

Classification Technique of Interviewer-Bot Result using Naïve Bayes and Phrase Reinforcement Algorithms

<https://doi.org/10.3991/ijet.v13i02.7173>

Moehammad Sarosa^(✉), Mochammad Junus, Mariana Ulfah Hoesny
State Polytechnic of Malang, East Java, Indonesia
msarosa@polinema.ac.id

Zamah Sari
Universitas Muhammadiyah Malang, East Java, Indonesia

Martin Fatnuriyah
Myria Technologies, Malang, East Java, Indonesia

Abstract—In recent years, both foreign and national companies tend to conduct English-based interviews when recruiting new employees. Consequently, college graduate must be ready for English-based interviews during the process of seeking employment. To meet these requirements potential candidates tend to practice conversing in English with someone who is proficient in the language. Nevertheless, it is not easy to have someone who is not only proficient in English, but also have a good understanding of common interview questions. This paper presents the development of a machine which is able to provide practice on English-based interviews, specifically on job interviews. Interviewer machine (interviewer bot) is expected to help students practice speaking English appropriately for job interview. The interviewer machine design uses words from a chat bot database named ALICE to mimic human intelligence that can be applied to a search engine using AIML. Naïve Bayes algorithm is used to classify the interview results into three categories: POTENTIAL, TALENT and INTEREST students. Furthermore, based on the classification result, the summary is made at the end of the interview session by using phrase reinforcement algorithms. By using this bot, students are expected to practice their listening and speaking skills, also to be familiar with the questions often asked in job interviews so that they can prepare the proper answers. In addition, the bot users could know their potential, talent and prospects in finding a job. Hence, they could apply to the appropriate companies. Based on the validation results of 50 respondents, the accuracy degree of interviewer chat-bot (interviewer engine) response obtained 86.93%.

Keywords—Job interview; Words classification; Natural language processing; Phrase reinforcement algorithm; Naïve Bayes algorithm.

1 Introduction

Competing in the global economy has given companies the need to recruit employees who are proficient in English. However, many of the prospective employees are facing difficulties in meeting the required English skills of the companies. Lack of facilities that support the students practising English in the educational environment will greatly affect the number of students who have the ability to communicate in English [1]. To date, the students' English skills, especially for a job interview, were only trained traditionally in English courses or English clubs which are managed independently by the students. There has not been a specialized training aimed at improving and measuring students' ability to communicate in an English-based job interview. Thus, a breakthrough software program that could significantly improve students' ability to speak English is highly needed, particularly those related to a job interview, aiming at increasing the opportunity of passing the job interview [2]. The development of technology and human lifestyle allows people to conduct online conversations (chat) in English through the internet. Communication could be in the form of text (text chat) or voice (voice chat). Users could send messages using text or voice to the online receiver, then the receiver could also reply with text or voice [3]. Artificial intelligence is created and inserted into a machine (computer) so that the machine could perform the task of conversing similar to humans. One of the artificial intelligence areas is Natural Language Processing (NLP) which enables the computer/answering machine the ability to read and understand the language used by humans. The NLP system requires computational linguistics ability to process the input resulted in an output that can understand human language as natural as possible. One of the machines that was created using this system is ELIZA, which is developed into today's chatterbot named ALICE (Alicebot) by Wallace. The AIML programming language is used in the first generation of Alice bot as a basic implementation of the system [4] [5] [6] [7] [8] [9] [10]. Artificial intelligence in the ALICE system used by the interviewer-bot to create the basic ability to interact with humans. The ALICE system was then integrated with Naive Bayes method that serves to classify the responses given by the users into groups of assessors with the assessment results. Then the classification results were processed using Phrase Reinforcement Algorithm to obtain conclusions of each assessor group [11] [12] [13].

Motivated from above researches, this paper presents the development of a machine which is able to provide practice on English-based interviews, specifically on job interviews. In summary, the contributions of this work are described as follows:

- a) We propose an alternative solution to the student's lack of ability in an English-based job interview.
- b) We build interviewer-bot application using PHP framework. This application will make a positive contribution and interactive services to enhance the students' potency, especially the ability to answer questions during the job interview in English [14] [15].

2 Related Works

Intelligent Tutoring Systems are computer programs aiming to provide personalized instruction to students. One of the mostly used is chatter-bot, a machine which provides conversational practice for users. The most popular chatter-bot on the internet, ALICE (Artificial Linguistic Internet Computer Entity) is written in AIML (Artificial Intelligence Markup Language), an open XML language. Burguillo, *et al.* in [16], the authors use of AIML-based bots for tutoring purposes in open e-Learning platforms i.e. Claroline or Moodle as the basic idea of the research. Burguillo, *et al.* in [16] developed two different user-friendly bots that already integrated in Claroline and Moodle, aim at not only to help students learn, but also support the lecturers on their teaching. A tutor bot (T-Bot), which is developed for students, could analyse the requests in written text and provide suitable answers about the course contents. On the other hand, evaluation bot (Q-Bot), developed for the professors, used questionnaires to track and supervise each student's learning progress.

As stated before, chatter-bots are human-like conversational-based software programs. It could be used as a conversation partner in specific knowledge domain depending on the software designer. The AIML, Artificial Intelligence Markup Language, is a XML derived language mostly-used to build chatter-bot knowledge bases on a case-based reasoning and textual pattern matching algorithms. In [17], a novel algorithm is implemented on an Italian-based chatter-bot to automatically generate AIML knowledge bases to answer the frequently asked question (FAQ) text file and a glossary of terms. This chatter-bot could be applied in e-learning to assist users navigate the learning system in the form of speaking avatar, such as in a distance learning session. The students could ask the digital assistant, in this case the chatter-bot, questions about the learning materials on the form of text based question and answer system.

Access to an institution's or company's information system is generally performed by manual navigation on the website menu and content built by the institution or company itself. Along with the development of technology and the company's desire to expand their services to customers, information can also be obtained through a chat between user and a virtual customer service. This study aims to build chatter-bot which serves as a virtual customer service that provides information to users with access to database query and website content. The research resulted on a chatter-bot prototype built using the O program, which is AIML interpreter based on PHP programming language and MySQL database. Artificial intelligence (in this case, the ALICE system) used by chatter-bot to form its basic ability to interact with humans. The advantage of this chatter-bot is the speed to access information; because data are sent and received by users in form of a text. Additionally, since its data searching methods; crawler (get an index or a link on a web page) and grabber (obtain data or information on an index or link) are not dependent on the development method used in an information system, chatter-bot can be easily integrated with any web information system. This chatter-bot had been tested in terms of verification, validation, and prototype testing. From the results, we concluded that this chatter-bot prototype was working properly, in accordance with the planning purpose, and reached the user satisfaction level required by the Turing Test. Chatter-bot system that

has the artificial intelligence ability could positively support the website as a form of customer service to improve user satisfaction of getting the required information either from the information system website or its database [18] [19].

Massive Online Open Courses (MOOCs), introduced in 2008 become one of the reason of the conversational bot development. Conversational bot solves one of MOOCs' drawbacks as it could replace the interaction between the students and the real instructor. Lim and Goh in [19] developed a MOOC-bot which is a prototype of MOOCs conversational bot and integrated it into MOOCs website to respond to the students' questions using text or speech input. MOOC-bot used AIML, took advantage of its ability to create suitable answer and easily adopt in the new domains. MOOC-bot system architecture consists of knowledge base equipped with AIML interpreter, chat interface, MOOCs website and Web Speech API to provide speech recognition and synthesis capability. The initial MOOC-bot prototype has the general knowledge from its predecessor – ALICE, such as frequent asked questions and a content implemented by Universiti Teknikal Malaysia Melaka (UTeM). Aside of the basic ones, it could be used at the same time by multiple sites, serves 24/7 in different time zones, and has multiple knowledge domains. MOOC-bot evaluation conducted based on the competition questions from Chatterbox Challenge (CBC) and Loebner Prize. The result showed that it was able to provide correct answers and had the capability to prolong the conversation.

3 Rudimentary

3.1 Naïve Bayes Algorithm of Classification

In this study, Naïve Bayes algorithm is used to classify the result of a job interview session between the user and the interviewer-bot. There are three categories of conversation, which are interests, potential and talent. Each category has the following classification: interest (not interested, lack of interest, interested, very interested), potential (unskilled, less skilled, skilled, highly skilled), and talent (visual, psychomotor) [11] [12]. Classification category is done by calculating the probability using Bayes Theorem. There are 10 of the 30 conversations of "interest" categories that belong to "very interested" to the job conversation. There are 18 of the 30 conversations that contain words listed on the "interest" categories such as "expect", "willingness", "effort", "interest", "concern", "enthusiasm", "support", "provide". Five conversations containing the words listed on the "interest" categories classified as "very interested" conversations. If there is a conversation that belongs to "interest" category made by the user and the interviewer-bot, then the probability of the conversation classified to a "very interested" conversation and contains the words: "expect", "willingness", "effort", "interest", "concern", "enthusiasm", "support", "provide", is calculated using the simple form of Bayes' theorem: [20]

- P(A)= The probability of a conversation belongs to “very interested” category
- P(B)= The probability of a conversation contains words listed on “interested” category (“expect”, “willingness”, ”effort”, ”interest”, ”concern”, “enthusiasm”, “support”, “provide”) → keyword.

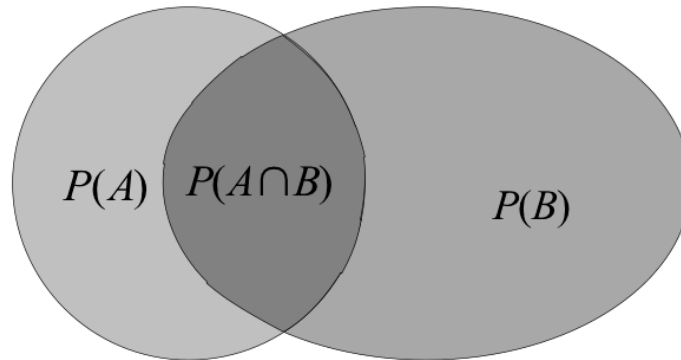


Fig. 1. Probability of occurring conversation

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(B|A)P(A)}{P(B)} \tag{1}$$

Thus, we have the following

$$P(A|B) = \frac{\frac{5}{10} * \frac{10}{30}}{\frac{18}{30}} = \frac{1/6}{3/5} = \frac{5}{18} = 0.27$$

3.2 Naïve Bayes Approach

Based on the calculation results, these conditions could be applied to solve the problems using Naïve Bayes approach as in [12] [21]:

There are five conversations containing the words (“learn”, “motivate”, “improve”) → keyword_B. These 5 conversations classified as “very interested” within the “interest” categories.
 The probability of a conversation contains the words (“expect”, “willingness”, “effort”, “interest”, “concern”, “enthusiasm”, “support”, “provide”) and (“learn”, “motivate”, “improve”) → keyword_A

The above problems become more complex, so it can no longer be solved by using a simple form of Bayes' Theorem, hence the modification formula is required:

$$P(A|B_1, B_2) = \frac{P(A|B_1 \cap B_2) * P(B_1|A) * P(A)}{P(B_1|B_2) * P(B_1)} \tag{2}$$

where $P(A)$ = very interested, $P(B_1)$ = Keyword_A, and $P(B_2)$ = Keyword_B

To simplify the problem, it is assumed that that the incidence of the emergence event of words listed inkeyword_A is not relying solely on the emergence event of words listed inkeyword_B on the conversation. So the above formula can be simplified into

$$P(A|B_1, B_2) = \frac{\frac{5}{10} * \frac{5}{10} * \frac{10}{30}}{\frac{18}{30} * \frac{6}{30}} \frac{250/3000}{108/900} = \frac{0.083}{0.12} = 0.694$$

The probability value means the possibility that the response given by the user during job interviews with the interviewer-bot contain words that are listed on keyword_A and keyword_B and classified as a "very interested" conversation. The above calculation is made to classify a conversation of "interest" categories belongs to a "very interested" conversation by using the probability of words contained in that conversation.

Naïve Bayes approach is done by making the assumption that the events of a words list group is not relying solely on the appearance of the others. This was done to simplify the probability calculation.

3.3 Phrase Reinforcement (PR) Algorithm

The PR algorithm used in this study refers to the implementation of Phrase Reinforcement used on Microblog to make a conclusion using these steps: the algorithm begins with the initial phrase, which is stating a topic that will be used to determine a conclusion. This topic usually is a trending topic or not. By using the initial phrase, Phrase Reinforcement algorithm submit a query to Twitter.com to get tweets containing the phrase. If the searched terms were just discussed on Twitter, the maximum Twitter data that could be taken is 1500 tweets (which are containing the search terms). If the searched terms had already discussed, for example the topic was discussed a few days, weeks, months or years ago, the Twitter data obtained is less than 1500 tweets, or it may not even exist at all. From these data, the algorithm will do the selection procedure to remove spam or unwanted tweet. Selection is an important step because spam and unwanted tweet can influence the conclusion drawn by the PR algorithm. Spam and unwanted tweets are selected using Naïve Bayes Algorithm which trained using spam data obtained previously from Twitter.com. Tweets contain non English term are also eliminated as well as the tweets that have the same content, because this study only focused on the conclusion stated in English [7] [13].

3.4 Naïve Bayes and Phrase Reinforcement

Drawing a conclusion from the interview results between user and interviewer-bot need a modified PR algorithm. Since the objective of this study is not to infer from Twitter.com, but from conversation logs that have been made previously between the user and the interviewer-bot, then the conversation logs were classified using Naïve Bayes algorithm prior to classify each category into sub-categories, for example:

- a) Interest: not interested, interested, very interested
- b) Potential: not trained, less trained, trained
- c) Talent/intelligence: logical-mathematical, visual-spatial, physical-kinesthetic, musical, interpersonal, intrapersonal, naturalist and existential.

These algorithms also make the selection of unwanted conversations, through the stop words and stemming process. After the conversations grouped and passed a selection process (the result can be referred to as training conversation), the PR algorithm could be used. The main idea of the algorithm is to generate sorted acyclic charts of all conversations (in one sub-category of the same category) drawn from training conversation. These charts are arranged around a central point contains the initial phrase that will be used as a reference of the conclusions. Modifications made in this process because the training conversation does not have the same phrase / word in every conversation. Step-by-step modification of these algorithms can be explained as follows:

- a) Initial phrases/words used as conclusion terms are replaced with a set of keywords defined in the Naïve Bayes algorithm section. Sample keywords for “interested” sub category within “interest” category are "expect", "willingness", "effort", "interest", "concern", "enthusiasm", "support", and "provide".
- b) Words similar to these keywords (specific to the words appear after keywords) will be processed deeper to get the weight of each word appear on conversation. This procedure applied to the words appear before and after the keyword. The process could be explained as follows:
 - Words that appear before keyword will be processed whether the words contain negative meaning or not (containing word "Not").
 - Words appearing after the keyword will be used as the focus of user' object about one of the keywords. For example: "expect", "willingness", "effort", "interest", "concern", "enthusiasm", "support", "provide".
 - Calculate the conversations weight that contain determined keywords. This weight is the calculation result of the frequent appearance of a keyword in training conversation. This step is done for each node or a number of N training conversations. Each selection done on each conversation will create a node that originated from keywords contained in each training conversation.

If the node is already established, then the selection process is done to the node that contains keywords > 1 and keyword = 1 or keyword = 0. It is the initial step to deduct a conclusion of all selected nodes. Here are the steps taken for each number of keywords:

For a node that contains keywords > 1 and positive meaning. Selections are made only to words appear after the keyword. For example, there are some conversations such as:

- I concern about world of journalism.
- I don't have any concerns about teaching.
- I concern in outdoor activities.

The above conversations contain the keyword "concern" with positive and negative meaning. Analysis of the *training conversation* results:

- **Concern** |about the world of journalism.
- **Don't** have any **concerns** |about teaching.
- **Concern** |in outdoor activities.

Positive meaning, concern (2), means:

"He **concerns about the world of journalism and outdoor activities**"

Negative meaning, don't concern (1), resulting in:

"He does not **concern about teaching**"

For a node that contains keywords = 1. Summary made directly from the keywords and the word that came afterwards, with the pronoun "She/He" and the process of changing the verbs used for the subject of "He/She"

For a node contains keywords = 0. Summary will be made, the same as the one for the node that contains keywords = 1, but not classified into sub-categories. On the other hand, the summary results will be analyzed by admin then put into the existing sub-categories.

The implementation of the appearance of different words lists and in different categories needs a formula that will calculate the probability of words lists in each category. Categories having the highest probability of the appearance of words lists (keyword) can be calculated as:

$$P(B_1, B_2, B_n) = A_{max}P(A) * \prod_{i=1}^n P(B_i|A) \quad (3)$$

where:

$P(B_1, B_2, B_n)$	Classification with 1 st keyword, 2 nd keyword, ... n^{th} keyword
$A_{max} P(A)$	Probability of maximum argument of a category A
$\prod_{i=1}^n P(B_i A)$	Probability of a keyword appears in a category

The following section presents implementation of this study.

4 Implementation

The implementation section is divided into two processes i.e. training and elimination of stop words and stemming which is discussed as following:

4.1 Training

Training is used to show several conversation examples to a particular category and its results will be used to calculate the probability of the next conversation classification. Here is a training example of “interest” category:

```
"not interested": I don't know anything about this company.  
"not interested": I work in different field that does not related at  
all to this job.  
"interested": I have done project that related to this job before.  
"interested": I like to share with people and work with them in team.  
"very interested": I'm detailed person and enthusiastic about world  
of journalism.  
"very interested": I'd like to work in job that provides opportuni-  
ties to meet a lot of people and learn from them.
```

Each subcategory of “interest”category has conversations examples that indicate which conversation belongs to "not interested", "interested", and "very interested" subcategory. The more conversation examples trained in each category, the more accurate the probability of the next conversation belongs to the category.

4.2 Elimination of stop words and stemming

This step is used to eliminate the words often said in conversation and did not have a specific meaning related to the determination of categories; for example the words: the, on, it, that, you, to, be, if, what, there, since, and others. Thus, after the removal of the stop words, the words having special meaning associated with a particular category could be obtained. For instance, for the conversation of "very interested" subcategory within the “interest” category, words having special meaning associated with this category are detailed person, enthusiastic, opportunities, and others. These words are stored in a file, which is used as a reference at the time of software implementation. The next step is the word removal procedure to form the origin words (stemming). This algorithm does not consider word context and conversation grammar. For example, learn, learns, learned, learning has a different context but have the same origin of the word is "learn". This would be a problem because the conversation is done at different times, place and circumstances. Hence, there should be a method to eliminate the words in the form of past tense, continuous, or plural into its origin. Naïve Bayes classification flow chart for one of the categories (“interest”) has the same calculation with the other categories; the difference is the number of subcategories used in each conversation category. The Naïve Bayes classification flow chart for one of the categories shown in Figure 2.

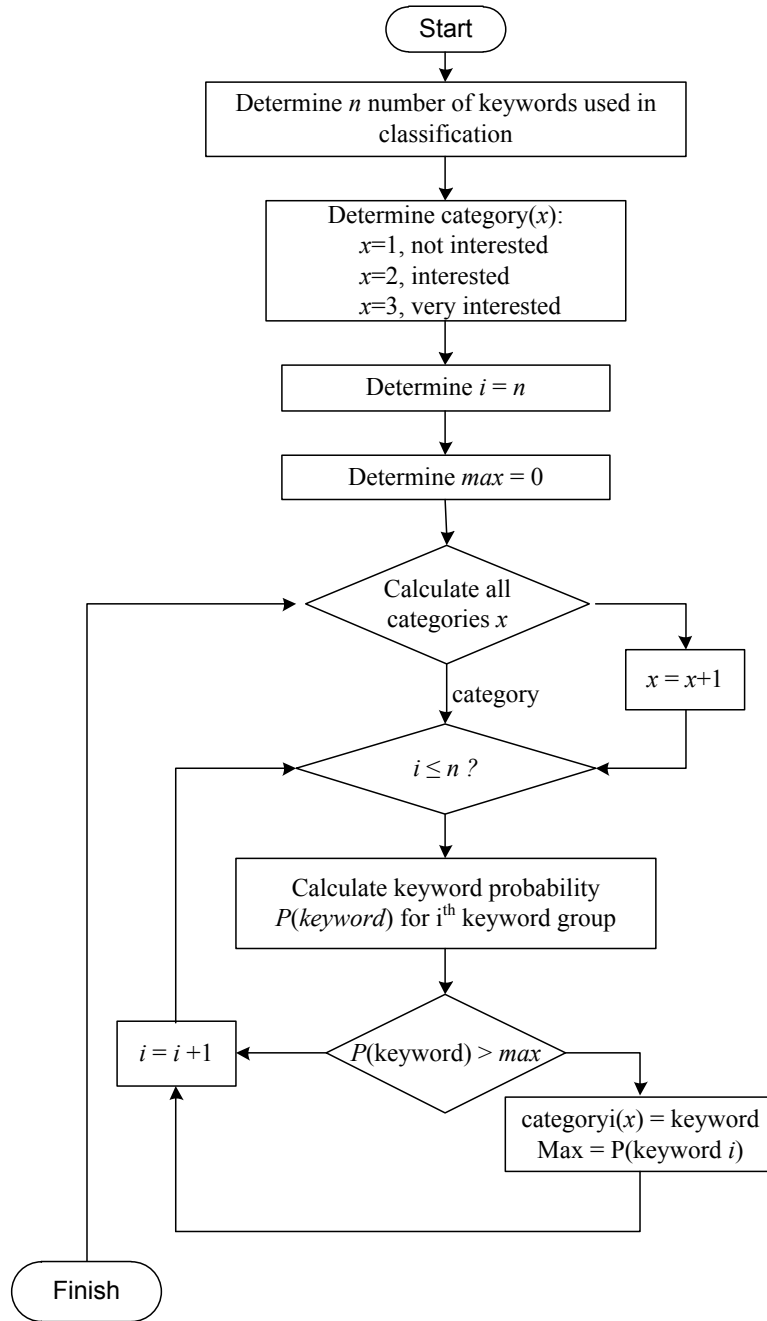


Fig. 2. Naïve Bayes algorithm flowchart

5 Results and Discussion

Software testing includes test verification and validation.

5.1 Verification Test

This test is intended to determine whether the translation of conceptual models into software applications, in this case an interviewer-bot, has been carried out properly or not. Verification test is done by comparing sub system model base design with algorithms, also by comparing the algorithms with program lists. The comparison process is performed in order to be aware of the relationship between design base models, algorithms, and coding (program lists) when the program starts. The result shows that the system has been in accordance with the analysis and design as well as meeting the required functionality in the interviewer-bot system.

5.1. Validation Test

This test is intended to evaluate the function suitability of the system, whether the function of interviewer-bot in these systems can be met by providing input and analyzing output or whether the response given is in accordance with the desired one. In validation test, the result of the response given by the interviewer-bot is compared with the response given by the user based on statements or answers related to the word or phrase related to the data, interests, potentials, and talents; and also word or phrase that is not related to the user' private data, interest, talent, and potential. The test is also conducted by comparing the responses given by the interviewer-bot, which includes:

- Topic relevance between the users' input words or phrases with the response words or sentences given by the interviewer -bot.
- Response by interviewer-bot if the input or sentence is not related to the user' private data, interest, potential, and talent.
- Response time required by the interviewer-bot to give a statement or an answer to the user related to the user' private data, interest, talent, and potential, and vice versa.

Validation test is performed by giving questionnaires to 50 respondents who did chat without knowing that interviewer-bot is a software. User satisfaction rate is calculated by:

- Questions number 1-6 and number 12 tests the interviewer bot response.
- Question number 7-11 test interviewer bot response time.

In summary, the results are given in Table 1 as following.

Table 1. Questionnaire result of 50 respondents

No	User satisfaction parameter	Response percentage
1	Questions' topic	89%
2	Answers or responses related to users' interest	88%
3	Answers or responses related to users' potential	87,5%
4	Answers or responses related to users' talent	90,5%
5	Answers or responses notrelated to users' data, interest, potential, and talent	93%
6	The relevance between users' input with interviewer-bot responses	83,5%
7	Immediate response related to users' data	77%
8	Immediate response related to users' interest	78%
9	Immediate response related to users' potential	87%
10	Immediate response related to users' talent	79%
11	Immediate response not related to users' private data, interest, potential, or talent	83%
12	Were you satisfied being interviewed by the interview-bot?	77%

From table 1, result based on the rate of interviewer-bot response and response time shows that users can access the system and utilize features on the interviewer-bot website. The average satisfaction rate for interviewer-bot response is 86.93% while the average satisfaction rate for response time or the speed of interviewer-bot giving answer is 80.8%. The averages of the two responses are still above the average satisfaction rate required by the Turing Test, which is 70%. Thus interviewer-bot is worthy to be called Virtual Job Interviewer with Artificial Intelligence. Here is an example of the test results for the interviewer-bot response during interviews with users:

```

Human : Hello
System : Hi there!
Human : How are you?
System : I'm, doing fine thanks. How are you?
Human : I'm fine, are you crazy
System : I think the polite term nowadays is "mentally ill".
Human : you monkey
System : I haven't heard anything like that before.
Human : I love you
System : Thanks I like you a lot too my friend.
    
```

6 Conclusion

This paper presented the development of a machine which is able to provide practice on English-based interviews, specifically on job interviews. The implementation of Naïve Bayes classification method and Reinforcement Phrase algorithm in AIML-based O Program software has accuracy level of interviewer-bot response by 86.93% as proven through validation test by 50 respondents who conducted chats without knowing that Interviewer Bot is a software. The response indicating that the Naïve Bayes method and Reinforcement Phrase algorithm can be used to classify the inter-

viewer-bot interview result. In addition, interviewer-bot system has the intelligent ability so it could contribute positively for students practicing interview in English.

7 Acknowledgement

This research is supported by the Directorate General of Higher Education, Indonesia through the “Leading Research Universities” (PUPT) grants.

8 References

- [1] Bodnar C.A. and Clark R.M., "Can Game-Based Learning Enhance Engineering Communication Skills?," *IEEE Transactions on Professional Communication*, 2017. <https://doi.org/10.1109/TPC.2016.2632838>
- [2] Villani D., Rotasperti C., Cipresso P., and Tribert , "Assessing the Emotional State of Job Applicants Through a Virtual Reality Simulation: A Psycho-Physiological Study. In *eHealth 360°*," Springer Internat, pp. pp. 119-126, 2017.
- [3] Chiru C. G. and Rebedea T., "Profiling of Participants in Chat Conversations Using Creativity-Based Heuristics.," *Creativity Research Journal*, vol. 29(1), pp. 43-55., 2017. <https://doi.org/10.1080/10400419.2017.1267464>
- [4] Negnevitsky M., *Artificial intelligence: a guide to intelligent systems.*: Pearson Education, 2005.
- [5] Collobert R. and Weston J., "A unified architecture for natural language processing: Deep neural networks with multitask learning.," in *In Proceedings of the 25th international conference on Machine learning*, 2008, pp. (pp. 160-167). ACM. <https://doi.org/10.1145/1390156.1390177>
- [6] Heller B., Procter M., Mah D., Jewell L., and Cheung B., "Freudbot: An investigation of chatbot technology in distance education.," in *In Proceedings of the World Conference on Multimedia, Hypermedia and Telecommunications*, 2005, pp. pp. 3913-3918.
- [7] Shawar B.A. and Atwell E., "Using corpora in machine-learning chatbot systems," *International journal of corpus linguistics*, vol. 10 (4), pp. 489-516., 2005. <https://doi.org/10.1075/ijcl.10.4.06sha>
- [8] Wallace R., *The elements of AIML style.*: Alice AI Foundation., 2003.
- [9] ALICE. (2002) A.L.I.C.E AI Foundation. [Online]. HYPERLINK "%20http://www.alicebot.org/%20or%20http://alicebot.franz.com/%20" <http://www.alicebot.org/> or <http://alicebot.franz.com/>
- [10] Cho K. A., "Emotional and Domain Concept Enhancements to Alice bot.," *The Faculty of the Department of Computer Science. San José State University.*, 2007.
- [11] Arifin D.D. and Bijaksana M. A., "Enhancing spam detection on mobile phone Short Message Service (SMS) performance using FP-growth and Naive Bayes Classifier. In *Wireless and Mobile (APWiMob)*," in *IEEE Asia Pacific Conference on*, September 2016, p. 8.
- [12] Patil T.R. and Sherekar S.S., "Performance analysis of Naive Bayes and J48 classification algorithm for data classification.," *International Journal of Computer Science and Applications*, vol. 6(2), pp. 256-261., 2013.
- [13] Jiwangi M.A. and Adriani M., "Topic Summarization of Microblog Document in Bahasa Indonesia using the Phrase Reinforcement Algorithm.," *Procedia Computer Science*, vol. 81, pp. 229-236, 2016. <https://doi.org/10.1016/j.procs.2016.04.054>

- [14] Zandstra M., Zandstra, M. (2016). PHP Standards. In PHP Objects, Patterns, and Practice.: Apress., 2016. https://doi.org/10.1007/978-1-4842-1996-6_15
- [15] Naim I., Tanveer M.I., Gildea D., and Hoque E., "Automated analysis and prediction of job interview performance.," IEEE Transactions on Affective Computing., 2016.
- [16] Burguillo F.A., Rodríguez J.C., Rodríguez D.A., Lamas E, and Mikic M, "T-Bot and Q-Bot: A couple of AIML-based bots for tutoring courses and evaluating students.," in In Frontiers in Education Conference, 2008. FIE 2008. 38th Annual, 2008, pp. pp. S3A-7. IEEE.
- [17] De Gasperis G., "Building an aiml chatter bot knowledge-base starting from a faq and a glossary.," Journal of e-Learning and Knowledge Society, vol. 6(2).
- [18] Fatmuriyah M, Sarosa M, and Santoso P.B., "Implementasi Chatter bot Berbasis Program O Menggunakan Web Crawler dan Web Service.," Journal ELEKTRAN Department of Electrical Engineering, State Polytechnic of Bandung., 2012.
- [19] Lim S.L. and Goh O.S., "Intelligent Conversational Bot for Massive Online Open Courses (MOOCs).," arXiv preprint arXiv:1601.07065., 2016.
- [20] Junus M., Sarosa M., Fatmuriyah M., Hoesny M.U., and Sari Z., "Interviewer BOT Design to Help Student Learning English," in International Conference on Information Technology and Security (IC-ITechS), Malang, 2014, pp. 45-50.
- [21] Wang Y. and Tseng M.M, "A Naïve Bayes approach to map customer requirements to product variants.," Journal of Intelligent Manufacturing, vol. 26(3), pp. 501-509., 2015. <https://doi.org/10.1007/s10845-013-0806-2>

9 Authors

Moehammad Sarosa received the diploma of engineering technology from Université de Nancy I, France in 1989. His master and doctoral degrees in computer technology is received from Bandung Institute of Technology, Indonesia, in 2002 and 2007, respectively. He is currently lecturer in Department of Electrical Engineering of both Brawijaya University and State Polytechnic of Malang. His current research interests lie in information communication technology, decision support system, artificial intelligence, and mobile computing.

Mochammad Junus is currently a lecturer at Department of Electrical Engineering, State Polytechnic of Malang and currently served as Secretary of same department. His research areas includes wireless communication system, antenna design, and network traffic engineering.

Mariana Ulfah Hoesny has been an English lecture in State Polytechnic of Malang since 2006. She gained her Bachelor in English Literature from State University of Malang. She was graduated as Master of Language and Literature Education in 2012 from State University of Surabaya. She taught at University of Szeged, Hungary in 2015. Her articles mainly focused on teaching and the use of information technology in language teaching. Her main interests are language teaching and ICT.

Zamah Sari is currently a lecturer at the University of Muhammadiyah Malang majoring in Informatics. He got his Bachelor's degree of Electrical Engineering University of Muhammadiyah Malang, and Master's degree Electrical Engineering University of Brawijaya Malang. His research interests lie in the fields of Food

Security and Renewable Energy, Artificial Intelligence, Data Mining, Networking, and IoT.

Martin Fatnuriyah is co-founder of the software development and IT consultant (Myriatek). She graduated from Sekolah Tinggi Teknologi Telkom with a bachelor's degree in Telecommunication Engineering, and a master's degree in Electrical Engineering from Brawijaya University. She is currently a Senior Software Engineer at Wirednest Singapore. Her research interest includes artificial intelligence, machine learning, software analyst, software architect, web (Laravel, Lumen, Javascript, jQuery, AngularJS) and mobile (IOS, Android, Ionic) development, REST API, and .NET.

Article submitted 18 May 2017. Resubmitted 15 October 2017. Final acceptance 04 February 2018. Final version published as submitted by the authors.