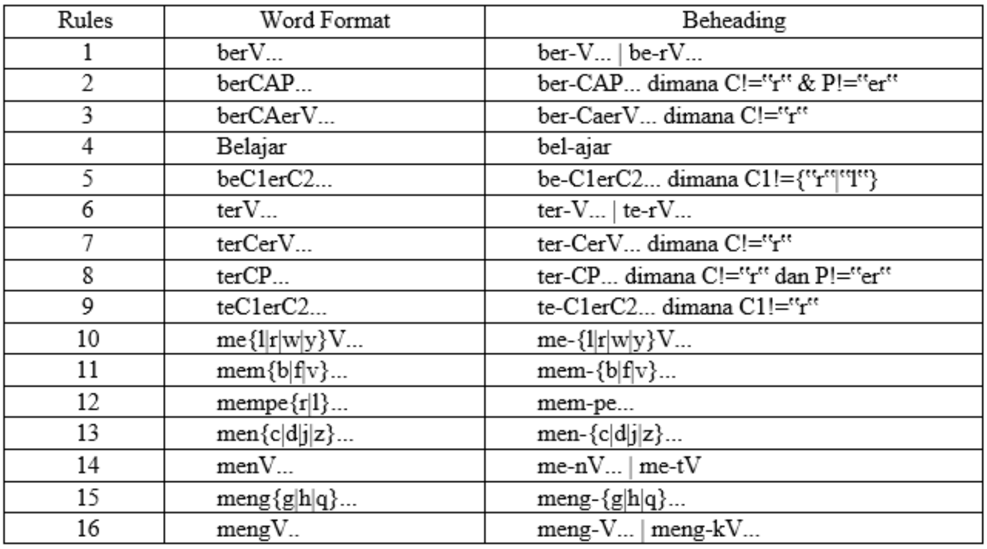
3. Delete Derivation Suffixes (“-i”, “-an” or “-kan”). If a word is found in the dictionary, the algorithm stops. If not then go to step 3a
   a. If "-an" has been deleted and the last letter of the word is "-k", then "-k" is also deleted. If the word is found in the dictionary, the algorithm stops. If not found then do step 3b.
   b. Deleted suffixes (“-i”, “-an” or “-kan”) are returned, go to step 4

4. Eliminate derivation prefixes.
   a. Step 4 stops if:
      i. A forbidden prefix and suffix combination occurs.
      ii. The prefixes currently detected are the same as previously omitted prefixes.
      iii. Three prefixes have been removed.
   b. Identify the prefix type and omit it. Prefixes are of two types:
      i. Standard (“di-”, “ke-”, “se-”) which can be omitted directly from words.
      ii. Complex ("men", "be-", "pe", "te-") are types of prefixes that can morphologically according to the root that follows them.
   Look for the prefixed word in the root dictionary. If not found, then step 4 is repeated. If found, the whole process is stopped.

5. If after step 4 the root word is still missing, then the recoding process is carried out by adding a recoding character at the beginning of the word that was cut off. For example, the word "catch" (rule 15), after being cut, becomes "catch". Because it is not valid, then the recording is done and results in the word "catch".

6. If all steps fail, then the word input tested on this algorithm is considered to be the root word.

Table 3. Examples of Word Hypothesis Rules



Rules	Word Format	Beheading
1	berV...	ber-V... | be-rV...
2	berCAP...	ber-CAP... dimana C!=“r“ & P!=“er“
3	berCAerV...	ber-CaerV... dimana C!=“r“
4	Belajar	bel-ajar
5	beC1erC2...	be-C1erC2... dimana C1!={“r“|“l“}
6	terV...	ter-V... | te-rV...
7	terCerV...	ter-CerV... dimana C!=“r“
8	terCP...	ter-CP... dimana C!=“r“ dan P!=“er“
9	teC1erC2...	te-C1erC2... dimana C1!=“r“
10	me{l|r|w|y}V...	me-{l|r|w|y}V...
11	mem{b|f|v}...	mem-{b|f|v}...
12	mempe{r|l}...	mem-pe...
13	men{c|d|j|z}...	men-{c|d|j|z}...
14	menV...	me-nV... | me-tV
15	meng{g|h|q}...	meng-{g|h|q}...
16	mengV..	meng-V... | meng-kV...

III.3. Weighting Term-Frequency – Inverse Document Frequency

The TF-IDF calculation phase is carried out in order to obtain the weight of each word in each document. The weighting of each word is carried out using the term index from preprocessing using the TF-IDF (Term-Frequency - Inverse Document Frequency) method and the weight of a word will be generated based on the number of times the word appears in the document and the appearance of words in other documents. The weighting of words on each word is obtained from calculating the frequency of appearance of terms in the document called TF (Term Frequency), then calculating the frequency of appearance of documents containing a term called DF (Document Frequency) as well as the calculation of IDF (Inverse Document Frequency) which calculates the number of documents containing terms searched from several existing documents (Savanti et al., 2018).

Table 4. Example of TF-IDF Calculation Results

Q	TF			Df	D/df	IDF	IDF+1	W=tf*(IDF+1)		
	D1	D2	D3					D1	D2	D3
Saya	1	0	0	1	3	0.477	1.477	1.477	0	0
Jalan	1	0	1	2	1.5	0.176	1.176	1.176	0	1.176
Santai	1	0	1	2	1.5	0.176	1.176	1.176	0	1.176
Budi	0	1	0	1	3	0.477	1.477	0	1.477	0
tono	0	1	0	2	1.5	0.176	1.176	0	1.176	0
main	0	1	1	2	1.5	0.176	1.176	0	1.176	1.176
bola	0	1	1	2	1.5	0.176	1.176	0	1.176	1.176
ibu	0	0	1	1	3	0.477	1.477	0	0	1.477
bapak	0	0	1	1	3	0.477	1.477	0	0	1.477
kakak	0	0	1	1	3	0.477	1.477	0	0	1.477
the value of each document weight								Sum(d1)	Sum(d2)	Sum(d3)
								3.829	5.005	9.135

IV. FINDING AND DISCUSSION

The design is made using a flowchart which describes the processes being carried out. The following is a flowchart design for the entire summarization process, starting from recording the sound until getting the summary results.

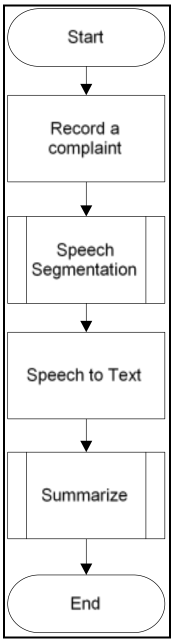


Figure 2. Overall Process Flowchart

The summary process begins with the text from the speech to text, summarized by the preprocessing process. After that, the TF-IDF calculations and reduction calculations were carried out using the LSA and SVD methods. From the results of the LSA, text selection will be carried out using a cross method process.

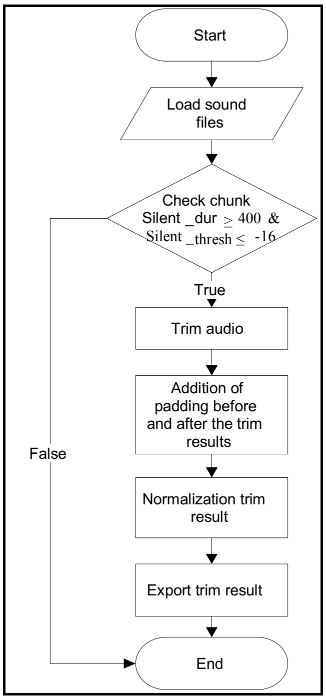

Start
Load sound files
Check chunk Silent _dur ≥ 400 & Silent _thresh ≤ -16
True
False
Trim audio
Addition of padding before and after the trim results
Normalization trim result
Export trim result
End

Figure 3. Speech Segmentation Flowchart

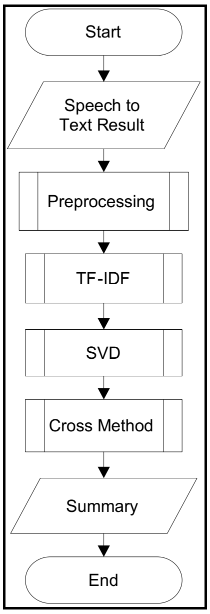


Figure 4. Summary Flowchart

V. CONCLUSION

Based on design research on criminal case description processing, it can be concluded that:

1. The design can be developed into a systematic application to help the results of the work become more effective and efficient.
2. There is a need for human resources that can support the creation of an effective system so that the optimization of the use of the description of complaints in criminal cases can be more accountable according to established procedures.

VI. REFERENCES

Akbar, A., Husodo, A. Y., Zubaidi, A., Studi, P., Informatika, T., Teknik, F., & Mataram, U. (2019). IMPLEMENTASI GOOGLE SPEECH API PADA APLIKASI KOREKSI HAFALAN AL-QUR ' AN BERBASIS Android (The Implementation of the Google Speech on Qur ' an Recitation Correction, *1*(1), 1–8.

Bharti, S. K., Babu, K. S., & Jena, S. K. (2017). Automatic Keyword Extraction for Text Summarization : A Survey.

Chen, B., Chang, H., & Chen, K. (2013). Sentence modeling for extractive speech summarization. *IEEE International Conference on Multimedia and Expo (ICME)*.

Gong, Y., & Liu, X. (2001). Generic Text Summarization Using Relevance Measure and Latent Semantic Analysis. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 19–25). New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/383952.383955

Kannan, S., & Gurusamy, V. (2014). Preprocessing Techniques for Text Mining.

Liu, S., Chen, K., Chen, B., Wang, H., & Yen, H. (2015). Positional Language Modeling for Extractive Broadcast News Speech Summarization Institute of Information Science, Academia

Sinica, Taiwan National Taiwan Normal University, Taiwan, (Lm), 2729–2733.

Mani, I., Klein, G., House, D., Hirschman, L., Firmin, T., Sundheim, B., … Sundheim, B. (2012). SUMMARY : a text summarization evaluation SUMAC : a text summarization evaluation, (2002). https://doi.org/10.1017/S1351324901002741

Nada, K. T. H. (2004). Summaries and the Process of Summarization, (1997), 1–15.

Ribeiro R., de M. D. M. (2007). Extractive Summarization of Broadcast News: Comparing Strategies Title, *4629*, 74628.

Rott, M., & Cerva, P. (2016). Speech-to-Text Summarization Using Automatic Phrase Extraction from Recognized Text, 101–108. https://doi.org/10.1007/978-3-319-45510-5

Savanti, N., Gotami, W., & Dewi, R. K. (2018). Peringkasan Teks Otomatis Secara Ekstraktif Pada Artikel Berita Kesehatan Berbahasa Indonesia Dengan Menggunakan Metode Latent Semantic Analysis, *2*(9), 2821–2828.

Tixier, A. J., Hallowell, M. R., Rajagopalan, B., & Bowman, D. (2020). Automation in Construction Automated content analysis for construction safety : A natural language processing system to extract precursors and outcomes from unstructured injury reports. *Automation in Construction*, *62*(2016), 45–56. https://doi.org/10.1016/j.autcon.2015.11.001

Winata, F., & Rainarli, E. (2016). IMPLEMENTASI CROSS METHOD LATENT SEMANTIC ANALYSIS UNTUK MERINGKAS DOKUMEN BERITA, *15*(4), 266–277.