UGLEO: A WEB BASED INTELLIGENCE CHATBOT FOR STUDENT ADMISSION PORTAL USING MEGAHAL STYLE

[1]*Anneke Annassia Putri Siswadi*, [2]*Avinanta Tarigan*
[1,2]*Management Information System, Master Degree Program Gunadarma University*
Jl. Margonda Raya No. 100, Pondok Cina, Depok 16424, Indonesia
[1]*anneke_ps@student.gunadarma.ac.id*, [2]*avinanta@staff.gunadarma.ac.id*

Abstract

To fulfill the prospective student's information need about student admission, Gunadarma University has already many kinds of services which are time limited, such as website, book, registration place, Media Information Center, and Question Answering's website (UG-Pedia). It needs a service that can serve them anytime and anywhere. Therefore, this research is developing the UGLeo as a web based QA intelligence chatbot application for Gunadarma University's student admission portal. UGLeo is developed by MegaHal style which implements the Markov Chain method. In this research, there are some modifications in MegaHal style, those modifications are the structure of natural language processing and the structure of database. The accuracy of UGLeo reply is 65%. However, to increase the accuracy there are some improvements to be applied in UGLeo system, both improvement in natural language processing and improvement in MegaHal style.

Keywords: *Intelligence chatbot, question answering, MegaHal, Markov Chain.*

INTRODUCTION

Gunadarma University is one of universities in Indonesia. To fulfil the prospective student's information need, Gunadarma University already has many services, such as Gunadarma University's website, Gunadarma University's book, registration place, Media Information Center, and Question Answering's website (UG-Pedia). The services that offer the user to ask the question and get the answer in real time are registration place and media information center, but those services are limited by the working hours.

The number of people who looking for the same information about a college encourages the Question Answering (QA) service is created. Question answering systems are developed to accept user's questions in natural language, and retrieve answers from question-answer databases. The goal of the question answering system is to retrieve the answers to questions rather than full documents or even best-matching passages as most information retrieval systems currently do [1][2]. However, Question Answering system could also give a direct answer, if only one document matched the query. The retrieving process for this is not that simple, as these systems use sophisticated language processing to analyse the user input and retrieve answers by applying grammar and semantic parsers. As mentioned in [3] that

providing a QA system with a dialogue interface would encourage and accommodate the submission of multiple related questions and handle the user's requests for clarification, and chatbot can be used for this system.

Computers need some sort of interaction in order to perform a specific goal or task. Natural language is one of many interface styles that can be used in the dialogue between a human user and a computer through the use of speech or text [4]. Chatbot is a technology that makes interaction between human and machine using natural language possible [5]. A chatbot is a type of conversational agent, i.e., a computer program designed to simulate an intelligent conversation. It processes users' inputs in natural language and it looks up in its knowledge base to return an answer that imitates the human[6]. Chatbots are available online, and are used for different purposes, such as MIA, a German-language advisor on opening a bank account and Sanelma, a guide to talk with in a museum who provides information related to specific pieces of art [2].

Loebner Prize Competition is an annual competition for conversational agents. It is the first formal instantiation of a Turing Test [7]. Based on [8], the technical approaches and algorithms that are used in chatbot development are pattern matching, parsing, markov chain models, ontologies, AIML, and Chatscript. Among all the methods, markov chain models is one of method that implements machine learning theory which gives the chatbot possibility to predict the answer of a question, and the chatbot that implements this model is called MegaHal [9].

UG-Pedia is a question answering's website that give an answer based on question-answer system while media information centre and registration place that answer the prospective student's question in direct dialogue with human. It needs the system that is combining those system, the system that give an answer based on question-answer system in dialogue interface with machine learning implementation. The system that can make user seems talking with human. It can be able to be implemented by chatbot using MegaHal, since the chatbot can retreive the question in natural language form and MegaHal implements the machine learning method. The problems discussed in this thesis are:

1. How to adapt Indonesian language into MegaHal?
2. What kind of database that needed in the chatbot?
3. How to make an application that can retreive a question in natural language and predict the answer due to MegaHal result?

The aim of this research is to develop an application in dialogue interface that can retreive prospective student's questions about Gunadarma University admission in natural language and giving the best prediction information as an answer.

Artificial Intelligence

Artificial intelligence definitions can be organized into four categories, thinking humanly, thinking rationally, acting humanly, and acting rationally [10]. Thinking humanly defines artificial intelligence as thinking humanly. It means the program is developed to think like a human with observing how human thinks, how human's brain reacts (the cognitive modelling approach). Acting humanly is done with the turing test approach. The Turing Test was proposed by Alan Turing (1950). It works to test a computer if human interrogator, after posing some written questions, cannot tell whether that written responses are posed from a computer or a person. Thinking rationally defines artificial intelligence by the laws of thought approach. The Greek philosopher Aristotle provided patterns for argument structures that always yielded correct conclusions when given correct premises. All kinds of objects in the world is developed into notation for statement and all problems is described in logical notation and solved it with logics tradition. Acting rationally defines artificial intelligence with the rational agent approach. Computer agents are expected to do more: operate autonomously, perceive their environment, persist over a prolonged time period, adapt to change, and create and pursue goals. This approach has the same point with thinking rationally, logic, although there is also has the different thing. Thinking rationally solve the problem with logicist tradition but correct inference is not all rationally.

Artificial intelligence can be classified into two major types [10], those are weak AI and Strong AI. Weak AI is the thinking dedicated towards the development of technology proficient of carrying out pre-planned moves based on. Chess applications and Google robot car are weak AI example since those application is not really thinking but simulated thinking. As contrasted to that, Strong AI not just mimicking human demeanor in a certain province is developing technology that can think and function similar to humans. However, most people argue that strong AI will never be developed, at least need a long time.

Machine Learning

Machine learning is one of artificial intelligence branch. Machine learning is a system that can take known data as input, learn from the known data, and classify or draw conclusions from unseen data. It focuses on prediction based on known properties learned from data while data mining focuses on the discovery of previously unknown properties on the data. Machine learning classifies into two main types, supervised learning and unsupervised learning [10].

The machine learns with an instructor. It is learning from some known data and handle it to classify unknown data. The methods of supervised learning are decision tree, oneR, Lazy, Naive Bayes, Markov model, Hidden Markov model, Linear

Regression, Hyperplane, Artificial Neural Network, and Support Vector Machine (SVM).

The machine learns without an instructor. It is learning by trying something and see how it works. This machine needs utility function to calculate how well it worked. Reinforcement learning is an unsupervised learning method. It makes the machine interacts with its environment by producing actions then these actions affect the state of the environment which is turn results in the machine receiving some scalar rewards. The goal of reinforcement learning is to make the machine learns to act in a way that maximizes the future rewards it receives (or minimizes the punishments) over its lifetime [11]. Reinforcement Learning is divided into two types based on the goal of utility function, passive reinforcement learning and active reinforcement learning. It also has three types of reinforcemet learning agent, those are Utility-Based Agent learns a utility function on stales and uses it to select actions that maximize the expected outcome utility, Q-Learning Agent learns an action-utility function, or Q-function, giving the expected utility of taking a given action in a given state, and Reflex Agent learns a policy that maps directly from states to actions[10].

MegaHal

The Loebner Prize for artificial intelligence (AI) is the first formal instantiation of a Turing Test. The Loebner Prize is an annual event which cash prize and a bronze medal to the most human-like computer [7]. This event was held firstly on 8th of November 1991 in Boston's Computer Museum. In 1996, the primary author entered the Loebner contest with an ELIZA variant named HeX and in 1997 the more powerful program is entered, named SEPO. In that year, MegaHal chatbot was entered with a significantly different method of simulating conversation either HeX or SEPO. MegaHAL is able to construct a model of language based on the evidence it encounters while conversing with the user. How MegaHal works can be seen in Figure 1.

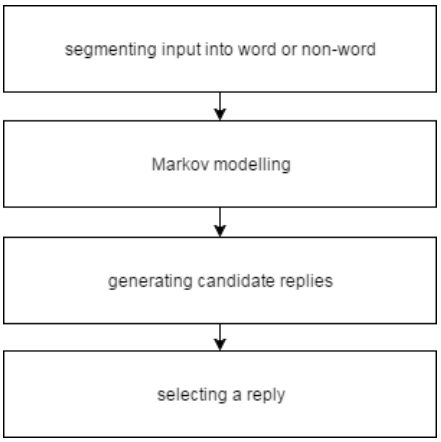

Figure 1. MegaHal works

Natural Language Processing

Natural Language Processing (NLP) is the computerized approach to analyzing text that is based on both a set of theories and a set of technologies [12]. NLP began in the 1950s as the intersection of artificial intelligence and linguistics [13]. Traditionally, work in natural language processing has tended to view the process of language analysis as being decomposable into a number of stages, mirroring the theoretical linguistic distinctions drawn between syntax, semantics, and pragmatics [14].

Chatbot

A chatbot is a conversational software agent, which interacts with users using natural language [15]. Kerly in 2007 described chatbots as "conversational agents, providing natural language interfaces to their users". In this way they are well-suited for use as the interactive layer in a question-answering system designed with dialogue in mind [7]. The purpose of a chatbot system is to simulate a human conversation; the chatbot architecture integrates a language model and computational algorithms to emulate informal chat communication between a human user and a computer using natural language [16]. There are some following issues required to develop a chatbot system: computer-based of natural languages processing, define and design knowledge base for the chatbot, and develop suitable algorithms for pattern matching. Loebner prize is a competition that methodologically compares chatbot technologies, rates them in a conversational sense and thus gives some sort of a general feedback over the used technologies. Due to the Loebner Prize, there are six technical approaches and algorithms [8]:

1. Pattern Matching

   This algorithm is the most common approach and technique used in Chatbots. The simplest patterns were used in earlier chatbots such as ELIZA and PCTherapist.

2. Parsing

   Textual Parsing is a method which takes the original text and converts it into a set of words (lexical parsing) with features, mostly to determine its grammatical structure.

3. Markov Chain Models

   The Idea behind Markov Chain Models is that each occurrence of a letter or a word in some textual dataset occurs with a fixed probability.

4. Ontologies (Semantic Nets)

   Ontology or semantic network as it is called in some chatbot systems is a set of hierarchically and relationally interconnected concepts.

5. AIML

   AIML's syntax is XML based and consists mostly of input rules (categories) with appropriate output.

6. ChatsSript

   ChatScript is successor of the AIML language. It focuses on the better syntax which makes it easier to maintain.

RESEARCH METHODOLOGY

Identify The Problem

The UGLeo is a question-answering web-based application in dialogue interface. This application focuses on helping the Indonesian prospective students for gathering information about Gunadarma University and the other information. The UGLeo system is the only one who interact with user, so the UGLeo chatbot must has the ability to retreive the question in natural language.

Determine the Chatbot's Method

AIML is the popular appropriate approach for building the chatbot. AIML represents the knowledge base in a graphmaster and uses the depth first for searching technique [2]. However, ALICE style is not suitable with this research's goal. The other machine learning method for developing the chatbot is Markov Chain. Both graphmaster and Markov chain are using decision tree form. The differences are graphmaster is only using the depth first searching technique for determining the reply based on its pattern, while determining the reply in Markov chain is based on the calculation of node's probabilities. It might be useful for selecting the node's reply when there are more than one node that rooted in one root node. Hence, the method used in developing the chatbot in this research is Markov chain.

Determine the Chatbot's Package

MegaHal is a chatbot which is using Markov Chain method to build. The MegaHal used in developing the application is JMegaHal which is MegaHal package in java programming language. JMegaHal package is actually already provided in many official sites but this research needs not only using the package but also modifying the code in the package. Since those pack-ages do not allow to do it, the JMegaHal package which is used in developing the chatbot is the package that developed by personal software engineering.

Analysis

1. Software System Analysis

   This research uses Megahal style which implements Markov modelling for guessing the answer for each statement that user typed. Since the target of this application is Indonesian prospective students, UGLeo application development needs to make this application adapts with Indonesian language.

2. Data Analysis

   The knowledge for UGLeo chatbot is about the Gunadarma University's global information and the information which usually asked by the Gunadarma University's prospective student. The name of chat-bot's knowledge is 'tb_kb'. This chatbot also needs the data support for doing the natural language processing (normalization, stemming, and swapping), like table normalization which contains the

informal word and its formal word, and table of swapping which contains the general acronym and abbreviation and its standing for. The name of them are 'tb_norm' and 'tb_swap'. The data needed for stemming processing is the list of root words. These data is gotten from the Indonesian dictionary (KBBI). The name of this table is 'tb_word'.

3. Software and Hardware Analysis
   The UGLeo chatbot application develop-pment is built with Java programming language for web-based application and MySQL for local database.

Designing

The designing step consists of four sections, those are UGLeo architecture, software system design, data design, and application design.

Implementation

The implementation step is showing how the system design implemented into source code and the screenshot about how the program executed.

Testing

Testing used in the UGLeo chatbot application is an accuracy testing. This test aim is finding out how accurate the information that system given to the user. The target of this test is the second grade or third grade students in senior high school. They have to ask a question about the given topic and select one category of accuracy information as their opinion about the program result. The number of students who do this test is 5. They have to ask 4 questions with different topics. The question topics are prospective student admission, the major, Gunadarma University's contact information, Gunadarma University's profile.

RESULTS AND DISCUSSION

Architecture Design

The architecture design of UGLeo chatbot application is divided into UGLeo system architecture and UGLeo chatbot arcitecture. The UGLeo system architecture can be seen in Figure 2.

The UGLeo system architecture describes the interaction between client and server in UGLeo application. The request and response are handled in JavaServer Page because JSP is the interface between human and system in HTML form. This JSP then send the request to the servlet as a connector to retrieve request from JSP and send the response to the JSP again. To do the answer prediction, the server must have connection to the UGLeo library which needs to get data from database. Another architecture is UGLeo chatbot architecture. This architecture is figured in Figure 3.

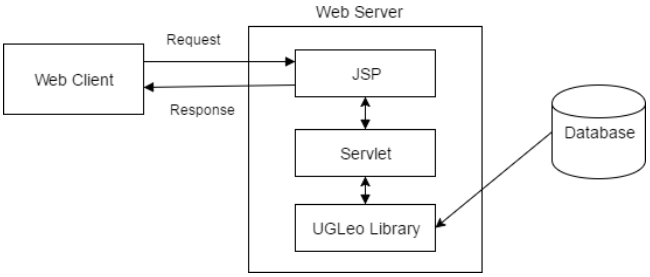

Figure 2. UGLeo System Architecture

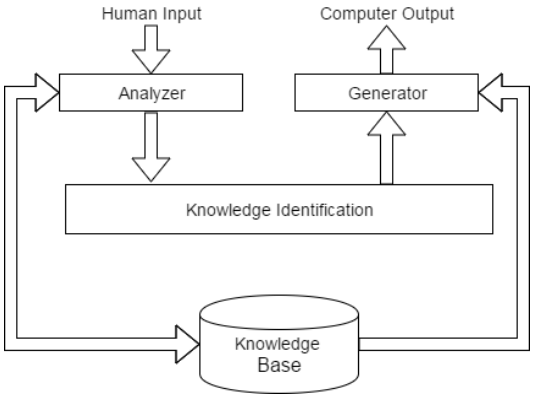

Figure 3. UGLeo Chatbot Architecture

Analyzer

The main process of analyzer processing is looking for words in the input sentence then creating symbols of the sentence. The output of analyzer processing is a sequence of symbols from the input sentence. In this process, chatbot retrieves the input and do the first main process in Megahal style, split the input sentence into word or non-word. As seen in Figure 3., there are two flow processes in analyzer.

First, chatbot retrieves the input from user and split the user input sentence into word or non-word. Second, chatbot loads knowledge from knowledge base and split each of them into word or non-word.

The analyzer process is described in Figure 4. The output of splitting word and non-word are a sequence of words and a sequence of non-words. Words are alpha-numeric characters while non-words are the other characters. Each word is checked whether the word need to do swapping or not. Swap processing is a process that checking if there is any general abbreviation or acronym word, then change them into its stand for. For example, the sentence of "Dimana pendaftaran maba?" will get the result "Dimana pendaftaran mahasiswa baru?". The swapping word usually has more than one stands for words. The list of general abbreviations and acronyms are listed in table tb_swap. Those general abbreviations and acronyms are classified into non word

category in type of word.

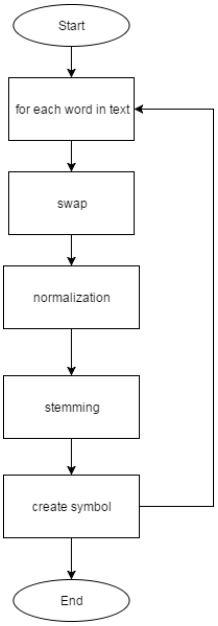

Figure 4. Analyzer Process

Normalization is a process for checking whether there is any non-formal word. This process then changes it into its formal word, such as ‘akun’ for ‘akuntansi’ and ‘gundar’ for ‘gunadarma’. The example of analyzer process is shown in Figure 5. Since there is no word needed to be normalized, the result of normalization process of knowledge has the same sentences with itself. Stemming is a process for finding the root word, if the current word is already root word, the result is still that current word, and if the current word is word with affix, the result is its root word. This process works by Stemming Porter algorithm and uses Kamus Besar Bahasa Indonesia (KBBI) for root word database.

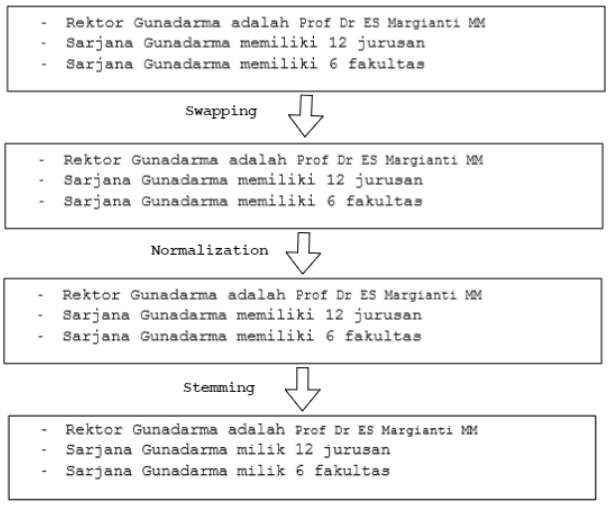

Figure 5. Example of Analyzer Process

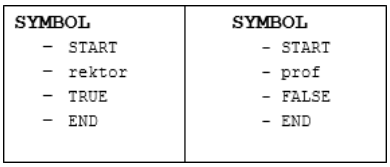

Figure 6. Example of Symbol

The next process is checking if the current word is not stopword and the current word is word (aphabet). Stopwords are natural language words which have very little meaning [11]. Due to [7], stopwords consist of determiners, coordinating conjunctions, and prepositions. Stopwords used in this research are the stopwords written in [17], lists of determiners, conjunctions, prepositions in Indonesian language, and the common words. In splitting process, the output of stemming has to enter the keyword checker (the not stopword and the word processing). It continues to the next process, creating symbol. Symbol is a new struct for each word. This struct consists of start identifier, the current word, its keyword's value, and end identifier. Figure 6 shows the examples of symbol for rektor symbol and prof symbol.

Knowledge Identification

The knowledge identification process implements three main processing of Megahal, Markov modelling, generating candidate reply, and selecting reply. The main process of knowledge identification is described in Figure 7. This main process is divided into four steps, train into Markov model which implements Markov modelling, generate candidate reply which implements generate candidate reply, the last is calculates information of each candidate reply and determine the list of symbol reply which implements selecting reply.

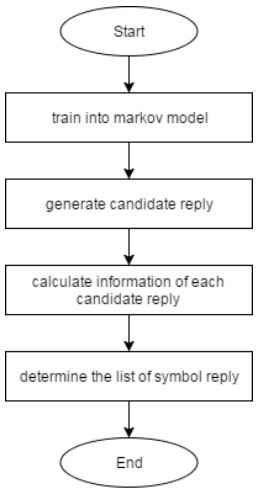

Figure 7. Knowledge Identification Process

The first thing to do when user input's symbols and knowledge base's symbols retrieved is training those symbols into Markov models. The UGLeo application builds Markov model for each symbols of knowledge base' words which have been created to be symbols. Those knowledge base' symbols and user in-put's symbols are trained into two kinds of Markov model, forward model and backward model. The forward model is used for predict which symbol will following any sequence of four symbols while the backward model is used for predict which symbol will precede any such sequence. The first sequence trained into Markov model is knowledge base' symbols. Then, user input is trained into the previous Markov model and used for determining the candidate reply.

The program implements Markov model building by tracking the children in every node. Markov model's nodes in this program implementation is assumed by the symbols. In this program implementation, node is built in TrieNode struct. TrieNode struct contains of node, child, usage, and count. Usage is the number of times node's context occurs while count is the total of the children's usages.

When both forward model and backward model have been built, the next process is generating the candidate reply. The candidate reply generated by generating the symbols randomly. It happens in some period of time, 5 seconds. There are two different ways to get the candidate reply. The first way is selecting the userKeyword if symbols is empty and userKeyword is not empty. Symbols is the list of symbols that is generated when process happens in the second time or more, and userKeyword is the list of the symbols' sequence output from analyzer process which have the true value of keyword's attribute symbol. Another way is passed through when both the symbols and userKeyword are not empty. In this condition, it will find the longest context in trie (backward or forward). Then, the userKeyword index selects randomly and get the child of that index gotten (subnode). If the subnode is the userKeyword, that subnode is selected being a member of candidate reply, and if the subnode is not the userKeyword, it will get the node's child for the previous index. It occurs until all nodes has been checked.

The candidate reply selection iterates as many as possible in 5 seconds. One iteration produces a list of candidates reply. Each candidate reply must have the information calculation since the candidate reply is selected by generated randomly. The information value is the total of previous information value with the calculateResult operation.

The calculateResult operation implements the equation below for calculating the quality of candidate reply's members. The last process in this calculation of information is scale the information.

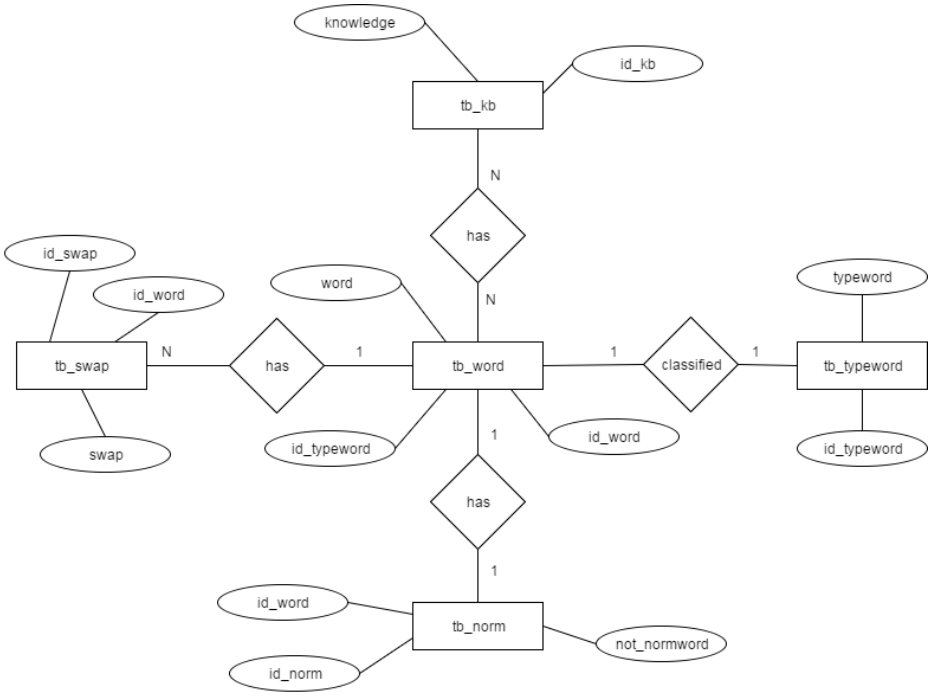

Figure 8. ERD of UGLeo Application

$$(w|s) = -log_2(w|s)$$

To select the reply, it must choose the highest information of each candidate replies. If the information value is higher than the previous value and candidate reply is not fully the same with userKeyword, that candidate reply is selected to be the reply. The next process after knowledge identification is generator. The task of the generator is generating the sentence for being showed to user. When the selected reply is not null, each member in selected reply's list will be joined into a string. Since the symbols in Markov models are full symbols (include not keyword symbol), and the question words like 'apa', 'siapa', 'kapan', 'bagaimana', and 'di mana' are also included, so the symbols in reply list which are joined into string are all symbols except those question words. This string joined is shown to the user as a reply from the system.

Database Design

The data needed in building UGLeo is modeled by ERD. The data diagram is shown in Figure 8. Due to ERD of UGLeo application as seen in Figure 8, UGLeo database contains five different tables, tb_kb for knowledge base table, tb_word for all words table, tb_typeword for type of word table, tb_norm for word normalization table, and tb_swap for word swap table.

.

Figure 9. Chat Page

Implementation

Figure 9 shows the main page and the only one page in UGLeo application. Before the system do the next process, it has to check whether all dataare loaded successfully.

Swap is the number of data which are used in swap processing (nonword) while norm is the number of data which are used in normalization processing. There are 36 data listed in table tb_swap and 28 data listed in table tb_norm. On the other hand, ban is the number of word data which are banword (stopword) while aux is the number of data which are auxword(rootword).

There are 779 data listed in table tb_word for type words 1 and 28252 data listed in table tb_word for the others type words. The next process is loading the knowledge base. Knowledge data is done separately because each data in knowledge base must be trained into Markov Models while the others are not.

Analyzer process is a process for splitting a sentence to be a sequence of words and creating symbols of them. This process happens for splitting each sentence in knowledge and user input. The sentence that shown in Figure 10 is “Prof. Dr. E. S. Margianti, SE., MM.”.

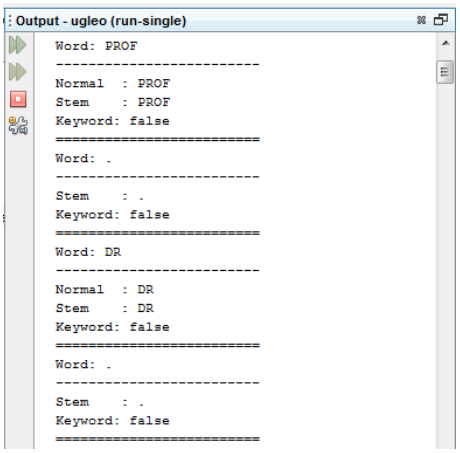

Figure 10. Splitting Process

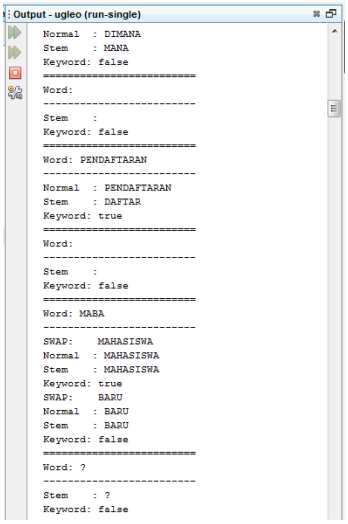

Figure 11. Analyzer Output

Figure 11 shows the stemming process in 'pendaftaran' word. That word is a word 'pe' prefixed and 'an' suffixed. The root word of that word is 'daftar'. The other words in Figure 11 have the same word for output and input. The last prediction answer processing (generator) is generating the reply for the user, so the affix removed word has to be built again into the first one (word with affix). For example, the 'daftar' word has to be built again into 'pendaftaran' word.

First task to do in knowledge identify-cation is training all symbols into markov models. The finishing of Markov models' training is marked by the sentence about the number of knowledges that are trained. Before the system starts to do the reply prediction, the system has to receive the input question from the user. The text input is 'dimana pendafataran maba?' and the analyzer result for this text is shown in Figure 12.

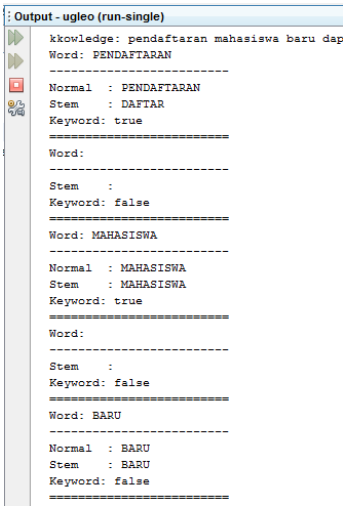

Figure 12. Analyzer Output for User Input

The process in analyzer is swapping, normalizing, and stemming. In user input, there is a word that is needed to be swapped, it is 'maba'. Maba word stands for 'mahasiswa baru', so the word 'maba' is swapped into 'mahasiswa' and 'baru'. The process continues into knowledge identification process then generating candidate reply. Candidate reply is generated by finding the longest chain and looking for the symbols in that chain which have the same word as user keywords input. The output of these processes is shown in Figure 13. The best prediction reply is the candidate reply which has the highest number of information value. The candidate replies selected then generated into String as a reply.

Testing

The objective of this testing is to measure the accuracy of the UGLeo's reply, how accurate the information which UGLeo gives to the user as a reply. The testing result is summarized in Table 1.

Based on the testing result, the system's reply depends on how many information in knowledge base that has the same keyword. There are more chains when there are more information. It makes the system generates more candidate replies. This condition gives the probability for predicting the wrong answer or not exactly right answer. In brief, MegaHal style is not really good way to develop a question answering chatbot, the reasons are:

1. It generates the candidate reply only based on the mathematical logic. It causes there is candidate reply which is generated meaningless.
2. The stop mark is not applied in Markov chain, so the system generates the candidate reply ends in the longest chain's stop.
3. It grows its Markov chain for one execution. The growth is deleted when the execution ends.

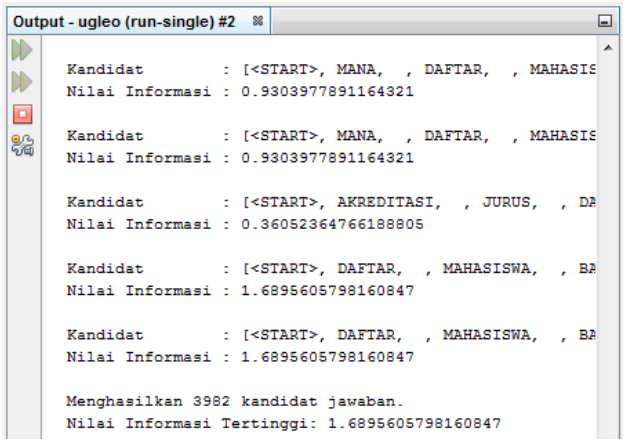

Figure 13. Candidate Reply and Information Value

Table 1. Testing Result

No.	Topic	Right Answer	Wrong Answer
1.	Prospective student admission	1	4
2.	The major	3	2
3.	Gunadarma University's contact information	4	1
4.	Gunadarma University's profile	5	0

CONCLUSION AND SUGGESTION

UGLeo is a web based intelligence chatbot for student admission portal. This chatbot is developed using MegaHal style which implements the Markov Chain method. UGLeo is able to predict and generate the answer of a question about prospective student information. The accuracy of UGLeo's reply is 65% from 20 questions. So, the chatbot development using MegaHal style for Question-Answering system is good enough, since the accuracy is more than 50%. However, it needs many improvements with this style to make a better chatbot with high accuracy.

Better result will be achieved by develop this application if weight princi-ple is added to calculate the answer's quality, gives the synonim principle in analyzer process, implements the stop mark for each last symbol in each sentence, grows the Markov chain every time, and also gives more knowledge to the chatbot.

BIBLIOGRAPHY

[1] B. A. Shawar and E. Atwell, "A chatbot as a question answering tool", In International Conference on Advances in Software, Control and Mechanical Engineering, 2015.

[2] B. A. Shawar, "A Corpus Based Approach to Generalising a Chatbot System". PhD thesis, University of Leeds School of Computing, 2005.

[3] S. Quarteroni and S. Manandhar, "A chatbot-based interactive question answering system" , In 11th Workshop on the S, 2007.

[4] G. R. Sankar, J. Greyling, and D. Vogts, "Towards a conversational agent for contact centres", In SATNAC, 2008.

[5] A. S. Lokman and J. M. Zain, "One-match and all-match categories for keywords matching in chatbot" American, Journal of Applied Sciences 7, pp. 1406– 1411, 2010.

[6] F. A. Mikic, J. C. Burguillo, A. Peleteiro, and M. Rey-Lopez, "Using tags in an aiml-based chatterbot to improve its knowledge", Computer Science, pp. 123– 133, 2012.

[7] L. Prize, "What is the loebner prize?", 1995. [Online]. Accessed on June 2016.Available:

http://www.loebner.net/Prizef/loebner-prize.html.

[8] L. Bradesko and D. Mladenic, "A survey of chatbot systems through a loebner prize competition", Research Gate, 2012. [Online]. Accessed on January 2016. Available: https://www.researchgate.net/profile/Luka_Bradesko/publication/235664166_A_Survey_of_Chatbot_Systems_through_a_Loebner_Prize_Competition/links/09e41512679b504a17000000.pdf?origin=publication_detail.

[9] J. L. Hutchens and M. D. Alder, M. D, "Introducing megahal", ACL Home Association for Computational Linguistics, 1993. [Online]. Accessed on January 2016. Available:http://www.csee.umbc.edu/courses/471/papers/introducing-megahal.pdf

[10] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern*, 3rd Edition. Pearson Education Limited, 2010.

[11] Z. Ghahramani, *Unsupervised Learning*, University College Lon-don, UK, Gatsby Computational Neuroscience Unit, 2004.

[12] E. D. Liddy, *Encyclopedia of Library and Information Science*, 2nd Edition, chapter Natural Language Processing. Marcel Decker, Inc, 2001.

[13] P. M. Nadkarni and L. Ohno-Machado, and W. W. Chapman, *Natural language processing: an introduction.* J Am Med Inform Assoc, 2011.

[14] N. Indurkhya and F. J. Damerau, *Handbook of Natural Language Processing*, 2nd Edition. Chapman & Hall, 2010.

[15] B. A. Shawar, A chatbot as a natural web interface to Arabic web qa," iJET, 2011.

[16] B. A. Shawar and E. Atwell, "Chatbots: Are they really useful?" LDV-Forum, 2007.

[17] D. Nopiyanti and K. Sekarwati, "Aplikasi pencarian kata dasar dokumen berbahasa indonesia dengan metode stemming porter menggunakan php dan mysql", In Prosiding Seminar Ilmiah Nasional Komputer dan Sistem Intelijen, volume 8: KOMMIT, 2014.