The implementation of web service based text preprocessing to measure Indonesian student thesis similarity level

Yan Watequlis Syaifudin*, Pramana Yoga Saputra, and Dwi Puspitasari

State Polytechnic of Malang, Information Technology Department, Indonesia

Abstract. The plagiarism of scientific work, especially undergraduate thesis, mostly happened in the college. In this research we used text mining, a new method which can be used to do the checking procedure, to obtain specific pattern of the document. After obtaining the document pattern, we compare the pattern with another document pattern. If the level of pattern similarity is high, it can be suspected as plagiarism. This paper will explain the development of the text preprocessing, a part of text mining. We choosed Nazief and Adriani Algorithm as a text preprocessing algorithm for this research. This research will result a text preprocessing web service. The web service is expected to be used for further development of text mining.

1 Introduction

Text is a language expression, in which one form is a work of writing. Text can be composed of one or more words, such as phrases, clauses and sentences. A text can contain information the author wants to convey to the reader. In information technology, the text is a combination of both characters and words that can be read by humans. They also can be encoded into a format that can be read by a computer, such as ASCII.

Currently the most widely used information exchange is through text exchanges. The text belongs to the category of semi structured and unstructured formats. The example of semi-structured formats is HTML document and the examples of unstructured format are email and full-text document. The structured format is data that stored in the database and obtaining information from the database is relatively easier than retrieving information from unstructured format documents. To retrieve information from the unstructured document, we can use text mining.

In this research, the software will be developed from earlier research on implementation of Nazief & Adriani algorithm with preprocessing of similarity level measurement of thesis title. And in this research we will develop a system with web service for stemming process based on previous research. Web service itself is a service provided by the web site and this service can be used by other systems to perform a data processing. With the web service, there is interoperability between systems. Interoperability is the ability to interact between one system with another, where the interacting systems have different characteristics. We have chosen the development of the stemming process into the form of web service, because we want to be able to develop text mining application based on web sustainably, and the

software can be developed by anyone by utilizing the web service.

2 Basic Concepts

2.1 Text Mining

Text mining is defined as a process for finding hidden, useful, and interesting patterns from unstructured documents [1]. To find the pattern in the document takes many stages because the text in the document is unstructured and complex. Some of the steps done in text mining are as follows:

1. Text preprocessing
2. Knowledge distillation

The preprocessing stage is the data preparation stage before the knowledge distillation stage. Once obtained, a collection of messages will be conducted text preprocessing stage [2]. The data in question is the text to be retrieved information. The preprocessing stage consists of:

2.1.1 Tokenizing

The cutting phase of input string is based on each word that compiles it. For these deductions, usually commas, periods and spaces are used as markers in separating words.

2.1.2 Filtering

Filtering is a phase of taking important words that have been resulted from the tokenization process. At this stage, the less important words are discarded (stopword removal) and important words are stored.

* Corresponding author: gulis@polinema.ac.id

2.1.3 Stemming

Stemming is the process of finding the basic word of a word in the text [3]. The stemming is tightly related to basic word or lemma and the sub lemmas. The lemma and sub lemma of Indonesian Language have been grown and absorb from foreign languages or Indonesian traditional languages [4]. The basic word search process of each word that have been generated from the filtering stage.

2.2 Nazief and Adriani Algorithm

In Indonesian, there have been some algorithms, such as Nazief & Andriani, Porter for Indonesian language, Vega, and Arifin & Setiono stemming algorithms. Nazief & Andriani algorithm has the highest accuracy [5]. The stemming algorithm Nazief and Adriani was developed based on the Indonesian morphology rules that grouped the affixes into prefixes, infixes, suffixes, and confixes. There are variations of affixes including prefixes, suffixes, infixes, and confixes [6]. The stemming process of Indonesian texts is become more complex because there are variations of affixes should be removed to get the root word of a word. The words in Bahasa Indonesia have the special and complex morphological structure compared with the words in other languages [7]. Generally, the basic word of Indonesian consists of a combination: Prefix 1 + Prefix 2 + Basic word + Suffix 3 + Suffix 2 + Suffix 1.

The Nazief & Adriani algorithm has the following stages:

1. Find the stemmed word in the basic dictionary. If found, then assume the word is a root word. So the algorithm stops.
2. Inflection Suffixes ("-lah", "-kah", "-ku", "-mu", or "nya") are discarded. If it was a particle ("-lah", "-kah", "-tah" or "-pun") then this step is repeated again to remove Possessive Pronouns ("-ku", "-mu" or "nya"), if there are.
3. Remove Derivation Suffixes ("-i", "-an" or "-kan"). If the word is found in the dictionary, then the algorithm stops. If not found, 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, then the algorithm stops. If not found, then do step 3.
 - b. The deleted suffix ("-i", "-an" or "-kan") is returned, go to step 4.
4. First find the stemmed word in the basic word dictionary. If found, then it is assumed that word is root word. So the algorithm stops.
5. Inflection Suffixes ("-lah", "-kah", "-ku", "-mu", or "nya") are discarded. If it is a particle ("-lah", "-kah", "-tah" or "-pun") then this step is repeated again to remove Possessive Pronouns ("mu", "ku" or "nya"), if there are.
6. Remove Derivation Suffixes ("-i", "-an" or "-kan"). If the word is found in the dictionary, then the algorithm stops. If not found, 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, then the algorithm stops. If not found, then do step 3.

- b. The deleted suffix ("-i", "-an" or "-kan") is returned, go to step 4.
7. Remove the Derivation Prefix. If in step 3 there is a suffix that have been deleted, then go to step 4a, else go to step 4b.
 - a. Check the end-of-end combination table that was not allowed. If found, then the algorithm stops, else go to step 4b.
 - b. For $i = 1$ to 3, specify the prefix type and then delete the prefix. If root word has not been found do step 5, if it is then the algorithm stops. Note: if the second prefix is the same as the first prefix the algorithm stops.
8. Doing Recoding.
9. If all the steps have been completed but not succeeded, then the initial word is assumed as root word. The process is complete.

2.3 Windows Communication Foundation (WCF)

Windows Communication Foundation (WCF) is a framework used in building service-oriented based applications. Using WCF can send data in the form of asynchronous messages from one endpoint to another, that shown in Figure 1 [8]. A service endpoint can be part of an ongoing service hosted by IIS, or it can be part of a hosted service within the application. An endpoint could be a client of a service that requests data from the service endpoint. Message either could be as simple as a single character or word, sent in XML or JSON or could be a complex message as a binary data stream.

Fig. 1. WCF Model.

In short, WCF is designed to offer a manageable approach in creating web services and web service clients.

3 Application Design

In this research, we designed an architecture of making web-based stemming service shown as in Figure 2. In the picture explained that the client has the text that we want to stem, then the text is sent by the client to the server via web service, in this study the service named Stemming.svc. The text will be stemmed by the server

and the results of the stemming will be sent by the server to the client through the web service in JSON format. The JSON documents obtained by the client can be followed up for example presented in tabular form, or can be used as resource in further application development, for example used for comparison of similarity level of a document to obtain similarity value.

Fig. 2. Application Architecture.

In Figure 2 there is a web service that uses Windows Communication Foundation technology, created using ASP.NET. As for the client, there are clients A and client B, which is a client that has a platform other than ASP.NET. The workings of the created application can be seen in the flowchart in Figure 3, and the explanation is as follows:

Fig. 3. Application workflow.

Explanation of workflow in Fig. 1 is as follows:

1. The client enters the text that you want to be stacked through URL: `http://localhost/STService/Stemming.svc/json/" stemmed text"`
2. The text will be stacked by the server
3. The result of stemming is dataset
4. The dataset is then converted into JSON format
5. JSON data is sent to the client
6. JSON data can be directly displayed to the client, or can be presented into a table, or can be used as a resource to check the level of similarity of a document.

4 Result and Discussion

This research, a web-based stemming service, produces a web service running on the ASP.NET platform. The web service utilizes Windows Communication Foundation (WCF) technology, which applies the principle of REST.

This research uses Microsoft Visual Studio 2010 for code generation program. First made a WCF project with the name STService. In the project there are four main files in making web service, namely:

- Global.asax
Used to set the application connection to the database, which in the database other than used to store data, is also used to process data using stored procedure
- IStemming.cs
Contains the exposed method and will be used by the client. The HTTP function used, as well as the message format (XML / JSON) are also defined in this section
- Stemming.svc
Contains detailed logic of the exposed method in IStemming.cs
- Web.config
Contains service settings, this part is very important, because even if the algorithm used is correct, but if there is an error in the settings, then the service will not run.

4.1 Application Result

To access this service, we can use a web browser such as Mozilla, Google Chrome, or Internet Explorer. Make sure that IIS on the server computer is activated and the service is running. Next, enter the following link in the browser:

`http://localhost/STService/Stemming.svc`

Next, in the web browser will appear as shown in Figure 4 below.

Fig. 4. Display of web service in the browser.

Next, we can enter the sentence to be processed. For example, the sentence to be stemmed is the following sentence:

“ketika masih anak-anak, dia suka bermain dengan anak yang sebaya dengannya”

Then input the following link to the web browser:

[http://localhost:8080/STService/Stemming.svc/json/ketika masih anak-anak, dia suka bermain dengan anak yang sebaya dengannya](http://localhost:8080/STService/Stemming.svc/json/ketika%20masih%20anak-anak,%20dia%20suka%20bermain%20dengan%20anak%20yang%20sebaya%20dengannya)

Next will appear JSON data as in picture 5 below.

Fig. 5. The JSON data result.

Furthermore, if checked on JSON reader online, it will be shown as Figure 6.

Fig. 6. JSON data on the JSON online reader.

The result is JSON data that will be used as resource to make further process by client.

4.2 Discussion

Software testing is performed to determine the time of measurement of similarity and will then be tested based on the ratio of the number of documents and the length of the process. Test results on 10 documents require a processing time of 12 seconds. Testing time will continue to increase along with the number of documents to be tested in plagiarism.

In fact, there are a lot of cases of plagiarism occurred in Indonesia, which is not only done by students but also by lecturers [9]. So, it becomes very important for every university to be able to have a repository of scientific works, both from lecturers and students. Furthermore, from the repository we can check the similarity of scientific work using software from the results of this research. The web service system will greatly assist each university in providing plagiarism checking services that can be accessed by outside systems based on existing repositories. The next step is to build a comprehensive plagiarism checking system and integrate with other systems, and this will be easier with the web service. To effectively deter plagiarism, a comprehensive and effective policy should be in place, which includes definition, detection, and penalties [10].

5 Conclusion

The result of stemming can be used well by the client application that has been made before to measure the level of similarity. Datasets in JSON format can also be converted in database format, so it can be used for applications that use database to store student thesis data.

Besides can be used for similarity measurement, web service can also be used to develop applications for learning Indonesian language, checking spelling, and so forth.

References

1. K.L. Sumathy, M. Chidambaram. "Text Mining: Concepts, Applications, Tools and Issues – An Overview". *International Journal of Computer Applications*, Vol. **80**, No. 4. (October 2013)
2. Y.W. Syaifudin, D. Puspitasari. "Twitter Data Mining for Sentiment Analysis on Peoples Feedback Against Government Public Policy". *MATTER: International Journal of Science and Technology*, pp. 100-122. (2017)
3. P.M. Prihatini, I.K.G.D. Putra, I.A.D. Giriantari, M. Sudarma. "Stemming Algorithm for Indonesian Digital News Text Processing". *International Journal of Engineering and Emerging Technology*. p. 1-7, Mar. (2018)
4. R. Setiawan, A. Kurniawan, W. Budiharto, I.H. Kartowisastro, H. Prabowo. "Flexible affix classification for stemming Indonesian Language". *13th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*. (2016)
5. R.B.S. Putra, E. Utami. "Non-formal affixed word stemming in Indonesian language". 2018 International Conference on Information and Communications Technology (ICOIACT). (2018)
6. H. Widayanto, A.F. Huda. "Comparison Nazief Adriani and CS Stemmer Algorithm for Stemm Real Data". *e-Proceeding of Engineering*: Vol. **4**, No.3 Desember 2017, Page 5215. (2017)
7. A.F. Hidayatullah, C.I. Ratnasari, S. Wisnugroho. "Analysis of Stemming Influence on Indonesian Tweet Classification". *TELKOMNIKA*, pp. 665~673, Vol.**14**, No.2, (June 2016)
8. Microsoft. What Is Windows Communication Foundation, ([https://msdn.microsoft.com/en-us/id/library/ms731082\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/id/library/ms731082(v=vs.110).aspx), Accessed 27 Oktober 2017)
9. R. Agustina, P. Raharjo. "Exploring Plagiarism into Perspectives of Indonesian Academics and Students". *Journal of Education and Learning (EduLearn)*. Vol. **11** (3) pp. 262-272. (2017)
10. T. Adiningrum. "Reviewing Plagiarism: An Input for Indonesian Higher Education". *Journal of Academic Ethics* **13**(1):107-120. (2015)