Internet Answering Machine Based on Knowledge Representation:

A Client Application for Human-Machine Conversation Modelling

Bayu Setiaji
Department of Informatics Engineering
Stmik Amikom Yogyakarta
Yogyakarta, Indonesia
e-mail: bayusetiaji@amikom.ac.id

Ferry Wahyu Wibowo
Department of Informatics Engineering
Stmik Amikom Yogyakarta
Yogyakarta, Indonesia
e-mail: ferry.w.wibowo@ieee.org

Abstract **– An internet answering machine is a machine which could be used as a conversation with a human via the internet networking. Usually, a machine is a server containing a knowledge. The appropriate mechanism could be done by many methods such as a pattern-matching. The user could ask to the machine via a client program and then the server will give a response. The received response then will be parsed and generally displayed in the chat history. In this paper, the machine implementation is a web service written using a PHP programming language and the knowledge, meanwhile processing of a pattern-matching is in the DBMS MySQL, but the data communication format is in the JSON document. The use of a web service could be made in many client programs with many platforms. In this paper, the client program display has been written using PHP and .NET programming languages.**

Keywords – intelligent system; internet; machine; natural language; web service

I. INTRODUCTION

A communication is a way for human interacts to each other. This aspect makes some innovations how humans could use the machine to be functioned as human's friends. In this case, the machine could answer that is asked by a human. The question is how to make machine's knowledge and how the machine could make an appropriate decision for answering human questions. A chatterbot or known as chatbot is one way to make a conversation between human and machine [1]. Advancing of the information technology and communication has been complex in implementing of artificial intelligence (AI) systems. The artificial intelligence in the computation field has some definitions, but all of the definitions have one conclusion. The AI is a part of computer science which is representing a knowledge more over using symbols than numerical and processing an information based on the heuristic method or using some rules. The AI systems are approaching of human activities such as decision support systems (DSS), robotics, natural language processing (NLP), expert systems, etc. The AI has some perspectives, namely 1) Intelligent, the AI is about how to make an intelligence machine and can do anything as which can be done by a human; 2) Business, the AI is a set of functioned tools and has a methodology to get a solution of business; 3) Programming, The AI and its parts are study of symbolic programming than numerical processing such as words, alphabets, and numbers. Besides that, the AI is included into study how to conclude problems as goal and study about searching process. Some domain of the AI's research could be divided into some aspects, namely 1) game, the computer game implements some rules using heuristic technique in determining of each problem; 2) natural language processing (NLP), giving an ability to the computer that may communicate between user and computer (human to machine) or computer to computer (machine to machine) in the human language; 3) robotic and sensor system, combination of sensor system and signal processing with the AI; 4) expert system, computer program has an ability as expert to help the users in their specific problems. Even in the artificial intelligent fields, there are some hybrid methods and adaptive methods those make more complex methods. Not only that but nowadays there is also a hybrid of natural language and intelligent systems those could understand human natural language. These systems can learn themselves and renew their knowledge by reading all electronics articles those has existed on the internet. Human as the user can ask to the systems like usually did to another human. These systems are often known as internet answering-engines.

The natural language processing (NLP) is making a program that has an ability to understand a human language. The main representations of the natural language are speaking and writing. The natural language computerization modeling may interact with both human and computer with the media of natural language. The natural language computerization modeling will involve computation process where the language should be implemented as a symbol set with such rules. There are three stages of analysis to understand a natural language namely 1) parsing, an activity to analyze sentence syntax structures. The analyzing is due to the identification of a main linguistic correlation such as subject, predicate, and object. The parser gives a diagram frame to the semantic interpretation; 2) semantic interpretation, this stage results in a meaning representation from the texts. The semantic representation is using the knowledge about the word meaning and linguistic structure as object activity or verb transitivity; 3) Structures from the

DOI 10.5013/IJSSST.a.17.35.17

ISSN: 1473-804x online, 1473-8031 print

knowledge base, added in the sentence representations to yield an extended representation of sentence meaning. The resulted structures represent a meaning of natural language texts and used by the system to advancing process. The NLP technology could do many processes in the natural language that is usually used by a human. Generally, this system has an input and output forming a text. Some of the NLP application categories are 1) natural language translator is translating one language into other languages. An example of this application is Google translate; 2) translator of natural language to artificial language is a translator from human language to the artificial language executed by machine or computer. An example of this is user could delete all files in the computer using word "computer, delete all files", then translator would translate the word as command of DEL *.*; 3) text summarization is system could make summarization of important things from given descriptions. The NLP process focuses on the sentence.

In addition, the internet answering-engines, currently on the internet also begins many applications of chatter-boot or known as chatbot which is often aimed for such purposes or just entertainment [2]. This application work is very simpler because the knowledge already programmed in advance [3]. One of the methods used in this application is to match the pattern (pattern-matching) [4]. The chatbot would match the input sentence from the speaker or user with a pattern that has existed on the knowledge. Each pattern paired with the knowledge of chatbot which taken from various sources. The input sentence prepared as the materials of chat pattern [5]. The chat patterns modeled in the pattern template stored in a relational database management system (RDBMS) tables. The process of pattern matching is using a sentence similarity measurement scores. The calculation method to achieve the scores of sentence-similarity measurement may apply bigram method as one way of measurement methods, although there are some other methods. The function programs for pattern matching and other support purposes written as a program stored in the RDBMS. Another knowledge storage method of chatbot is artificial intelligence markup language (AIML) [6,7]. The AIML has modularly knowledge processes. This system is a web service-based which could be accessed by the client. The chat patterns are language knowledge in the format of AIML stored in the database. This system could be added specific knowledge modules [8,9].

In this paper shows the collected facts as prepared references for chat-pattern and this chat uses the Indonesian language. The chat used in this project is commonly Indonesian conversational pattern and the RDBMS used in this project is MySQL. When connecting chat application to the database, it can miss in defining a sentence and how to respond it. So knowledge representation in the database tables and implementation of structured query language (SQL) in the pattern-matching operation are very needed. A data those have been modeled on the pattern of the conversation would be tested using a series of scenarios. The results of the conversation with the chatbot would be crosschecked back to the basic pattern. This is done to add some knowledge to the database because it hasn't been modeled before. So if the input sentences don't match in the database then it will be remodeled. The Application of chatbot is made using PHP language and the knowledge is taken from many sources.

II. Related Works

A natural language processing (NLP) gives the capability of a computer allows communication to happen between user-to-computer or human-to-machine and computer-to-computer or machine-to-machine using human natural languages. There are three analyses to understand natural language i.e. parsing, semantic interpretation, and knowledge-based structures. The parsing is an analysis of sentence syntax structures. In this step, identification of main linguistic relations is done to parse into subject, predicate, and object of the sentences. The semantic interpretation step yields meaning representation of the texts. The semantic interpretation uses knowledge of word meaning and linguistic structure such as noun or verb transitivity. Actually, processing focus in NLP is a sentence. A sentence could be meant as biggest syntaxes consist of two or more words. A structural relationship between inter-word and inter-sentence is different. Between both sentence and word, there are two media syntax units i.e. clause and phrase. A clause is a syntax unit that consists of two or more predicate elements. The predicate elements are a subject, predicate, object, complement, and adverb. The phrase is a syntax that consists of two or more words which are not including predicate elements.

Word categories in the Indonesia language consist of a verb (V), adjective (Adj), adverb (Adv), noun (N), preposition (Prep), and conjunctive (Conj). Based on the phrase, the phrase in Indonesian language divided into main elements. The phrase categories in Indonesian language are Phrase of Noun (PhN), Phrase of Adjective (PhAdj), Phrase of Adverb (PhAdv), and Phrase of Prepositional (PhPrep). Kinds of word grouped in such category could have syntax function and difference semantic role in such sentence. A word can be functioned as subject, predicate, object, complement, and adverb. The relation among form, category, and function is more clearly with this Indonesian language sentence "induk singa sedang mengejar rusa muda untuk anaknya", the description of this sentence can be shown in Table I.

TABLE I. Relation Among Form, Category, and Function of Sentence Elements

Form	Word	Phrase	Function
Induk	N	PhN	Subject
Singa	Pron		
Sedang	Adv	PhV	Predicate
Mengejar	V		
Rusa	N	PhN	Object
Muda	Adj		
Untuk	Prep	PhPrep	Complement
Anaknya	N		

The application of chatbot core that has been made is applying database in the testing purposes. The application of database development tool for this project employed MySQL Workbench 6 and tools for generating application employed programming language of Pascal, Java, and PHP. In the chatbot, if the pattern is matched, it will give suite template as a response to the user. The pattern template has been employed as the main knowledge of the chatbot stored in the database. The use of MySQL database in the chatbot is only limited to store the knowledge. All codes for requiring a pattern-matching written in programming language, so to make a service in other languages, the codes need to be rewritten.

The pattern template storage as knowledge in the relational database management system (RDBMS) or called as database allows the use of structured query language (SQL) to handle the process of pattern matching. The RDBMS already available in many built-in functions or procedures and can be made a user-defined stored program which can be called using SQL. This allows many programming languages can be implemented easily as a query of the database to send and receive an input response.

A. Bigram

The probability of calculating sentence could be represented mathematically as equation (1) [10].

$$\rho\left(\left|\psi_i\right|\right) \tag{1}$$

if ψ = “saya makan buah”, a sentence in Indonesian language which means “I eat fruit”, so it can be written as ρ(|ψ|=3, ψ_1=”saya”, ψ_2=”makan”, ψ_3=”buah”) = ρ(ψ_1=”saya”|ψ_0=”<s>”) * ρ(ψ_2=”makan”|ψ_0=”<s>”, ψ_1=”saya”) * ρ(ψ_3=”buah”|ψ_0=”<s>”, ψ_1=”saya”, ψ_2=”makan”) * ρ(ψ_4=”</s>”|ψ_0=”<s>”, ψ_1=”saya”, ψ_2=”makan”, ψ_3=”buah”). The sentence is started with <s> and ended with </s>. The probability of ψ_0=<s> is 1 or written as ρ(ψ_0=<s>)=1. From this case, it can also be written as equation (2).

$$\rho(\psi) = \prod_{i=1}^{|\psi|+1} \rho\left(\psi_i \middle| \psi_0 \ldots \psi_{i-1}\right) \tag{2}$$

For bigram model, it adds one word of context that can be represented as equation (3).

$$\rho\left(\psi_i \middle| \psi_0 \ldots \psi_{i-1}\right) \approx \rho\left(\psi_i \middle| \psi_{i-1}\right) \tag{3}$$

So, the equation 3 can be enacted a linear interpolation using Witten-Bell smoothing algorithm. The Witten-Bell smoothing algorithm is applied to predict the probability of bigram model with zero counts or $\rho(\psi_{i-1}|\psi_i)=0$ [11]. From Good-Turing estimation, the total mass of counts with a zero count in distribution is the number of things with one count. In this case, the probability mass in the back-off distribution should be $\xi(\psi_{i-1})/\varsigma(\psi_{i-1})$. Where $\xi(\psi_{i-1})$ is a number of unique words after ψ_{i-1} and $\varsigma(\psi_{i-1})$ is a count of ψ_{i-1}. So the probability for bigram is written as equation (4).

$$\rho\left(\psi_i \middle| \psi_{i-1}\right) = \omega\rho_{MLE}\left(\psi_i \middle| \psi_{i-1}\right) + \left(1-\omega\right)\rho\left(\psi_i\right) \tag{4}$$

which $\rho_{MLE}(\psi_i|\psi_{i-1})$ is the probability of maximum likelihood estimation which can be written as equation (5).

$$\rho_{MLE}\left(\psi_i \middle| \psi_{i-1}\right) = \frac{\varsigma\left(\psi_{i-1}\psi_i\right)}{\varsigma\left(\psi_{i-1}\right)} \tag{5}$$

while $\rho(\psi_i)$ is the estimated probability which can be written as equation (6).

$$\rho\left(\psi_i\right) = \omega\rho_{MLE}\left(\psi_i\right) + \frac{\left(1-\omega\right)}{v} \tag{6}$$

In this case, the symbol of v is the vocabulary size. Applying the value of ω is taken to make easier calculation and ω is a value set which can be written as equation (7).

$$\omega = 1 - \frac{\xi\left(\psi_{i-1}\right)}{\xi\left(\psi_{i-1}\right) + \varsigma\left(\psi_{i-1}\right)} \tag{7}$$

B. Sentence Similarity Measurement

A semantic similarity is giving a score for a semantic relation between two sentences or strings. So, if there are two sentences or strings, from measuring it can be determined the similar of two sentences or strings. The higher score of the sentence semantic similarity, the most similar meaning of two sentences. The score of the sentence semantic similarity is from 0 until 1. The equation of sentence similarity represented by equation (8).

$$\frac{\varsigma\left(s_1 \in s_2\right) \cup \varsigma\left(s_2 \in s_1\right)}{\varsigma\left(s_1\right) \cup \varsigma\left(s_2\right)} \tag{8}$$

The symbol of ς is a count of the sentence or string that is symbolized as s. The used method to compute the semantic similarity between two sentences could be written as bellow,

a. Each sentence is divided into a list of tokens
b. The divided sentence can implement bigram set [12]
c. After getting the divided sentence, it is counted and applied the equation (8)

For example, this method can be written as bellow,

s_1 = “singa”
s_2 = “singga”

so s1 and s2 can be divided into bigram set as

s_1 = {“si”, “in”, “ng”, “ga” } ≈ 4
s_2 = {“si”, “in”, “ng”, “gg”, “ga” } ≈ 5
$s_1 \in s_2$ = {“si”, “in”, “ng”, “ga” } ≈ 4
$s_2 \in s_1$ = {“si”, “in”, “ng”, “ga” } ≈ 4

Thus the similarity score for these sentences is obtained as follow,

$$\frac{4\cup 4}{4\cup 5} \approx \frac{8}{9} \approx \quad 0,889$$

C. *Database*

A relational database management system (RDBMS) is a program set that is employed to define, manage, and process a database. The database is a structure that is built to be functioned as data storage. Some categories of RDBMS products are Oracle, MySQL, PostgreSQL, Microsoft SQL Server, etc. Each RDBMS has a different license, there are free and commercial. MySQL is a server database or RDBMS software that can manage the database and can store data in many numbers. It can be accessed by multi-user and can do multi-threaded.

Structured query language (SQL) is one of 4th generation language developed by an international business machine (IBM) corporation in San Jose Research Laboratory. The characteristics of SQL are request oriented and non-procedural, so it is easy to learn because the used syntaxes are almost same as a common human language to communicate. That's way SQL is more flexible in the usage. Besides that, SQL is a non-case sensitive, so many of DBMS vendors nowadays are using SQL as their product in the standardization. The SQL consists of three languages namely data definition language (DDL), data manipulation language (DML), and data control language (DCL). The DDL is used to arrange the table and index structures. Some of the basic instructions are shown bellow,

1. CREATE

The instruction of CREATE is used to make a table and its structures. The example of this usage is shown bellow,

```
CREATE TABLE address (
Id INT PRIMARY KEY,
Name VARCHAR(50),
Address VARCHAR(100))
```

2. ALTER

The instruction of ALTER is used to modify table structures such as add or subtract of a column, change of column data type, etc. The example of this usage is shown bellow,

```
ALTER TABLE address ADD address
VARCHAR(255)
```

3. DROP

The instruction of DROP is used to delete a table. The example of this usage is shown bellow,

```
DROP TABLE address
```

The DML is used to add, update, and delete data. Some of the basic instructions are shown bellow,

1. INSERT

The instruction of INSERT is used to add a new data in the table. The example of this usage is shown bellow,

```
INSERT INTO address VALUES (
‘1’, ‘Bayu’, ‘+62274555283’, ‘SLEMAN’
)
```

2. UPDATE

The instruction of UPDATE is used to update data. The example of this usage is shown bellow,

```
UPDATE address SET name = ‘Bayu
Setiaji’ WHERE id = ‘1’
```

3. DELETE

The instruction of DELETE is used to delete one row of data in the table. The example of this usage is shown bellow,

```
DELETE FROM address WHERE WHERE id =
‘1’
```

The DCL is used to arrange the roles of users in the database. The main instructions of DCL are GRANT and REVOKE. The instruction of GRANT is used to add the roles of users, while REVOKE is deleted the roles of users.

For making database, in this paper used MySQL workbench. MySQL workbench is a visual tool integrated with MySQL server to develop SQL, administration, and database design. MySQL workbench is the successor of DBDesigner from Fabforce.net and replacing MySQL GUI tools bundle. The application of MySQL workbench is shown in Figure 1.

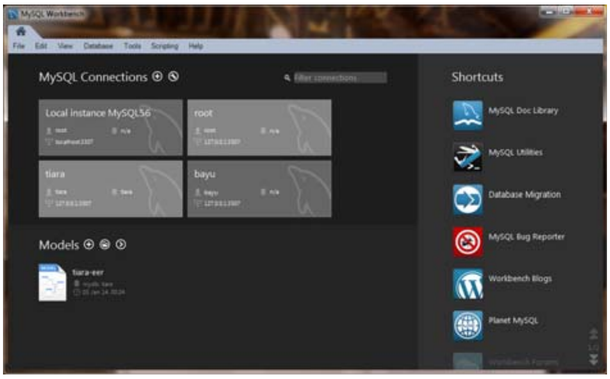


Figure 1. MySQL Workbench 6

As shown in Figure 1, the first page of MYSQL workbench 6 has some shortcuts to log in every user, database model, and some tools.

III. Methodology

The process of pattern similarity has some methods. A similarity level between both input and pattern could be obtained from sentence-similarity measurement scores. This process is run in the RDBMS. The global architecture of the chatbot scheme design is shown in Figure 2.

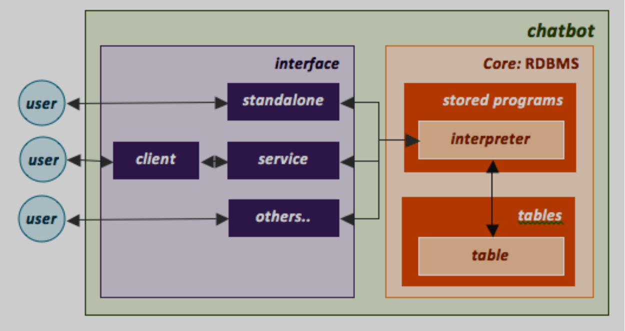


Figure 2. Design of Chatbot

The design of chatbot consists of core and interface accessing that core. The core is in RDBMS being the database. The database consists of tables to store knowledge, while the interpreter is a stored program of function and procedure sets for requiring of pattern matching. The interface could be a standalone application that can be employed by a user for chatting or conversation. It can also be employed by service that needs an additional client application to converse with the user. This application in the interface side can be expanded more over as user needs it and can also be written using other programming languages.

The chatbot core is in the RDBMS covers knowledge storage and pattern-matching process. Some limitations of chatbot in this method have some requirements [1]:

1. Chatbot should be able to differ each conversation session that is running, so it has to store data due to the conversation session such as session of identity (sessionid), user name on the session. The session of identity must always be sent together with user input by application in the interface side a long conversation process
2. Chatbot must store knowledge in the pattern template form
3. All user inputs must be free of misspellings, punctuation, and must be in lower case, so to anticipate these cases the chatbot should be able to do normalization of the input that doesn't fit
4. For the purposes of misspellings correction, the chatbot should have a list of misspelled words and the correction stored in the tables of database
5. The chatbot should be able to pick up the keywords from the user input, so the chatbot should have a list of keywords which is stored in the tables of database
6. The chatbot should be able to do a search template using a sentence-similarity measurement scores between both pattern and input. The searching of pattern is narrowed based on the result identification as described at point 4
7. The chatbot should keep a conversation log containing session of identity, time, input, and response.

Some points those have been identified above provide a global description scheme of the chatbot core related with conversation processing that is shown in Figure 3.

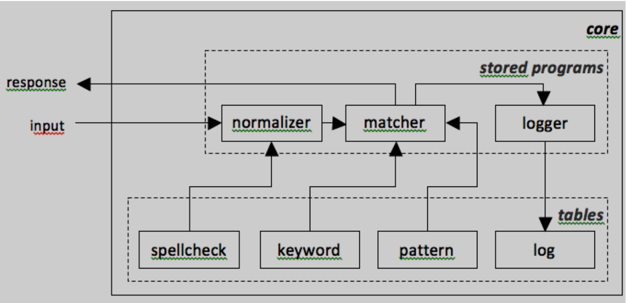


Figure 3. Scheme of Chatbot Core

Based on Figure 3, the chatbot core consists of tables and stored programs.

A. *Tables*

The table blocks consist of main tables as scheme and supporting tables i.e. spellcheck, keyword, pattern, and log. The spellcheck stores list of misspelling words from user input and correction. The keyword stores a list of keywords that is probably found in the user input. The keywords are used to narrow the range of pattern searching. The pattern stores pattern-template pairing and roles as main knowledge. Each pattern could be paired with one or more templates and each template could be paired with one or more patterns, so it would be divided into three tables i.e. pattern, template, and pattern template. The pattern is functioned to store pattern containing pattern of identity (patternid) and pattern, the template is functioned to store template containing template of identity (templateid) and template, and the pattern template roles as table connecting pattern and template tables. The table of convlog stores conversation history containing session of identity, time, user input, and a response given by the user. The session stores attribute of such conversation session including session of identity, user name, and other attributes. The array is a temporary table outside the main tables. It is used as array data structure representation. This table is only supporting for internal operation requirements of array operation function. And tid is used as supporting table for id generator process that is applied in other requiring tables. These tables contain id as id name, counter as counter of identity, and table of rtable as table name reference.

B. *Stored Programs*

The stored program is containing stored procedures and functions for pattern-matching requirements. The normalizer is a function to norm user input that corrects spellings, eliminates punctuations, and changes into lower case. The matcher is the main function in pattern matching to find appropriate template based on the sentence-similarity measurement scores between both input and pattern. Before doing pattern-matching process, it needs to take keywords on the input, so it is a just pattern with same keywords that will be matched. The logger is a procedure functioned as conversation history storing for user input and response of question. The array is a set of functions and procedures for array tables used for the internal operations of 3 major processes that have been described previously. The array of functions and procedures includes array_intersect() which is a procedure for operating the intersection of two arrays, array_push() which is a procedure for operating the push element to the array, array_pop() which is a function for pop operation element of the array, array_count() which is a function to count the number of elements in the array, array_clear() which is a procedure to clear the contents of the array, and bigram which is a procedure for preparing bigram of a string.

C. Design

Based on the problem identification, the database that is built consist of some tables, process input of normalization and pattern matching with other supporting tables like the tables of spellcheck, keywords, pattern, template, pattern_template, convlog, session, array, and tid. The inter-table relationship is shown in Figure 4.

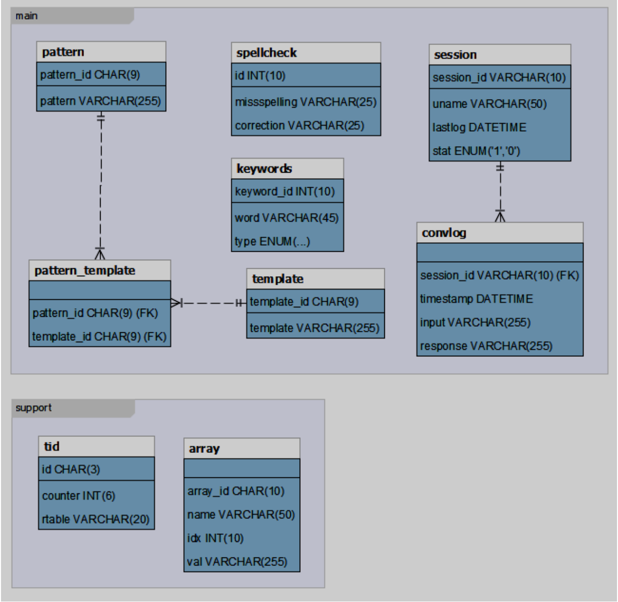
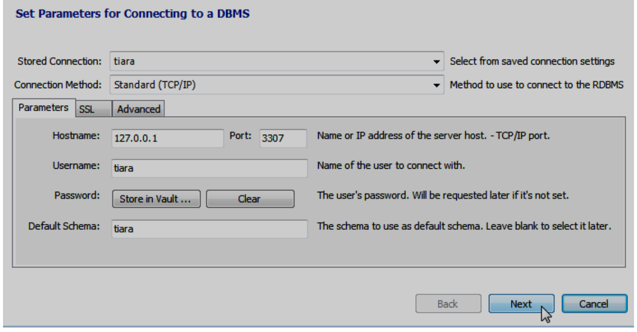


Figure 4. Entity-Relationship Diagram of Chatbot

Figure 4 shows that the tables of pattern, template, pattern_template, spellcheck, keywords, session, and convlog are grouped as main tables. Meanwhile other tables are grouped as supporting tables.

IV. RESULTS

The process of configuration between both application and database should be concerned about the method. In making a table of the database for chatbot, it had implemented a forward-engineering technique. This technique is generating Entity Relationship (ER) into DDL scripts those could be executed as table generating. All designs of tables and stored programs had been implemented in the RDBMS MySQL. Before the testing process was done, it ought to be entered some knowledge which input sentence patterns stored in the pattern table and response sentences stored in the template table. In additional, it had to be entered mapping as a representation of the relationship between both pattern and template stored in the pattern_template table. The set parameters for connecting to a DBMS is shown in Figure 5.


Set Parameters for Connecting to a DBMS

Stored Connection: tiara — Select from saved connection settings

Connection Method: Standard (TCP/IP) — Method to use to connect to the RDBMS

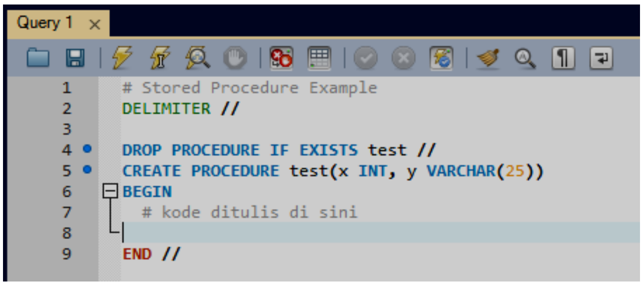
Hostname: 127.0.0.1 Port: 3307 — Name or IP address of the server host. - TCP/IP port.

Username: tiara — Name of the user to connect with.

Password: — The user's password. Will be requested later if it's not set.

Default Schema: tiara — The schema to use as default schema. Leave blank to select it later.

Figure 5. Wizard Set Parameters for Connecting to a DBMS



```
# Stored Procedure Example
DELIMITER //

DROP PROCEDURE IF EXISTS test //
CREATE PROCEDURE test(x INT, y VARCHAR(25))
BEGIN
  # kode ditulis di sini

END //
```

Figure 6. Tab Query

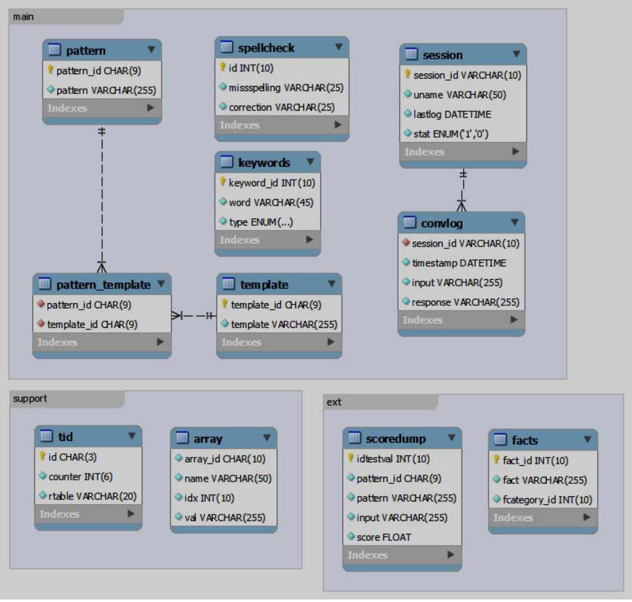
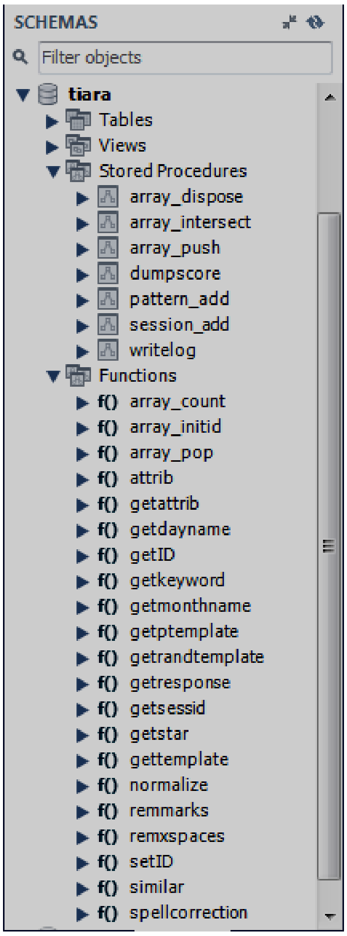


Figure 7. ER Diagram Generated Using MySQL Workbench

The stage of stored program is code writing process of stored procedure and stored function from the previous design and implemented for the next process. Tab query for writing all codes in the MySQL workbench is shown in Figure 6. The entity relation (ER) diagram has been made in the design stage could be generated into default notation of MySQL workbench using Barker's notation and Crow's foot. This ER diagram is shown in Figure 7. The stored program that has been written could be stored in the database. All of the stored programs those have been stored in the database is shown in Figure 8.



SCHEMAS
Filter objects
tiara
Tables
Views
Stored Procedures
array_dispose
array_intersect
array_push
dumpscore
pattern_add
session_add
writelog
Functions
f() array_count
f() array_initid
f() array_pop
f() attrib
f() getattrib
f() getdayname
f() getID
f() getkeyword
f() getmonthname
f() getptemplate
f() getrandtemplate
f() getresponse
f() getsessid
f() getstar
f() gettemplate
f() normalize
f() remmarks
f() remxspaces
f() setID
f() similar
f() spellcorrection

Figure 8. Stored Program

Some tests had been applied to know the functionality of the application. A modular test was done to check and analyze the functionality of the stored program. This test was done with executing each stored program using various parameters as inputs and analyzing outputs.

TABLE II. RESULT OF FUNCTION TESTING OF REMMARKS()

No	Input	Result
1	Halo, apa kabar	Halo apa kabar
2	Sekarang hari apa?	Sekarang hari apa
3	Sekarang hari Minggu.	Sekarang hari Minggu
4	Senin; Selasa; Rabu	Senin Selasa Rabu
5	Tanda Seru!!	Tanda Seru
6	Next: continue	Next continue
7	Under_score	Under_score
8	Stripped-line	Stripped-line

A normalizer consists of some stored programs those have been implemented to process input normalization. A function of remmarks() is employed to omit punctuation of dots, commas, semicolons, colons, exclamation points, and question marks. Testing of the function of remmarks() is shown in Table II.

According to the Table II shows that testing results of number 7 and 8 yield same strings as input strings because the function of underscore (_) and stripped line (-) hasn't been included in the function of remmarks(). Thus the function of remxspaces() has been employed to eliminate more spaces. Testing of the function of remxspaces() is shown in Table III.

TABLE III. RESULT OF FUNCTION TESTING OF REMXSPACES()

No	Input	Result
1	Halo, apa kabar	Halo,apa kabar
2	Halo	Halo

According to the Table III, the function of remxspaces() has been successful to eliminate spaces which are more than one including tab as it is shown in number 1. While the function of spellcorrection() is employed to fix spelling in writing. The correction of spelling depends on the number of data that has been stored in the table of missspelling. Testing of the function of spellcorrection() is shown in Table IV.

TABLE IV. RESULT OF FUNCTION TESTING OF SPELLCORRECTION()

No	Input	Result
1	Halo pa kabar	Halo apa kabar
2	Kamu sdh tau blm	Kamu sudah tahu belum
3	Brp jml bintang di langit	Berapa jml bintang di langit
4	Aq ga tau	Aku ga tahu

According to the Table IV, the function of spellcorrection() could be employed to fix spelling except for word of "jml" as stated in the number 3 because it hasn't been stored in the table of missspelling. And the function of normalize() has been employed to call function of remmarks(), remxspaces(), and spellcorrection(). These functions are used in the block of matcher. Testing of the function of normalize() is shown in Table V.

TABLE V. RESULT OF FUNCTION TESTING OF SPELLCORRECTION()

No	Input	Result
1	Hai…. pa kbr?	Hai apa kabar
2	Hitung! Brapa: 2+3	Hitung Berapa 2+3
3	Jarak Jauh sekali	Jarak jauh sekali

The block of matcher consists of stored programs for requiring a pattern-matching process. The function of getkeyword() is employed to get the keywords in the input. The keywords identification in the input depends on the keywords list stored in the table of keywords.

TABLE VI. RESULT OF FUNCTION TESTING OF GETKEYWORD()

No	Input	Result
1	Apa kabar kamu	[[:<:]]apa[[:>:]]
2	Sekarang hari apa	[[:<:]]apa[[:>:]]
3	Benarkah 1+1=2	.
4	Di mana ibukota Indonesia	.

The output of this function is a string which is applied as a pattern in the regular process of the function of gettemplate(). Testing of the function of getkeyword() is shown in Table VI. According to the Table VI, the testing results of number 1 and 2 yield a regular expression patterns. While testing results of number 3 and 4 yield dot (.) because the keywords weren't found in the pattern table. The function of similar() is employed to count similarity scores of two strings. This function has 2 parameters of string to be compared with the similarity scores. Testing of the function of similar() is shown in Table VII.

TABLE VII. RESULT OF FUNCTION TESTING OF SIMILAR()

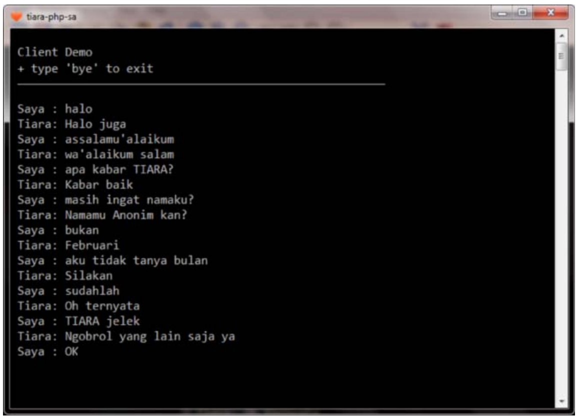
No	String Input(s)		Score
	1	2	
1	siapa namamu	siapa namamu	1
2	siapa namamu	namamu siapa	0,81818
3	siapa namamu	siapa nama kamu	0,92000
4	siapa namamu	Test	0

According to the Table VII, testing number 1 yields score of 1 because the string inputs of 1 and 2 have same sentences. The string input of 1 is the reference and the string input of 2 is the variation. The function of gettemplate() is employed to get the response from the input. The template that becomes a response can differ from one to another when it results in responses because the response has been set randomly. Testing of the function of gettemplate() is shown in Table VIII.

TABLE VIII. RESULT OF FUNCTION TESTING OF GETTEMPLATE()

No	Input	Result
1	Apa kabar?	Kabar baik
2	Apa kabar?	Baik
3	Apa kabar?	Alhamdulillah kabar baik
4	Apa kabar?	Baik

The application has been built to make a conversation with a chatbot. The application is standalone in the console and it also needs libraries to call the database. For the application using PHP language to make a conversation between human and machine in this paper used PHP command line interface (CLI) or console mode. The driver to connect to the MySQL server used php_mysql. The system that is used to run the application should have installed PHP as an interpreter which will be executed line by line of code in the application. The application is shown in Figure 9.



```
Client Demo
+ type 'bye' to exit
________________________________________

Saya : halo
Tiara: Halo juga
Saya : assalamu'alaikum
Tiara: wa'alaikum salam
Saya : apa kabar TIARA?
Tiara: Kabar baik
Saya : masih ingat namaku?
Tiara: Namamu Anonim kan?
Saya : bukan
Tiara: Februari
Saya : aku tidak tanya bulan
Tiara: Silakan
Saya : sudahlah
Tiara: Oh ternyata
Saya : TIARA jelek
Tiara: Ngobrol yang lain saja ya
Saya : OK
```

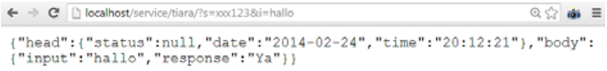
Figure 9. Chatbot Application using PHP Language

In Figure 9 shows that application could receive input containing a single quote (‘) because in the code has been completed with the escaping character. In the testing stage, it wasn't found a run time error.

A core chatbot was also implemented to build web service in order to be accessed by client applications written in the programming language. A web service in the need of testing was written using PHP and need web server of Apache to communicate via HTTP protocol between server and client application. The format data for web service was using javascript object notation (JSON). The structure of JSON document resulted by web service is shown bellow,

```
{
“head”: {“status”:[status],
”date”:[date], “time”:[time]},
“body”: {“input”:[input],
”response”:[response]}
}
```

The structure of JSON document consists of two main parts, i.e. head as metadata for data sent and body as data read. Head consists of information status functioned as sending status, date functioned as sending date, and time functioned as sending time in the format using hour:minute:second (h:m:s). Meanwhile, body consists of input functioned as input given by client and response functioned as response given by the chatbot. Testing of web service could be done from web browser with giving a parameter of s as id session and parameter of i as input sent to the chatbot. The Figure 10 shows testing has been done using the server in the localhost.



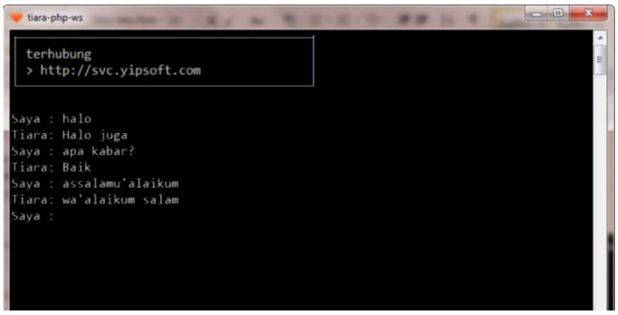
```
{"head":{"status":null,"date":"2014-02-24","time":"20:12:21"},"body":
{"input":"hallo","response":"Ya"}}
```

Figure 10. The JSON Document in The Web Browser

Based on Figure 10, the web service could be accessed on the address of http://localhost/service/tiara/?s=xxx123&i=hallo, thus the server will give the response as ‘Ya’ for input ‘hallo’. The client application that is used to communicate with web service written using PHP CLI in the console mode. This application applied operation of input/output (I/O) file in reading uniform resource locator (URL) for input sending and response receiving. Figure 11 shows client application when run to access the web service on site http://svc.yipsoft.com/tiara/?i=Hello. The parameter i on the URL is input from a user containing “Hello”. Thus web service will give response as JSON shown bellow,

```
{
“head”
{
 “status”: “OK”,
 ”date”: ”26-04-12”,
 “time”: “05-57-23”
},
“body”
{
 “input”: “Hello”,
 ”response”: “Hello juga..”
}
}
```

Client program does parsing to the JSON document as shown above to be displayed on the user interface.

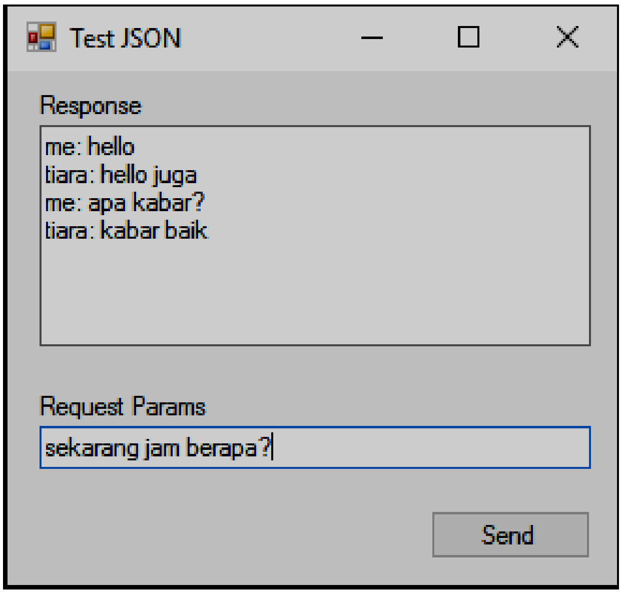


```
terhubung
> http://svc.yipsoft.com

Saya : halo
Tiara: Halo juga
Saya : apa kabar?
Tiara: Baik
Saya : assalamu'alaikum
Tiara: wa'alaikum salam
Saya :
```

Figure 11. Text-Based Client Application in PHP

The I/O process that is shown in Figure 11 depends on the internet access. Testing shows that web service could receive input containing single quote character (‘) because in the web service have existed mechanism of escaping character. This made all client applications written using many programming languages could send input without a mechanism of escaping character. Other client chat program based on graphical user interface (GUI) using .NET/C# is shown in Figure 12.



```
me: hello
tiara: hello juga
me: apa kabar?
tiara: kabar baik
```

Request Params: sekarang jam berapa?

Figure 12. GUI-Based Client Application in PHP

Client program only needs to send a request using GET method and receives a response in the format of JSON. The client then does parsing on JSON and displays in the user interface as shown in the function of GetFromWeb() bellow,

```
async private Task GetFromWeb()
{
    HttpClient client = new
HttpClient();
    Task<string> JSONData =

client.GetStringAsync("http://svc.yipsof
t.com/tiara/?i=" + msg);

    string response = await JSONData;
    dynamic answer =
JObject.Parse(response);

    Console.WriteLine(response);
    txtResponse.AppendText("tiara: " +
answer.body.response + "\n");
  }
```

V. Conclusions

All tables and stored program those have been designed, implemented, and tested are packaged in the database. The database is the core for chatbot application written using PHP programming languages. The PHP programming language has a driver in order to connect to the core in the database. Before application was made, it needed to be installed MySQL server in the development environment and made sure that to do the dumping process for the database.

VI. Future Works

There are some works should be done and have not been done in this paper, so this paper can be a reference to the next work. Some of those should be concerned, i.e.:

1. SQL code used in the database needs to be portable so it can result in more cores in the implementation in many RDBMS
2. It needs more stored programs for knowledge management
3. The value of the similarity is in the range between 0 up to 1 only, so it could be added more parameters in implementation of fuzzy logic to determine the precision of sentence

Acknowledgment

We thank STMIK AMIKOM Yogyakarta which has given us chance to publish our research paper

References

[1] B. Setiaji, and F. W. Wibowo, "Chatbot Using A Knowledge in Database," Proc. of 2016 IEEE 7th International Conference on Intelligent Systems, Modelling and Simulation, 2016.

[2] A. Augello, G. Pilato, A. Machi, and S. Gaglio, "An Approach to Enhance Chatbot Semantic Power and Maintainability: Experinces Within The FRASI Project," Proc. of 2012 IEEE Sixth International Conference on Semantic Computing, 2012, pp. 186-193, doi:10.1109/ICSC.2012.26.

[3] H. Al-Zubaide and A. A. Issa, "OntBot: Ontology Based Chatbot," Proc. IEEE of 2011 Fourth International Symposium on Innovation in Information & Communication Technology (ISIICT), 2011, pp. 7-12, doi:10.1109/ISIICT.2011.6149594.

[4] C. Erdogan, H. Nusret Bulus, and B. Diri, "Analyzing The Performance Differences Between Pattern Matching and Compressed Pattern Matching on Texts," Proc. IEEE of 2013 International Conference on Electronics, Computer and Computation (ICECCO), 2013, pp. 135-138, doi:10.1109/ICECCO.2013.6718247.

[5] J. P. McIntire, L. K. McIntire, and P. R. Havig, "Methods for Chatbot Detection in Distributed Text-Based Communications," Proc. IEEE of 2010 International Symposium on Collaborative Technologies and Systems (CTS), 2010, pp. 463-472, doi:10.1109/CTS.2010.5478478.

[6] Y. Wu, G. Wang, W. Li, and Z. Li, "Automatic Chatbot Knowledge Acquisition from Online Forum via Rough Set and Ensemble Learning," Proc. IEEE of 2008 IFIP International Conference on Network and Parallel Computing, 2008, pp. 242-246, doi:10.1109/NPC.2008.24.

[7] S. Ghose and J. J. Barua, "Toward The Implementation of A Topic Specific Dialogue Based Natural Language Chatbot As An Undergraduate Advisor," Proc. IEEE of 2013 International Conference on Informatics, Electronics & Vision (ICIEV), 2013, pp. 1-5, doi:10.1109/ICIEV.2013.6572650.

[8] A. Augello, M. Scriminaci, S. Gaglio, and G. Pilato, "A Modular Framework for Versatile Conversational Agent Building," Proc. IEEE of 2011 International Conference on Complex, Intelligent and Software Intensive Systems (CISIS), 2011, pp. 577-582, doi:10.1109/CISIS.2011.95.

[9] G. Pilato, A. Augello, and S. Gaglio, "A Modular Architecture for Adaptive Chatbots," Proc. IEEE of 2011 Fifth IEEE International Conference on Semantic Computing (ICSC), 2011, pp. 177-180, doi:10.1109/ICSC.2011.68.

[10] G. Neubig, "NLP Programming Tutorial 2 – Bigram Language Models," Presentation Module of Nara Institute of Science and Technology (NAIST).

[11] M. Dickinson, "Smoothing," Presentation Module of Dept. of Linguistics, Indiana University, Fall 2009.

[12] Y. Bin, P. Cunlin, and L. Dan, "Chinese Text Feature Extraction Method Based on Bigram," Proc. IEEE of 2013 International Communications, Circuits and Systems (ICCCAS), 2013, pp. 342-346, doi: 10.1109/ICCCAS.2013.6765352