Combination of Natural Language Understanding and Reinforcement Learning for Booking Bot

1st Devira Anggi Maharani
School of Electrical Engineering and Informatics
Bandung Institute of Technology
Bandung, Indonesia
deviraanggi@students.itb.ac.id

2nd Dinda Ayu Permatasari
Electrical Engineering Department
State Polytechnic of Malang
Malang, Indonesia
prmtsrdinda@gmail.com

Abstract— **At present, some popular messaging applications have evolved specifically with bots starting to emerge into development. One of the developments of chatbots is to help humans booking flight with Named Entity Recognition in the text, trace sentences to detect user intentions, and respond even though the context of the conversation domain is limited. This study proposes to conduct analysis and design chatbot interactions using NLU (Natural Language Understanding) with the aim that the bot understands what is meant by the user and provides the best and right response. Classification using Support Vector Machine (SVM) method with (erm Frequency-Inverse Document Frequency (TF-IDF) feature extraction is suitable combination methods that produce the highest accuracy value up to 97.5%. Conversation dialogue on chatbots developed using NLU which consists of NER and intent classification then dialog manager using Reinforcement Learning could make a low cost for computing in chatbots.**

Keywords—chat bot, natural language understanding, reinforcement learning, SVM, TF-ID, NER

I. INTRODUCTION

Current world, our way to interact with digital devices is mostly limited, based on what features and accessibility offered on each device. Simply put, there is a learning curve associated with every new device that we interact with. Chatbots provides a solution to this problem by interacting with users automatically. Chatbot is currently the easiest means for software to be as genuine to humans as it provides the experience of talking to others. One of the developments of chatbots is to help humans booking a flight and also find comfortable restaurants. Also, chatbots (such as [1]) are for entertainment. Response from the chatbots that are built is the result of understanding natural language so that the computer only concludes what is actually meant by the speaker and emphasized the words they utter. Building an effective chatbot consisting of software must identify entities in the text, trace sentences to detect user intentions and respond even though the context built on a limited conversation.

Many of these chat agents are built using rule-based techniques, retrieval techniques or simple machine learning algorithms. In retrieval techniques, chat agents scan for keywords within the input phrase and retrieves relevant answers based on the query string. Relevant answer depends on similarity of keywords and text taken from internal or external data sources including the worldwide web or organizational database. Some other advanced chatbots are developed with Natural Language Processing (NLP) techniques and machine learning algorithms. Also, there are many commercial chat engines available, which help build chatbots based on client data input.

On the other hand, with a goal-oriented dialogue system [2] [3], chatbots are also used to chat with human users for any subject in everyday life [4] [5]. The conventional chatbot is based on the seq2seq model [6] which generates a response which means user input. Generally without emotion, the main limitation of chatbots today is that emotion is an important role for social social interaction in chatting [7]. There are two problems that arise even though the Seq2Seq model has been successful in the generation dialog [8]: First, predicting the next dialogue turn in a particular conversation using the maximum likelihood estimation (MLE) function using the SEQ2SEQ model. Another problem is that the system gets stuck in repeated loops and repetitive responses

We draw on the insights of reinforcement learning, which have been widely applied in Q-Learning dialogue systems to achieve the goals. Reinforcement learning (RL) method used in this paper, which this method can optimized for long-term rewards design. The goal-oriented dialogue systems are consisted of several subcomponents. In general, there are three steps to implementing dialogue systems in chatbots such as Q-Learning as a method for dialogue managers, which are fully monitored by applying machine learning to classify intentions such as Support Vector Machine (SVM), Naïve Bayes, and K-Nearest Neighbors methods, and Named Entity Recognition (NER) labels sequences of words in a text.

II. PROPOSED APPROACHES

In this study consists of several sub-chapter approaches, such as named entity recognition (NER), machine learning, and dialog managers.

A. Named Entity Recognition (NER)

Named Entity Recognition (NER), which functions to label the word order in the text in the form of object names, or people or companies, names of genes and others. Current natural language processing, which mostly uses statistical models only represents local structures. Although this is very important to allow traceable model inference, in many tasks this is the main limitation because natural languages contain nonlocal structures [9]. CRF are basically a way to combine the advantages of classification and graphic modeling, multivariate data modeling with capabilities that are combined to utilize a large number of input features for prediction. The generative model and the CRF have exactly the same differences as the Naive Bayes and logistic regression classifiers. Indeed, simplest type of CRF is multinomial logistic regression model, in which there is only one output variable [10].
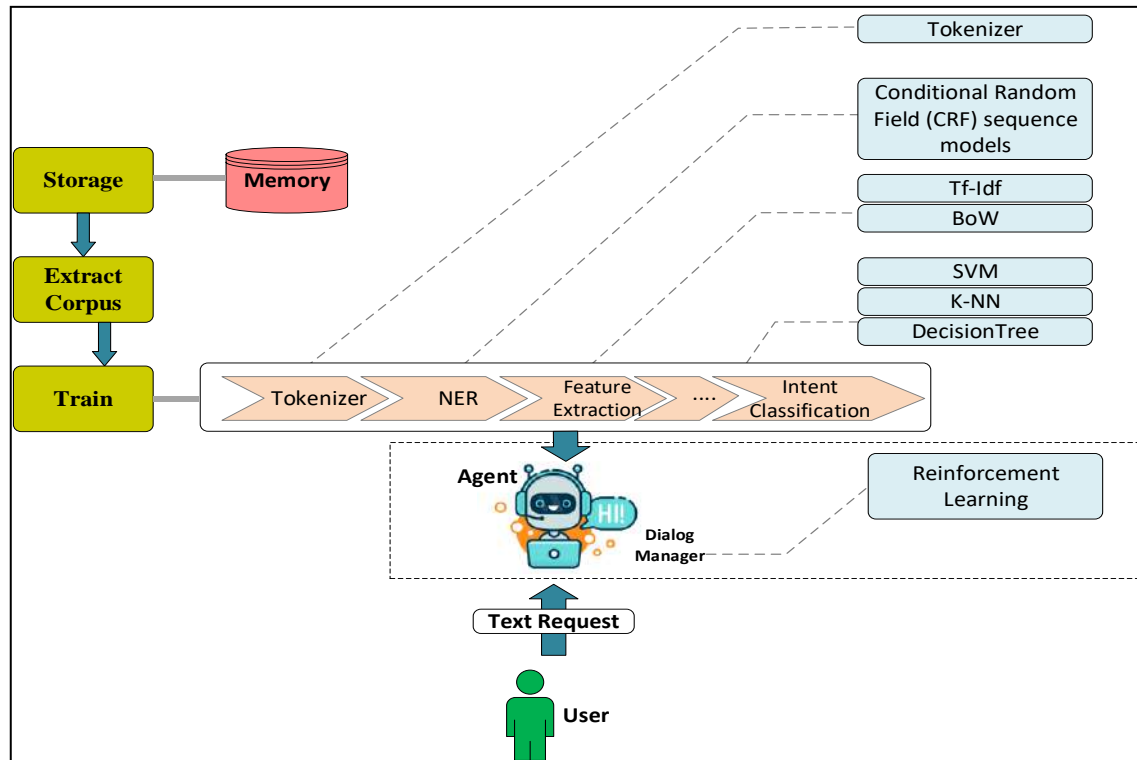
12

Journal of Electrical, Electronic, Information, and Communication Technology (JEEICT)
Vol. 3 No. 1, April 2021, Pages 12-17

Fig. 1. Proposed System

B. Machine Learning

Machine learning can be categorized into major groups as shown in Fig. 2. These groups represent how the learning method works.
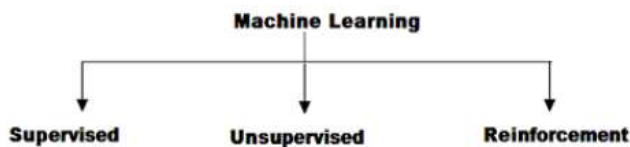

Fig. 2. Machine learning types [11]

Supervised learning consists of reasonable algorithms or learning from externally provided examples to produce general hypotheses that makes predictions about future events. There is an outcome or output variable to guide the learning process. Many supervised learning algorithms such as decision trees, K-Nearest Neighbor, Support Vector Machines and Naïve Bayes.

K-Nearest Neighbor KNN or what is called instant learning which is a type of supervised learning algorithm. KNN only stores the current memory-driven data set and when a new query is executed, that set of instances or the same neighbor is retrieved from memory, then used to classify new instances [12]. It is very beneficial to consider more than one neighbor at the same time, the classification is called K-Nearest Neighbor. This nearest neighbor is measured by the Euclidean distance which means it measures between the measured sample as a vector input and some related measures.

This system interacts with the environment with the results in the form of certain actions on reinforcement learning. Such actions affect the condition of the surrounding environment, which in turn produces a machine that accept a scalar gift (or

punishments). The purpose of this machine is to learn to act by maximizing the future rewards received (or minimizes the punishments) over its lifetime.

C. Dialogue Manager

Form of dialogue for each dialog that is built represents the information the user wants to be processed by the system and to improve the intent of the dialogue [13]. Generating the appropriate response for the user, the system changes the dialog conditions on the total dialog and is also responsible for representing the dialogue state in turn from the NLU results. On this study uses reinforcement learning for dialog managers. In assessing the success of the dialogue, it can be said if the dialogue is successful if the dialogue agent succeeds in answering all user requests into four dialog turns. Based on this, the reward function is set to reward 10 for successful dialogue, and a -10 penalty for failed dialogue. Fig. 3. Shows how RL works in dialog managers.
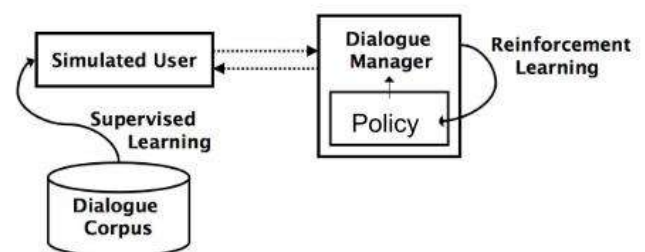

Fig. 3. Reinforcement learning process in dialog manager [14]

III. DESIGN AND IMPLEMENTATION

The chatbots design for flight booking uses the NLU which creates chatbots understand the message from the user and

respond correctly. When a user sends a message with "Thank you", this NLU lets the chatbot know that the user has posed a standard greeting, which allows chat to leverage its AI capabilities to come up with a fitting response. In this research, the chat will likely respond with a return greeting. The method offered is as shown in Fig. 1 which consists of:

A. Preprocessing

At the preprocessing step consists of tokenization of the word to split a sentence into words. Word tokenization becomes a crucial string to numeric data conversion. Fig. 4. Shows the word tokenize module is imported from the system library in c # language program.
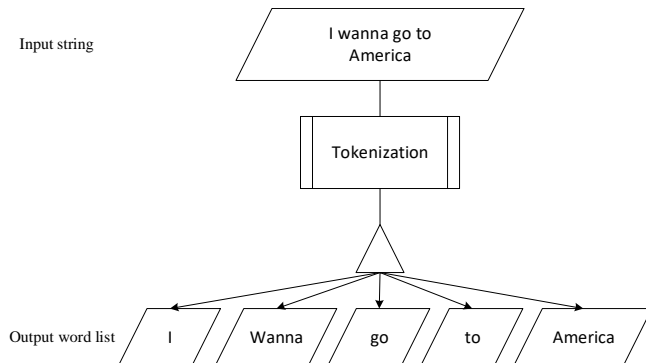

Fig. 4. Word tokenization

B. Named Entity Recognition

After tokenization in the sentence, the next process is entity identification, entity chasing and entity extraction by using a model made by Stanford, namely Conditional Random Field (CRF) which was pioneered by Lafferty, McCallum, and Pereira in 2001. Introduction of the entity later it will produce pre-defined categories such as location, departure date, departure time, and ticket price. This is done to make flight ticket reservations more specific consisting of the 4 entities above. In the expression named entity, task to one or many string entities is sufficient for multiple references, although in practice NER deals with many names and references that are not philosophically rigid. At Fig. 5. indicates the NER that can be used to determine the entity from the result of the intent classification.

Fig. 5. Named Entity Recognition using CRF

C. Feature extraction

Feature extraction involves reducing the amount of resources required to describe a large set of data. It creates a vocabulary of all the unique words occurring in all the documents in the training set. In this research, feature extraction uses Bag of word (BoW) and TF-IDF where BoW represents text that describes the appearance of words in the document and involves known vocabulary and the size of the presence of known words. The complexity of these two determines how to design the vocabulary of known words / tokens and how to judge the emergence of known words. The easiest scoring method is to mark the apperance of words as

a Boolean value, 0 for not appear, 1 for appear. Fig. 6. shows that the structure of BoW for intent classification.

Fig. 6. Feature extraction using Bag of Word

Each word that has been tokenized, is transformed in to 435 arrays with a total of 1200 training data sets and 40 testing data. Then, for feature extraction that will be compared later using the TF-IDF (term frequency–inverse document frequency) method. This approach involves changing the word frequency scale so that the intensity of the word appearing in all documents, so the scores for the most frequent words such as "the" and also appearing frequently in all documentary meals will be penalized. From this approach it can be said that Term Frequency - Inverse Document Frequency is usually abbreviated as TF-IDF, Term Frequency itself is a weighting of the frequency of words that appear in the document while for the reverse document frequency it is the weighting of how often the word appears throughout the document. Fig. 7 shows the structure of TF-IDF as method for feature extraction.

Fig. 7. Feature extraction using TF-IDF

D. Intent Classification

After the feature extraction step, the next step is the intent classification where at this step using 1200 training data and 40 as testing data. Examples of sentences that will be classified using 3 methods that will be compared are SVM, K-NN, and Naïve Bayes as shown in Fig. 8.

14

Journal of Electrical, Electronic, Information, and Communication Technology (JEEICT)
Vol. 3 No. 1, April 2021, Pages 12-17

Fig. 8.   Intent classification with 4 classes

Intent classification aims to make the machine understand what is written by the user so that it does not only use similar algorithms such as using cosine similarity. Intent classification for chatbots, where small data sets are made of 1240 sentences. Sometimes, users often make spelling mistakes and, models for learning are not trained in ways that users will make mistakes. Models depending on word vocabulary will always face such problems. The ideal classifier must handle spelling errors inherently. With intent classification, we overcame this challenge and sophisticated results in four classes, such as ticket price destinations, greetings, and dealing sentences as in Fig. 8.

E. Dialog Managers

An RL-based dialogue system, responses from users and agents are converted into dialogue action and temporarily assigned a value of 10 to build a values and policies networks. After the RL policy determines the best of action, then the appropriate value is carried out, then is the appropriate slot for the post-processing step. The dialogue state of each of these dialogs contains information that the user wants the system to do. So in order to make an appropriate response, the system must track changes in the dialogue status during the entire dialogue and is also obliged to represent the dialog status using the results from the NLU.

Network parameters are optimized to maximize what is expected in the future with policy lookups. The method used for policy gradients uses Q-Learning [15] which considers the expected future rewards of each action. An action a is the dialogue utterance to generate. The action space is infinite since arbitrary-length sequences can be generated. RL formula using Q-Learning policy with value of each parameter Episodes =1000, α=0.1, γ=0.9.

$$Q(s,a) = Q(s,a) + \alpha[r(S,A)+ \gamma \max(Q'(s',a')-Q(s,a)] \quad (1)$$

IV. EXPERIMENTS AND RESULTS

Table 1 summarizes the comparative training accuracy results of intent classification using SVM, K-NN, and Naïve Bayes with BoW and TF-IDF as a feature extraction method.

TABLE I.   TRAINING ACCURACY

Feature Extraction	Method	Chi square	Accuracy	Recall	Precision
TF-IDF	SVM	3600	100%	c1 = 1 c2 = 1 c3=1 c4=1	c1 = 1 c2 = 1 c3=1 c4=1
	kNN	3600	100%	c1 = 1 c2 = 1 c3=1 c4=1	c1 = 1 c2 = 1 c3=1 c4=1
	Naïve Bayes	3485	99.5%	c1 = 1 c2 = 1 c3=1 c4=0.95	c1 = 1 c2 = 1 c3=0.95 c4=0.98
BoW	SVM	3576	99%	c1 = 1 c2 = 1 c3=0.99 c4=1	c1 = 1 c2 = 1 c3=1 c4=0.99
	kNN	3600	99.7%	c1 = 1 c2 = 1 c3=0.97 c4=1	c1 = 0.99 c2 = 1 c3=1 c4=0.98
	Naïve Bayes	3485	99.5%	c1 = 1 c2 = 1 c3=1 c4=0.95	c1 = 1 c2 = 1 c3=0.95 c4=0.98

TABLE II.   TESTING ACCURACY

Feature Extraction	Method	Chi square	Accuracy	Recall	Precision
TF-IDF	SVM	112.7	0.975	c1 = 1 c2 = 1 c3=1 c4=0.9	c1 = 0.9 c2 = 1 c3=1 c4=1
	kNN	23.9	0.425	c1 = 1 c2 = 0.3 c3=1 c4=0	c1 = 0.6 c2 = 1 c3=0.1 c4=0
	Naïve Bayes	105.9	0.95	c1 = 1 c2 = 1 c3=1 c4=0.83	c1 = 0.9 c2 = 1 c3=0.9 c4=1
BoW	SVM	106.6	0.95	c1 = 1 c2 = 1 c3=1 c4=0.83	c1 = 0.8 c2 = 1 c3=1 c4=1
	kNN	101.5	0.925	c1 = 1 c2 = 1 c3=1 c4=0.76	c1 = 0.7 c2 = 1 c3=1 c4=1
	Naïve Bayes	105.9	0.925	c1 = 1 c2 = 1 c3=1 c4=0.83	c1 = 0.9 c2 = 1 c3=0.9 c4=1

Table 2 shows the comparative testing accuracy results of intent classification using SVM, K-NN, and Naïve Bayes with BoW and TF-IDF. The results show that the lowest accuracy is by using the kNN method and TF-IDF feature extraction, this is due to data sets for testing which should be class 4 (dealing) registered in class 2 (destination) this can be caused by overfitting and data the set still has similar words. Preprocessing uses TF-IDF sometimes the dimensionality of text data which affects the size of the vocabulary across the entire data set and it brings out a huge computation of the occurrence in each document.

The kNN method requires the most optimal k value which states the number of nearest neighbors so that in the selection it must be appropriate and the most optimal selection of k values is shown in Table 3.

15

TABLE III.   TESTING ACCURACY (kNN ALGORITHM)

Feature Extraction	k	Chi square	Accuracy	Recall	Precision
TF-IDF	100	23.9	0.425	c1 = 1 c2 = 0.3 c3=1 c4=0	c1 = 0.6 c2 = 1 c3=0.1 c4=0
	4	94.72	0.90	c1 = 0.75 c2 = 0.90 c3=1 c4=1	c1 = 0.9 c2 = 1 c3= 1 c4=0.7
	10	86	0.85	c1 = 1 c2 = 1 c3=0.6 c4=1	c1 = 0.9 c2 = 1 c3= 1 c4=0.6
	20	82.35	0.82	c1 = 1 c2 = 1 c3=0.58 c4=1	c1 = 0.8 c2 = 1 c3= 1 c4=0.5
BoW	100	97.14	0.90	c1 = 1 c2 = 1 c3=1 c4=0.71	c1 = 0.6 c2 = 1 c3= 1 c4=1
	4	90.13	0.875	c1 = 1 c2 = 1 c3=0.67 c4=1	c1 = 0.8 c2 = 1 c3= 1 c4=0.7
	10	86	0.85	c1 = 1 c2 = 1 c3=0.6 c4=1	c1 = 0.8 c2 = 1 c3= 1 c4=0.6
	20	86	0.8685	c1 = 1 c2 = 1 c3=0.6 c4=1	c1 = 0.8 c2 = 1 c3= 1 c4=0.6

Based on the accuracy in Table 3, it is obtained if using TF-IDF feature extraction the accuracy of testing data drops dramatically with a value of k = 100 which is 0.425 while using BoW feature extraction reaches 0.90 this can be caused by the TF-IDF class 4 feature dealing same as class 2 (destination).For a confusion matrix, true or false, positive or negative refers to a classification that has been determined to be true or false, while positive or negative refers to positive or negative categories as in Table 4.

TABLE IV.   CONFUSION MATRIX DATA TESTING

Feature Extraction	Method	TP	FP	FN	TN
TF-IDF	SVM	c1 = 9 c2 = 10 c3=10 c4=10	c1 = 0 c2 = 0 c3=0 c4=0	c1 = 1 c2 = 0 c3=0 c4=0	c1 = 30 c2 = 30 c3=30 c4=29
	kNN k=4	c1 = 9 c2 = 10 c3=10 c4=7	c1 = 3 c2 = 1 c3=0 c4=0	c1 = 1 c2 = 0 c3=0 c4=3	c1 = 27 c2 = 29 c3=30 c4=30
	Naïve Bayes	c1 = 9 c2 = 10 c3=9 c4=10	c1 = 0 c2 = 0 c3=0 c4=2	c1 = 1 c2 = 0 c3=1 c4=0	c1 = 30 c2 = 30 c3=30 c4=28
BoW	SVM	c1 = 8 c2 = 10 c3=10 c4=10	c1 = 0 c2 = 0 c3=0 c4=2	c1 = 2 c2 = 0 c3=0 c4=0	c1 = 30 c2 = 30 c3=30 c4=28
	kNN k=4	c1 = 8 c2 = 10 c3=10 c4=7	c1 = 0 c2 = 0 c3=5 c4=0	c1 = 2 c2 = 0 c3=0 c4=3	c1 = 30 c2 = 30 c3=25 c4=30
	Naïve Bayes	c1 = 9 c2 = 10 c3=9 c4=10	c1 = 0 c2 = 0 c3=0 c4=2	c1 = 1 c2 = 0 c3=1 c4=0	c1 = 30 c2 = 30 c3=30 c4=28

TABLE V.   DATA TESTING

Dialog	SVM class	kNN class	Naïve Bayes class
" yeh ",	3	3	3
" yes it is correct ",	3	3	3
" yeah that's right ",	3	3	3
" yap ",	3	3	3
" confirmed ",	3	3	3
" yes that's fine ",	3	3	3
" yeah go ahead ",	3	3	3
" yeah I'm sure ",	3	3	3
" okay that's fine ",	3	3	3
" yeah exactly ",	3	3	3
" thank you ",	2	2	3
" thanks",	2	2	2
" thanks a lot ",	2	2	2
" terrific thank you ",	2	2	2
" great thank you ",	2	2	2
" thanks so much ",	2	2	2
" thank you so much",	2	2	2
" thanks for your help”	2	2	2
" thank you for your help ",	2	2	2
" nice thank you ",	2	2	2
" Ticket prices return to Iceland ",	0	0	0
" what is the price of a ticket to go to Ireland",	0	0	0
" what is the price of a ticket to go to Israel",	0	0	0
" what is the price of a ticket to go to Kenya",	0	0	0
" what is the total price of this ticket ",	0	0	0
" return ",	3	3	0
" tariff ",	3	3	3
" the cheapest ticket price go to Algeria ",	0	0	0
" ticket for family ",	0	3	0
" promo ticket this month ",	0	0	0
" I need to be in Denmark ",	1	1	1
" I need to be in Djibouti ",	1	1	1
" I need to be in Dominica ",	1	1	1
" I need to be in Dominican Republic ",	1	1	1
" I wanna be in Australia ",	1	1	1
" I wanna be in Austria ",	1	1	1
" I wanna be in Azerbaijan ",	1	1	1
" I have to be in Guinea ",	1	1	1
" I have to be in Guyana ",	1	1	1
" I have to be in Haiti ",	1	1	1

Table 5 shows the class for each dialogue with 3 types of classifiers such as SVM, KNN, and NB. Class 0 as a class of ticket prices, class 1 as a destination class, class 2 as a greeting, and class 3 as a dealing. Class errors can occur due to the similarity of training data between the *greeting* and *dealing* classes so that *"thank you"* that should have entered the *greeting* class goes into the *dealing* class when using the Naive Bayes method. Then, for the word *"tariff"* almost all of it goes into the *dealing* class and word *"return",* this is because only one word is almost the same as the *greeting* class which mostly has one to three words in one sentence which makes preprocessing using TF-IDF produces the same value in array 435.
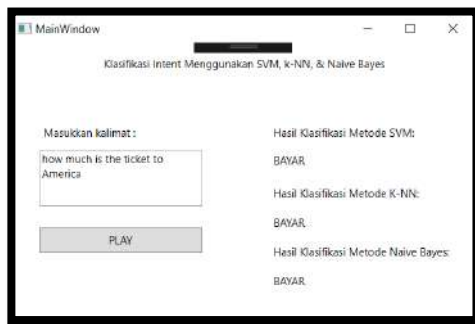
Fig. 9. Intent classification GUI

Fig. 10. Chatbots GUI

Fig. 11. Chatbots GUI

Dialogue managers use RL and their policies are based on Q-Learning rules. As shown in Table 4 and 5, the rule-based policy always achieves Q maximum value from the destination state to the destination state with an estimated Q = 1.89, from the state destination to the ticket price with an estimated maximum Q = 0.11, from the destination state to the ticket price with estimated maximum Q = 0.11, from the ticket price state to the agreement with estimated maximum Q = 18.92, from the state agreement to thank you with estimation of maximum Q = 1.1.

V. CONCLUSION

For the classification system using the SVM method with TF-IDF feature extraction is a suitable combination method that produces the highest accuracy value of 97.5%. Conversation dialogue on chatbots developed using NLU which consists of NER and intent classification and the dialog manager using RL is one of methodology combination which is suitable for low cost for computing in chatbots. It can be concluded that reinforcement learning is a simple but powerful technique and has a tremendous potential to contribute to the development of AI-based conversations combined with the NLU. In the AI conversation, challenges related to reinforcement learning are related to the reward function so how to measure user experience and personalization in terms of reward functions is one of the future research and development in other bot fields.

REFERENCES

[1] A. A. I. Foundation and N. Kramer, "Super aiml bot," in *ALICE AI Foundation*, 2006.

[2] C. Lee, S. Jung, S. Kim, and G. G. Lee, "Example-based dialog modeling for practical multi-domain dialog system," in *Speech Communication*, 2009, vol. 51, no. 5, pp. 466–484.

[3] T.-H. Wen *et al.*, "A network-based end-to-end trainable task-oriented dialogue system," in *arXiv preprint arXiv:1604.04562*, 2016.

[4] I. V. Serban, A. Sordoni, Y. Bengio, A. Courville, and J. Pineau, "Building End-To-End Dialogue Systems Using Generative Hierarchical Neural Network Models," in *AAAI,* 2016, pp. 3776–3784.

[5] and H. L. Lifeng Shang, Zhengdong Lu, "Neural re- sponding machine for short-text conversation," in *arXiv preprint arXiv:1503.02364*, 2015.

[6] O. Vinyals and Q. Le, "A Neural Conversational Model," in *arXiv preprint arXiv:1506.05869*, 2015.

[7] Dacher Keltner and Ann M Kring, "Emotion, social function, and psychopathology," *Rev. Gen. Psychol.*, vol. 2, no. 3, pp. 320–342, 1998.

[8] N. Asghar, P. Poupart, X. Jiang, and H. Li, "Deep Active Learning for Dialogue Generation," in *Conference on Empirical Methods in Natural Language Processing*, 2016, vol. 2, no. 4, pp. 1192–1202.

[9] J. R. Finkel, T. Grenager, and C. Manning, "Incorporating non-local information into information extraction systems by Gibbs sampling," in *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics*, 2007, pp. 363–370.

[10] C. Sutton and A. McCallum, "An Introduction to Conditional Random Fields," 2010.

[11] A. Chaudhary, "Machine Learning Techniques for Mobile Intelligent Systems : A Study," 2012.

[12] T.Mitchell, "Machine Learning," in *MIT PRESS*, 1997.

[13] D. A. Permatasari, H. Fakhrurroja and C. Machbub,"Human-Robot Interaction Based On Dialog Management Using Sentence Similarity Comparison Method," International Journal on Advanced Science, Engineering and Information Technology, vol. 10, no. 5

[14] K. Leino, "Reinforcement Learning based Dialog Manager."

[15] J. Gao, M. Galley, and L. Li, "Neural Approaches to Conversational AI," 2018.

17

Journal of Electrical, Electronic, Information, and Communication Technology (JEEICT)
Vol. 3 No. 1, April 2021, Pages 12-17