IMPLEMENTATION SVM METHOD FOR QUESTION AND ANSWERING IN A FRONT OFFICE CASES

Tommy Edityantama Hutapea[1], Kania Evita Dewi[2]
[1,2] Universitas Komputer Indonesia
Jl.Dipatiukur No. 112-116, Coblong, Bandung, Jawa Barat, 40132
E-mail : t.edityantama@email.unikom.ac.id[1], kania.evita.dewi@email.unikom.ac.id[2]

ABSTRACT

The Question Answering System is a computer system that can answer questions using natural language commonly used by humans automatically. Previous research on question and answer system has been done before one of them using rule based using indonesian language, but in the research has not yielded good accuracy that is equal to 60%. The other research on question and answer system using machine learning approach method is SVM method, but in the research it produces 80% accuracy using English, Arabic, and English-Arabic. So in this research, will apply approach method machine learning SVM for question and answer system in front office with Indonesian language. Stages performed system to get the answer is divided into three stages, namely Preprocessing, Classification and Prediction answers. Classification is the initial stage in which the training data will be classified according to the type of question. then look for answers that are in the database based on the proximity of the question with the sentence on the database. The result of this main process is the pre-filled answer on each training data. The result shows the accuracy obtained at machine learning approach svm for question and answer system in case of front office get accuracy value equal to 39%. The accuracy value is due to lack of training data.

keyword : Question Answering System, *Support Vector Machine*, Bahasa Indonesia, SVM

1. INTRODUCTION

Question Answering System is a system on a computer that can answer questions using natural language human commonly used automatically. The answers obtained from the system contains information that came from a source database [1]. A question answering, expected to help people to get more information about certain things without having to ask someone else [2].

In previous research on question and answer system in Indonesian language has been done, one of them by using rule base [2] [3]. In the research has not got a good accuracy that 60% and only detect keywords contained in the rule, so if the system detect keywords that are not in the rule will got incorrect answers.

To fix keywords that are not in the rule can use Machine Learning approach [4]. On the research, explaining that using the Machine Learning approach can produce a system that includes thousands of features of questions and classify the question automatically and improve the productivity of the question and answer system.

The other research on question and answer system using 5 methods of Learning Machine approach that is K-Nearest Neighbor, Naive Baiyes, Support Vector Machine (SVM), Sparse Network of Winnows (SNoW) dan Decision Tree (DT) [5]. On that research SVM method got an 87.4% accuracy and used English.

The other research on question and answer system using SVM method. The results obtained are the above accuracy value 80-90% [6] [7] [8] [9]. On that research using Arab and English.

Therefore, in this study intends to develop a question and answer system in the Front Office UNIKOM by applying the approach machine learning method Support Vector Machine (SVM) in the Indonesian language. On This research is expected to produce and accurate good enough.

1.1. Sistem Tanya Jawab

Question Answering System is a system on a computer that can answer questions using natural language human commonly used automatically. The answers obtained from the system contains information that came from a source database [1]. A question answering, expected to help people to get more information about certain things without having to ask someone else [2].

The main purpose of an automated answer system is to find the right answer using natural language determined by humans. Currently the question and answer system has been developed in various fields such as personal assistant, customer service applied to the service in order to obtain information and meet the needs of consumers.

1.2. Preprocessing

Preprocessing is the process raw data before being processed in the next stage. In this study the data used is question data in the form of text from the results of interviews from unstructured questionnaires. Therefore, this process is needed because it will be used to convert unstructured data into structured (normalization) in order to be processed at a later stage. In this study the preprocessing stages consist of casefolding, cleansing, tokenizing and remove stopword, and TF-IDF weighting.

1.2.1. Case Folding

Case Folding is the process to transform text into the same case into lowercase or uppercase) [10]. steps of case folding can be seen in Figure 1.

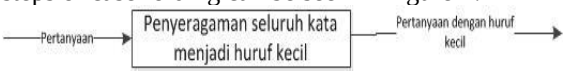


Figure 1 Stage Process of Casefolding

1.2.2. Cleansing

Cleansing is a process to cleaning up words that are not needed. The word is contains url, hashtag (#), username (@username), and email. Also punctuation marks such as dots (.), Commas (,), and other punctuation will be omitted. [10]. Stages of case folding can be seen in Figure 2.

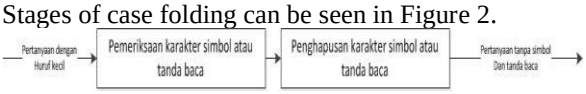


Figure 2 Stage Process Cleansing

1.2.3. Tokenizing

Tokenizing is the process of separating words in text into several word sequences separated by spaces or other characters [10]. Stage case folding can be seen in figure 3.

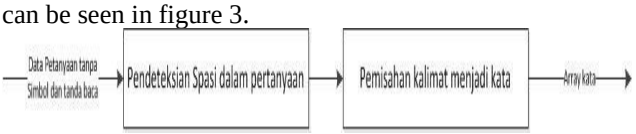
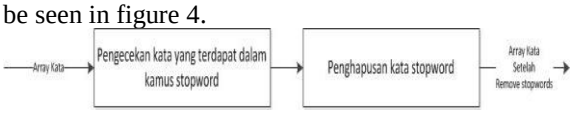


Figure 3 Stage Process Tokenizing

1.2.4. Remove Stopwords

Remove Stopword is a to remove words contained in the stoplist. The words contains unrelevance to the text and are defined in the stopword list [10]. Steps of Remove Stopwords can be seen in figure 4.



1.2.5. TF-IDF Weighting

The TF-IDF method is a method for calculating the weight of every word most commonly used in information retrieval. This method is also known to be efficient, easy and has accurate results [11]. This method will calculate the weight of each term in a document by the formula.

$$w_i = tf_i * IDF_{t_i} \quad \textbf{(1)}$$

where :
d : dokumen on -d
t : *term on* -t from keyword
tf_i : *term* frequency
W_i : document weighting on-d from term on-t
Value of IDF_{t_i} got from:

$$IDF_{t_i} = log(\frac{D}{df_i}) \quad \textbf{(2)}$$

where:
D : total count of document,
df_i : the number of documents containing the keyword

1.3. Support Vector Machine

Support Vector Machine (SVM) is a learning system that uses space hypothesis in the form of linear functions in a feature space (feature space) high-dimensional, trained with learning algorithms based on optimization theory by implementing learning bias derived from the theory of statistical learning. The theory underlying SVM itself has evolved since the 1960s, but it was only introduced by Vapnik, Boser and Guyon in 1992 and since then SVM has grown rapidly.

The concept of classification with SVM can be explained simply as a search for the best hyperplane that serves as a separator of two data classes in the input space. Figure 1 shows some patterns that are members of two classes: +1 and -1. data incorporated in class-1 is symbolized by red (box), while pattern in class +1, symbolized with yellow (circle) [12].

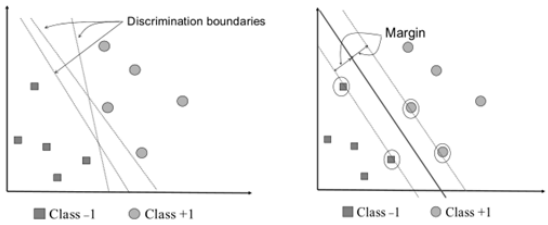


Figure 4 Decision SVM Boundaries

The data residing in this boundary field is called support vector. In the example above, the two classes can be separated by a pair of parallel boundary fields. The first delimiter field limits the first class while the second boundary field limits the second class [13].

Each training data is declared by (x_i, y_i), where i=1,2,...,N, and $x_i = \{x_{i1}, x_{i2}, \ldots, x_{iq}\}^T$ are feature set for training data i, $y_i \in \{-1,+1\}$ declared the class label. Hyperplane SVM linear classification, denoted by

$$\boldsymbol{w}.\boldsymbol{x}_i + b = 0 \quad \textbf{(3)}$$

w and b is the model parameter. is the inner product between w and xi. The data xi that goes into the 1st class is the data that satisfies the following inequalities:

$$w.x_i + b \leq -1 \quad \textbf{(4)}$$

While the data xi that goes into the +1 class is the data that satisfies the following inequalities:

$$w.x_i + b \geq +1$$

Optimal margins are calculated by maximizing the distance between hyperplane and nearby data. This problem is formulated into Quadratic Programming (QP) problems, by minimizing inverse equations. This optimization can be solved by Lagrange Multiplier with equation 5.

$$\max_{\alpha} L_D = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1,j=1}^{n} \alpha_i \alpha_j y_i y_j x_i x_j \quad \textbf{(5)}$$

1.4. Cosine Simlarity

Cosine similarity is used to calculate the query relevance approach to the document. The greater the value of the query vector similarity with the document vector the query is considered more relevant to the document. Cosine of 0^0 is 1, and less than 1 for every other angle. Thus two vectors of the same orientation have cosine similarity of 1, the two vectors in 900 have the similarity of 0. Cosine similarity is mainly used in positive space, where the result is limited (0.1). Cosine similarity then gives a benchmark how similar the two documents [14].

$$CS(d1,d2) = \frac{\sum_{t=1}^{n} W_{t,d_1} W_{t,d_2}}{\sqrt{\sum_{t=1}^{n} W_{t,d_1}^2} \sqrt{\sum_{t=1}^{n} W_{t,d_2}^2}} \quad \textbf{(6)}$$

where :

W_{t,d_1} = document term weighting d_1.

W_{t,d_2} = document term weighting d_2.

2. RESEARCH CONTENTS

In this research, will develop question and answer system in front-office case by applying SVM method. Input data in the form of question will be through preprocessing stage, then classification with SVM method, and search the answer by cosine similarity method. The amount of data used in this study, ie 146 data consisting of 82 data train and 64 test data.

2.1. System Analyze

As for some processes that will run on this system, which in this system for some parts that is training and testing. The following processes performed in this system are shown in Figure 2.

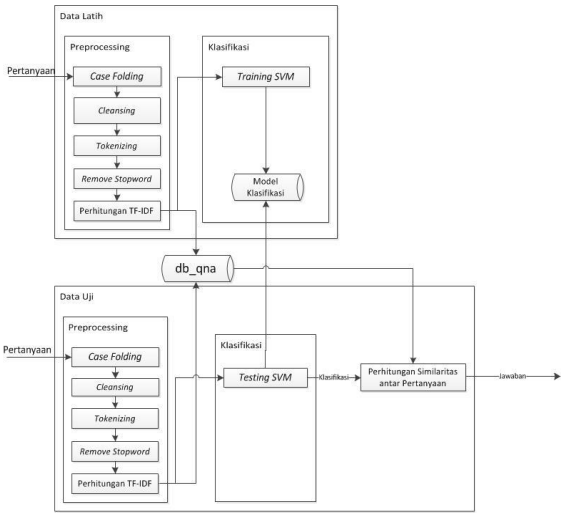


Figure 5 System Architecture

2.2. Analysis Data Input

Data Input used is data in the form of questions. The input data will go through the preprocessing step process

2.2. Analysis Preprocessing

There are several stages done in the preprocessing stage. The preprocessing stage is Case Folding, Cleansing, Tokenizing, Remove Stopword, and TF-IDF weighting. Here is an overview of the Preprocessing step shown in Figure 3.

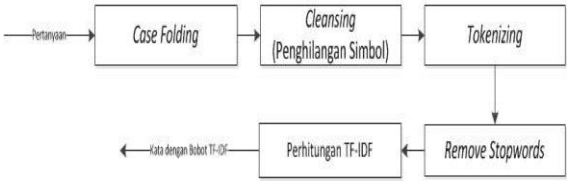


Figure 6 Stage Preprocessing

2.2.1. Case Folding

Case Folding is the process to transform text into the same case into lowercase or uppercase) [10].

2.2.2. Cleansing

Cleansing is a process to cleaning up words that are not needed. The word is contains url, hashtag (#), username (@username), and email. Also punctuation marks such as dots (.), Commas (,), and other punctuation will be omitted. In this study, the cleansing process to remove symbols such as punctuation and emoticons in the text. The following list of cleansing is shown in table 1.

Table 1 Cleansing List

Character	Information
?	Question mark
!	exclamation mark
“	quotation marks
☺	Smile
☹	Sad
😐	Flat
.	fullstop

,	Comma
;	semicolon

2.2.3. Tokenizing

Tokenizing is the process of separating words in text into multiple word sequences separated by spaces or other characters. In this study, separator characters use (") spaces.

2.2.4. Remove Stopwords

Remove Stopword is a to remove words contained in the stoplist. The words contains unrelevance to the text and are defined in the stopword list. Here is a list of Stopword / Stoplist shown in table 2.

Table 2 Stopword/Stoplist

No	*Stopwords*	No	*Stopwords*
1	yang	18	dari
2	untuk	19	menjadi
3	saya	20	masih
4	harus	21	aja
5	saja	22	itu
6	terdapat	23	dalam
7	setiap	24	pada
8	di	25	ini
9	tentang	26	yg
10	ada	27	ke
11	jika	28	mau
12	cara	29	kok
13	tetap	30	soalnya
14	dengan	31	apakah
15	hanya	32	katanya
16	seperti	33	kata
17	atau		

2.2.5. TF-IDF Weighting

At this stage, after the process of remove stopwords will be processed by using TF-IDF weighting. The weighting process is carried out on the term in the question sentence with formula (1), so we get the weight of each term in the sentence that will be used the training process with SVM.

2.3. Support Vector Machine

In this study, the process of classification using SVM method. Classification is done after the preprocessing process. Class The classification used is 7 classes and the SVM method used is One Vs One with the RBF kernel. The 2 stages of the classification process is the training and testing process. Here is an explanation of the training process.

2.3.1. SVM Training Analysis

In the process of SVM training stage, the input data is class data and training data. Where in the data train amounted to 82 data train. Then, proceed to the kernelization and calculation process to get the value of support vector and get the dividing field for the data testing process. The process stages of SVM training are illustrated in Figure 4.

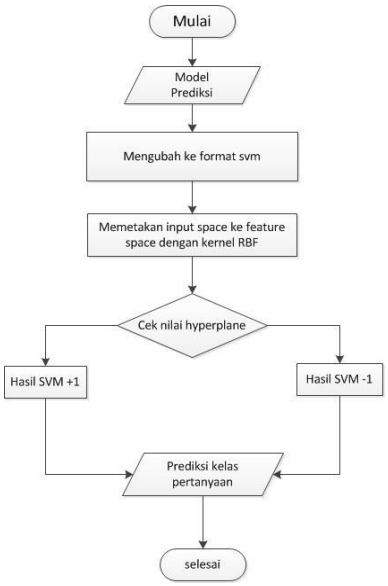


Gambar 7 Stage Training SVM

2.3.2. SVM Testing Analysis

In the test data analysis, test data in the process with preprocessing stages up to TF-IDF weighting process as in the training process. Then it will be classified with SVM.

In Multiclass Support Method Vector Machine One Vs One built $k = \frac{k(k-1)}{2}$ SVM binary, with k is the number of classes. For each model the i-th classification is trained using the entire data, to find the solution of the problem. Whereas in the test data there are 64 data. Then, proceed to the kernelization process and calculation with the dividing field obtained from previous training data. The final result of the SVM test is the question class decision. The process stages of SVM testing described in Figure 5.



Gambar 8 Stage SVM Testing

2.4. Cosine Similarity

In this study, cosine similarity method is used as a search for the test data. This method will look for closeness between the test data on the train data. The final result of this method is the prediction of the answers to the test data.

2.6. Calculate Accuracy

In testing the accuracy of question and answer system research with SVM method is calculated from the accuracy of the questions and answers given. In this test given the data train as much as 82 questions and test data amounting to 64. The Accuracy Result obtained shown in table 3.

Table 3 Accuracy Result

Process	Sum of Question
Training	82
Testing	64

Correct Answer = 25
Total of true classification = 40
Number of questions= 64

$$akurasi = \frac{jumlah\ jawaban\ yang\ benar}{jumlah\ pertanyaan} x100\%$$

$$akurasi = \frac{25}{64} x100\%$$

$$akurasi = 39\%$$

3. CLOSING

3.1. SUMMARY

The conclusions from the test show that the application of svm method for question and answer system in front-office case resulted in 39% accuracy and classified question data yielding 40% accuracy. This is influenced by the amount of training data along with different language patterns from the test data to the training data despite having the same intention to the training data.

3.2. CONCLUSION

In this final project research, there are still many deficiencies that can be improved for further research. The author's suggestion to be submitted is

1. For further research it is suggested to add data for training.Untuk penelitian selanjutnya disarakan untuk tidak mencoba menggunakan *stopwords* apakah mempengaruhi akurasi atau tidak.
2. It is recommended to use different classification methods such as Fuzzy KNN-C.
3. For further research it is advisable not to use remove stopwords whether it affects accuracy or not.
4. For further research it is suggested to use other SVM multiclass methods such as ECOC or One Vs All.

BIBLIOGRAPHY

[1] R. A. Ewerton J. Wantroba, "An Interactive Question Answer System with dialogue for a Receptionist avatar," *2th Latin American Robotics Symposium, 2015 Third Brazilian Symposium on Robotic*, pp. 360-365, 2015.

[2] Y. K. Maharlika, "Rule Based Method Implementation For Question Answerin System in A Front Office Study Case," *Perpustakaan UNIKOM*, 2016.

[3] R. H. Gusmita, Y. Durachman, S. Harun, A. F. Firmansyah, H. T. Sukmana dan A. Suhaemi, "A Rule-Based Question Answering System on Relevant Documents of Indonesian Quran Translation," *International Conference on Cyber and IT Service Management (CITSM)*, pp. 104-107, 2014.

[4] F. S. Gharehchopogh dan Y. Lotfi, "Machine Learning based Question Classification Methods in the Question Answering Systems," *International Journal of Innovation and Applied Studies*, vol. 4, pp. 264-273, 2013.

[5] A. Mathur dan M. Haider, "Question Answering System : A Survey," *International Conference on Smart Technologies and Management*, pp. 44-57, 2015.

[6] H. Abdelmasser, M. Ragah, B. Farouk, R. Mohamed, A. Mohammed and N. El-Makky, "Al-Bayan : An Arabic Question Answering for the Holy Quran," *Workshop on Arabic Natural Language Processing (ANLP)*, pp. 57-64, 2014.

[7] W. Ahmed, A. PV dan B. Anto, "Web-Based Arabic Question Answering System using Machine Learning Approach," *International Journal of Advanced Computer Science*, vol. 8, 2017.

[8] S.-J. Yen, Y.-C. Wu, J.-C. Yang, Y.-S. Lee, C.-J. Lee dan J.-J. -Liu, "A Support Vector Machine-Based Context Ranking Model For Question Answering," *Information Sciences*

224, pp. 77-87, 2013.

[9] N. Othman dan R. Faiz, "Question Answering Passage Retrieval and Re-rangking Using N-grams and SVM," *Computacion y Sistemas*, vol. 20, pp. 483-494, 2016.

[10] A. M. Bachtiar dan A. R. Sentiaji, "Analisis Sentimen Terhadap Acara Televisi Berdasarkan Opini Publik," *Jurnal Ilmiah Komputer dan Informatika*, 2014.

[11] A. A. Maarif, "Penerapan Algoritma TF-IDF Untuk Pencarian Karya Ilmiah," 2015.

[12] E. Prasetyo, Data Mining, Mengolah Data Menjadi Informasi Menggunakan Matlab, Yogyakarta: ANDI, 2014.

[13] K. Sembiring, "Penerapan Teknik Support Vector Machine untuk Pendeteksian Intrusi pada jaringan," pp. 2-26, 2007.

[14] J. Pardede dan J. Sinataria, "Implementasi Maximum Marginal Relevance Dan Matriks Cosine Similarity Pada Aplikasi Peringkasan Dokumen," *Itenas Library*, pp. 1349-1352.

[15] Suprapto, "Bahasa Pemrograman untuk Sekolah Menengah Kejuruan," Direktorat Pembinaan Sekolah Menengah Kejuruan Direktorat Jenderal Manajemen Pendidikan Dasar dan Menengah Departemen Pendidikan Nasiona, 2008.

[16] R. A. Yuana, "Dasar-dasar Pemrograman PHP".

[17] M. Fowler, UML Distilled Edisi 3, Panduan Singkat Bahasa Pemodelan Objek Standar, vol. 53, Yogyakarta: ANDI Jogyakarta, 2013, pp. 1689-1699.

[18] I. Fathansyah, Basis Data, Bandung: Informatika Bandung, 2016.

[19] T. Karlita, S.Kom.,M.Kom, A. S. Ahsan, S.Kom dan A. R. Barakbah, S.Kom.,Ph.D, "Logika dan Algoritma," 2013.

[20] C.-W. Hsu dan C.-J. Lin, "A Comparison of Methods for Multi-Class Support Vector Machines," *Department of Computer Science and Informatics Engineering Nationa Taiwan University*, 2002.

[21] O. Nurdiana, J. dan D. Nursantika, "Perbandingan Metode Cosine Similarity dengan Metode Jaccard Similarity pada Aplikasi Pencarian Terjemah Al-Qur'an dalam Bahasa Indonesia," vol. 1, pp. 59-63, 2016.

[22] E. Prasetyo, Data Mining Konsep dan Aplikasi menggunakan Matlab, Yogyakarta: ANDI Yogyakarta, 2012.

[23] H. Tohari, Astah : Analisis serta perancangan sistem informasi melalui pendekatan UML, Yogyakarta: Andi Yogyakarta, 2014.

[24] "UML : ATM Case Study".

[25] A. Sunyoto, Membangun Web dengan Teknologi Asyncronous, Yogyakarta: ANDI, 2007.

[26] I. H. Witten, Text Mining, New Zealand, 2002.

[27] J. M. Pricila, "Comparison of Approaches multiclass SVM classification Articles Speak Indonesia," *Perpustakaan UNIKOM*, 2016.

[28] R. S. Pressman,Ph.D dan B. R. Maxim,Ph.D, Software Engineering, A Practicioner's Approach, Eight Edition, Mc Graw Hill Education, 2015.

[29] I. Sommerville, Software Engineering Edisi 8 (Rekayasa Perangkat Lunak), New York: Mc Graw Hill, 2015.

[30] Sugiyono, Metode Penelitian Kuantitatif, Kualitatif Dan R&D, Alfabeta, 2012.