

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/335975491>

# Hotel Reservation Bot

Technical Report · May 2018

---

CITATIONS

0

---

READS

83

1 author:



Ferdy Mulyadi

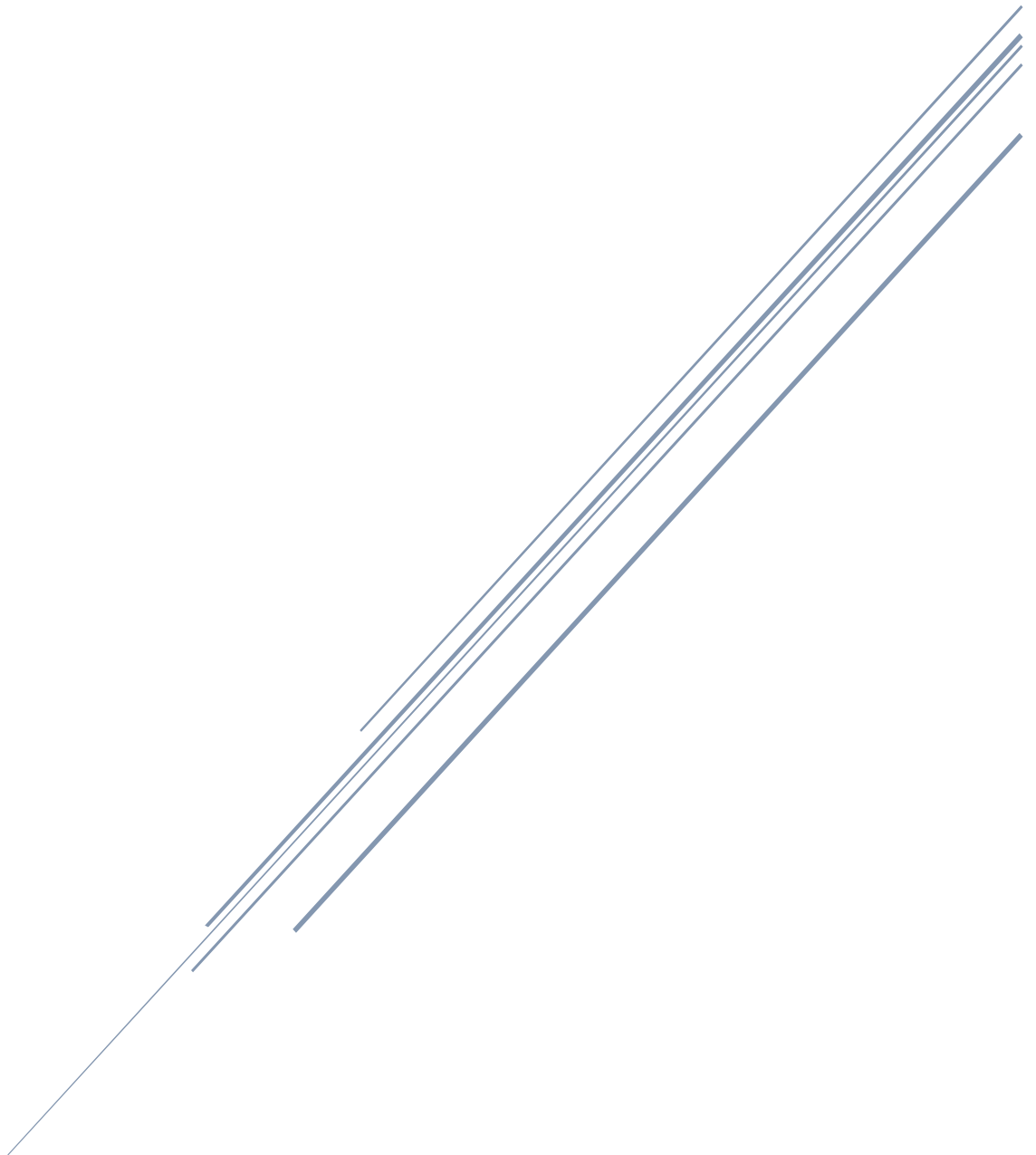
National Electronics and Computer Technology Center

7 PUBLICATIONS 0 CITATIONS

SEE PROFILE

# HOTEL RESERVATION BOT

Ferdy Mulyadi - 60070701701



CPE 641  
Natural Language Processing

# TABLE OF CONTENTS

<b>CHAPTER 1 INTRODUCTION .....</b>	<b>2</b>
<b>CHAPTER 2 LITERATURE REVIEW .....</b>	<b>3</b>
2.1 Chatbot Definition.....	3
2.2 The Concept of Chatbot .....	3
2.3 Chatbot Capabilities .....	4
2.4 POS Tagging .....	5
2.5 Lexical Semantics .....	7
2.6 Levenshtein Distance .....	8
2.7 Regular Expression .....	10
<b>CHAPTER 3 METHODOLOGY .....</b>	<b>12</b>
<b>CHAPTER 4 IMPLEMENTATION AND RESULT .....</b>	<b>14</b>
4.1 Chatbot Specification .....	14
4.2 Conversation Model .....	14
4.3 Result.....	16
<b>CHAPTER 5 CONCLUSION .....</b>	<b>17</b>
<b>REFERENCE.....</b>	<b>18</b>

## **CHAPTER 1 INTRODUCTION**

Chatbot is one of computer program which can be able to do a conversation between computer and human. Nowadays, chatbot is very helpful to do anything, such as service center, reservation, or even informant. For instance, in term of reservation, chatbot will do the reservation such as hotel or travel reservation, thus user do not have to do reservation manually. This is very convenient way, user just message chatbot as same as talking to the reservation staff, and chatbot will do the reservation for the user.

The main part inside chatbot is Natural Language Processing (NLP). NLP is an ability of computer to understand natural (human) language and give the feedback in natural language as well. There are many abilities that computer can do in NLP, for example regular expression, word normalization, word stemming, word tokenization, part-of-speech (POS) tagging, sentiment analysis, and so on.

In this project, one kind of chatbot will be created which is Hotel Reservation Bot. This chatbot will collect the data of user's name, preferred hotel name, check-in and check-out date. This data can be submitted to the hotel booking company (eg. Agoda) and processed. In the end, user will get the confirmation email about hotel reservation. This can be likened for future works. In this current work, only focuses to the making of chatbot.

As aforementioned, this chatbot actually requires a connection at least to one booking company in order to query the best hotel based on the location. However, this project is limited only chatbot creation. Hence, hotel selection will be simulated as Hotel A, Hotel B, and Hotel C. After the conversation, chatbot will generate the user data as the reservation data.

## **CHAPTER 2 LITERATURE REVIEW**

### **2.1 Chatbot Definition**

Chatbot is a computer program which is made in such a way that can understand human language. Chatbots can simulate the human conversation that usually found in call center or service center by using natural language processing to adapt the responses to fit in the following situation [1]. Nowadays, chatbots can be found in e-commerce websites and social messaging services in order to help human if someday they need service that concerned to following websites such as for reservation or call center.

### **2.2 The Concept of Chatbot**

Before the age of chatbot, online users have some frustrated experiences about the online services. Mostly is about service center or call center that has been provided by the online services. From [2], there is a survey that comes from 1000+ participants about the online services that they use today and compare with the past online services (without chatbot technology adoption). The result of the survey can be seen in Figure 2.1.

Based on Figure 1, online users have bad experience mostly is about the sites are hard to navigate. The problem here is about the user-friendly website issue. Some of websites might be hard to navigate because of many features inside that cause some of online users confused about how to navigate the following websites.

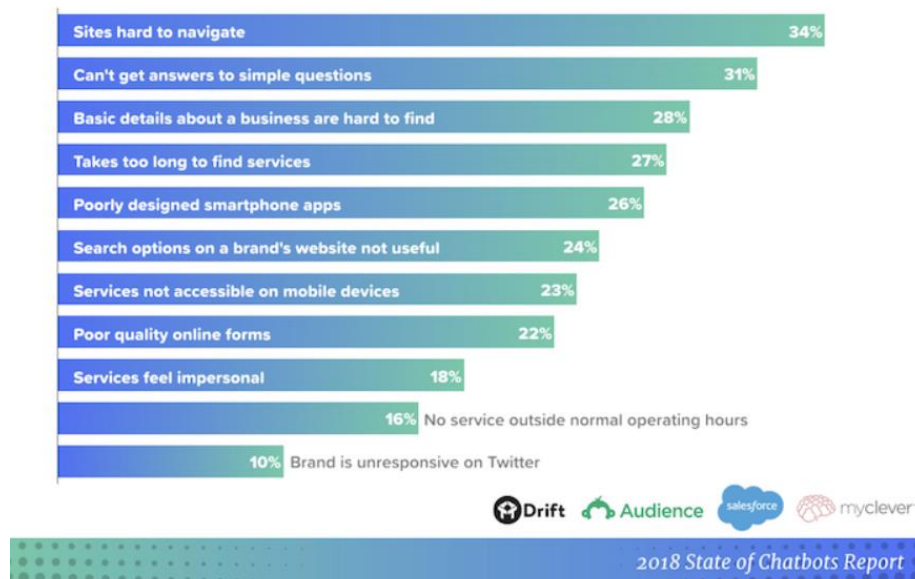
Nowadays, they do not need to confuse about how to navigate the website, let chatbot do it. They just chat with chatbot that adopted by the following website what do they want and chatbot can help to provide what user want just like chat with their friend. For instance, in order to do reservation, users do not need to navigate to the website if they do not want. Users just chat to the chatbot in order to make a reservation or booking, and do not need to navigate by their self anymore.

Moreover, in the real-time world especially in the present time, people would like to be everything easier and one-click navigate. People expect to be able to find the information quickly and easily, otherwise they would get frustrated. This is the part for chatbot to take

place. They do not need to find by their self, just ask to the chatbot and they will get the information what they want. By using this solution, people will get any information easier and comfortable, just like they chat to their friend even it is only chatbot.

## Problems With Traditional Online Experiences

*What frustrations have you experienced in the past month?*



**Figure 2.1** Problem with Traditional Online Experiences [2]

## 2.3 Chatbot Capabilities

All chatbots are made by using natural language processing (NLP) inside. NLP is the main point of chatbot. By using NLP, chatbot can be able to process human language and give feedback according to the input from human or user. Some of NLP capabilities as follows:

- Morphological processing.
- Spell checking.
- Sentiment analysis.
- Parsing.
- Grammar checking.
- Semantic analysis.
- Question answering.
- Machine translation.

In this project, the hotel reservation chatbot has some NLP capabilities in order to process input that comes from the user to get the information from the user. The kind of information is same

as when user input their information regarding to the following reservation questions on the website. Moreover, user do not need to navigate the reservation website even input their information; only use chatbot, chatbot will give some question regarding to the reservation, instead of navigating, user just answer the questions like normal chatting activity. This is one of the chatbot benefits, practical and efficient as well.

Hotel reservation bot has some NLP capabilities in order to process the information that comes from the user. Some of NLP capabilities are:

- Part-of-Speech (POS) tagging.
- Word checking.
- Levenshtein distance.
- Regular expression (*regex*).

The overview of these capabilities will be discussed in the next sub-chapter.

## 2.4 POS Tagging

Part-of-Speech (POS) tagging is an NLP capability which is focus to the label of each word in a sentence. This label used to indicate the part of speech for grammatical categories (noun, verb, tense, etc.) of each token/word in a text corpus/sentence. Table 2.1 is a complete table for POS tagging that comes from [3].

POS Tag	Description	Example
CC	coordinating conjunction	and
CD	cardinal number	1, third
DT	determiner	the
EX	existential there	there is
FW	foreign word	les
IN	preposition, subordinating conjunction	in, of, like
IN/that	that as subordinator	that
JJ	adjective	green
JJR	adjective, comparative	greener
JJS	adjective, superlative	greenest
LS	list marker	1)
MD	modal	could, will
NN	noun, singular or mass	table
NNS	noun plural	tables
NP	proper noun, singular	John
NPS	proper noun, plural	Vikings
PDT	predeterminer	both the boys
POS	possessive ending	friend's

PP	personal pronoun	I, he, it
PP\$	possessive pronoun	my, his
RB	adverb	however, usually, naturally, here, good
RBR	adverb, comparative	better
RBS	adverb, superlative	best
RP	particle	give up
SENT	Sentence-break punctuation	. ! ?
SYM	Symbol	/ [ = *
TO	infinitive 'to'	togo
UH	interjection	uhhuhhuhh
VB	verb be, base form	be
VBD	verb be, past tense	was, were
VBG	verb be, gerund/present participle	being
VCN	verb be, past participle	been
VBP	verb be, sing. present, non-3d	am, are
VBZ	verb be, 3rd person sing. present	is
VH	verb have, base form	have
VHD	verb have, past tense	had
VHG	verb have, gerund/present participle	having
VHN	verb have, past participle	had
VHP	verb have, sing. present, non-3d	have
VHZ	verb have, 3rd person sing. present	has
VV	verb, base form	take
VVD	verb, past tense	took
VVG	verb, gerund/present participle	taking
VVN	verb, past participle	taken
VVP	verb, sing. present, non-3d	take
VVZ	verb, 3rd person sing. present	takes
WDT	wh-determiner	which
WP	wh-pronoun	who, what
WP\$	possessive wh-pronoun	whose
WRB	wh-abverb	where, when
#	#	#
\$	\$	\$
“	Quotation marks	“ “
``	Opening quotation marks	“ “
(	Opening brackets	( {
)	Closing brackets	) }
,	Comma	,
:	Punctuation	– ; : — ...

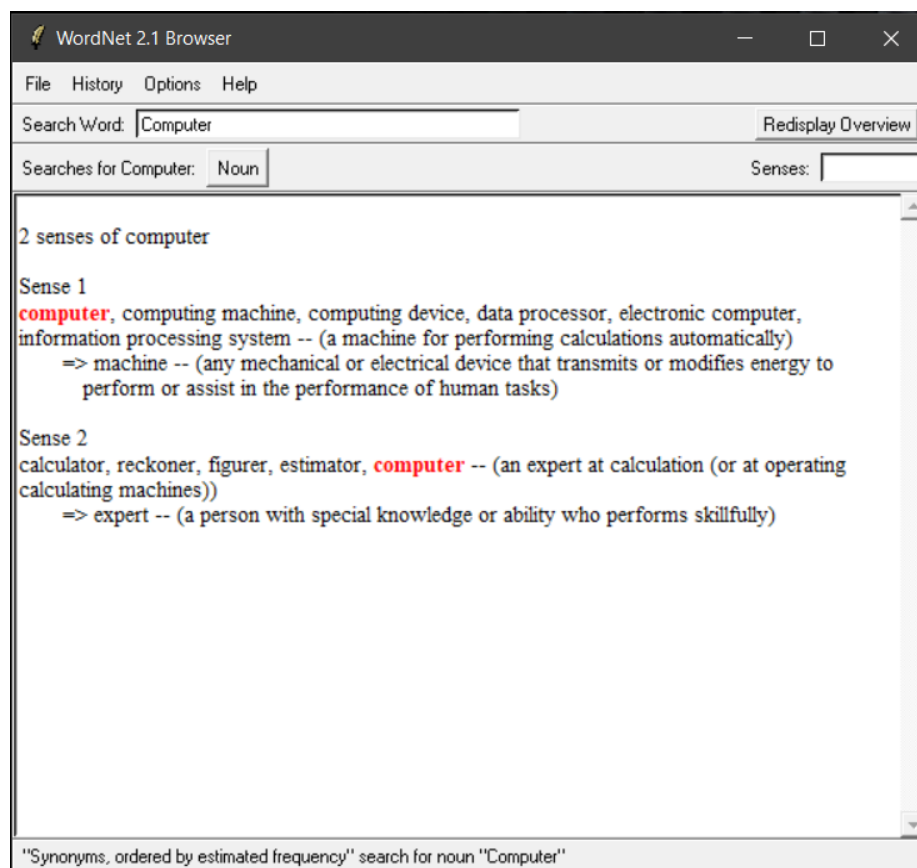
**Table 2.1** POS Tagging Table [3]



## 2.5 Lexical Semantics

Another NLP capability is lexical semantics. The reputable system for lexical semantics is WordNet. WordNet is a system that has a large lexical database of English, including nouns, verbs, adjectives, and adverbs [4]. WordNet is created by Princeton University. This system allows to check every English word including the meaning and the type of the word (noun, verb, or adjective).

The structure among words in WordNet is synonymy. Synonymy is a term to point words that denote the same concept and are interchangeable in many contexts. Each of WordNet 117,000 synsets is linked to other synsets by means of conceptual relations [4]. Figure 2.2 is a graphical user interface of WordNet application.



**Figure 2.2** WordNet 2.1 Browser

## 2.6 Levenshtein Distance

Levenshtein distance or minimal edit distance is an algorithm of string metric for measuring the differences between two sentences or corpus. It named after the Soviet mathematician Vladimir Levenshtein. This algorithm has three basic operations:

- Insertion.
- Deletion.
- Substitution.

There are some aspects that involved in this algorithm:

- For the two comparable strings:
  - o X of length n
  - o Y of length m
- Defining  $D(i,j)$ :
  - o Act as the edit distance between  $X[1..i]$  and  $Y[1..j]$
  - o The first i characters of X and the first j characters of Y
  - o Thus, the edit distance between X and Y is  $D(n,m)$

Here is the Levenshtein distance equation:

- Initialization:

$$D(i,0) = i$$

$$D(0,j) = j$$

- Recurrence Relation:

*For each  $i = 1 \dots m$*

*For each  $j = 1 \dots n$*

$$D(i,j) = \min \left\{ \begin{array}{l} D(i-1,j) + 1 \\ D(i,j-1) + 1 \\ D(i-1,j-1) + 2; \end{array} \right. \left\{ \begin{array}{ll} \text{if } X(i) \neq Y(j) \\ 0; & \text{if } X(i) = Y(j) \end{array} \right.$$

- Termination:

$D(n,m)$  is distance.

For example:

How is the Levenshtein distance to change 'STALL' word to be 'TABLE'?

- The first step is fill the number from 0 – 5 for the initial value as seen from example below.

i	L	5					
	L	4					
	A	3					
	T	2					
	S	1	X				
	#	0	1	2	3	4	5
	#	T	A	B	L	E	
	j						

- Now, how to find X?
  - Firstly, look at  $D(i-1, j)$  which is (#,S), the value is 1, then just add +1. Thus, it becomes 2. This is deletion operation.
  - Secondly, look at  $D(i, j-1)$  which is (T,#), the value is 1, then just add +1. Thus, it becomes 2. This is insertion operation.
  - Thirdly, look at  $D(i-1, j-1)$  which is (#,#), the value is 0, then just add +2 if  $X(i)$  is not equal with  $X(j)$ , otherwise add 0 if  $X(i)$  is equal with  $Y(j)$ . Since  $X(i)$  is 'S' and  $X(j)$  is 'T', +2 will be added. Thus, it becomes 2. This is substitution operation.
  - Last, the minimum cost will be chosen for X. Since all costs have the same value which is 2, the value of X should be 2 and the allowed operations for this point are deletion, insertion, and substitution.

L	5					
L	4					
A	3					
T	2					
S	1	2				
#	0	1	2	3	4	5
	#	T	A	B	L	E

- The complete Levensthein table as seen on below.

<b>L</b>	<b>5</b>	4	3	4	3	<b>4</b>
<b>L</b>	<b>4</b>	3	2	3	2	3
<b>A</b>	<b>3</b>	2	1	2	3	4
<b>T</b>	<b>2</b>	1	2	3	4	5
<b>S</b>	<b>1</b>	2	3	4	5	6
<b>#</b>	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
	<b>#</b>	<b>T</b>	<b>A</b>	<b>B</b>	<b>L</b>	<b>E</b>

- Thus, the minimal edit distance to change from 'STALL' to 'TABLE' is 4.

## 2.7 Regular Expression

The regular expression (*regex*) is a sequence of characters which define a pattern of string searching. This *regex* usually used in algorithm for string searching in order to find a string or for input validation. The person which is behind this is Stephen Cole Kleene who is the inventor of this concept.

The basic concept of *regex* or sometimes called pattern used for specifying test strings. Some examples of regular expression as follows:

- Disjunction (example, [Ww]indow → Window and window).
- Ranges (example, [0-9] → single digit from 0-9)
- Quantification (example, baa+ → There is at least one 'a' after 'ba'; baa, baaa, baaaa).
- Anchors (example, ^[A-Z] → Palo Alto).

Table 2.2 is the complete reference for *regex*.

Character classes	
.	any character except newline
\w \d \s	word, digit, whitespace
\W \D \S	not word, digit, whitespace
[abc]	any of a, b, or c
[^abc]	not a, b, or c
[a-g]	character between a & g
Anchors	
^abc\$	start / end of the string
\b	word boundary
Escaped characters	
\. \* \\	escaped special characters
\t \n \r	tab, linefeed, carriage return
\u00A9	unicode escaped ©

Groups & Lookaround	
(abc)	capture group
\1	backreference to group #1
(?:abc)	non-capturing group
(?=abc)	positive lookahead
(?!abc)	negative lookahead
Quantifiers & Alternation	
a* a+ a?	0 or more, 1 or more, 0 or 1
a{5} a{2,}	exactly five, two or more
a{1,3}	between one & three
a+? a{2,}?	match as few as possible
ab cd	match ab or cd

**Table 2.2** *Regex Table* [<https://www.regexpal.com/>]

## CHAPTER 3 METHODOLOGY

The challenge of this chatbot is how bot can process the answer that comes from the user and get information from it by using NLP capabilities. First, chatbot will ask the question regarding to the reservation, then user will answer the following question. Chatbot will check either this answer is for the following question or not. If yes, chatbot will process the answer and get the information to be saved, otherwise chatbot will ask the same question again.

Some questions which are asked by chatbot to user have own NLP method in order to get the information from the answer. The first method is POS tagging. The purpose for using this method is to determine which word is proper noun (NNP) type. For example, from these questions:

- Bot: “What is your name?”  
User: “My(PRP) name(NN) is(VBZ) John(NNP)”
- Bot: “Would you like to stay at hotel or hostel?”  
User: “I(PRP) would(MD) like(VB) to(TO) stay(VB) at(IN) hotel(NNP)”
- Bot: “Which city you want to stay in?”  
User: “I(PRP) would(MD) like(VB) to(TO) stay(VB) in(IN) Tokyo(NNP)”

As seen from the example, the information that can be obtained from the answer for the following question is NNP type. Thus, this method is very effective if the question is about the noun-type things.

The next method is lexical semantics for location questioning. Chatbot will clarify whether the answer is a location or not. Chatbot will check in the WordNet system. Usually, location will be marked as city or even capital. The disadvantage of this method is, WordNet should be updated in order to show the correct meaning. For instance:

- Bot: “Which city you want to stay in?”
- User: “I would like to stay in Tokyo(NNP)”

According to the WordNet, Tokyo is the capital, capital of Japan – (the capital and largest city of Japan; the economic and cultural center of Japan). Thus, Japan is a location.

The third method is Levenshtein distance. This method is to determine the answer which is for choice question. This method will decide which choice that has been chosen by the user based on the answer. For example:

- Bot: “We have three options for you, which one do you prefer?”
  - o Hotel 1 --- XXX USD
  - o Hotel 2 --- XXX USD
  - o Hotel 3 --- XXX USD
- User: “I prefer hotel 2”

By using Levenshtein distance, the minimal edit distance can be counted between each of choice and user’s answer. Thus, the result as follows:

- Hotel 1 -- I prefer hotel 2 = 10
- Hotel 2 -- I prefer hotel 2 = **9**
- Hotel 3 -- I prefer hotel 2 = 10

From the result, the smallest distance comes from ‘Hotel 2’ which is 9. Hence, the answer ‘I prefer hotel 2’ is directly for ‘Hotel 2’ choice.

The last method that used for this chatbot project is regular expression (*regex*). This method is for the check-in date question. This regular expression will find the string which is date for check-in. For example:

- Bot: “What date for the check-in and how many nights?”
- User: “I think I will stay around 3/31/2018 for 3 nights”

By using this *regex* `(\d+/\d+/\d\d\d\d)`, date can be obtained from the answer and this *regex* `(\d+)(nights?)`, the number before ‘night’ word can be obtained in order to get the number of nights.

These methods are the basic methods in order to build a simple chatbot. More NLP capabilities are needed for the complex chatbot. Some libraries for NLP are provided for each programming language in order to utilize the NLP capabilities. Developers do not need to construct the NLP algorithm for each NLP capability, just initialize the library and use the methods inside it. For example, OpenNLP for Java and TextBlob for Python.

## CHAPTER 4 IMPLEMENTATION AND RESULT

For this project, the chatbot will be written in Java programming language. To be more reliable this chatbot should be connected to the online data or database for querying the accommodation choice especially the price according to the location. However, this project is only focus to the implementation of NLP itself, thus the accommodation choice will be simulated as Hotel 1, Hotel 2, and Hotel 3. The specification and the model of conversation between user and chatbot will be explained in the next sub-chapter.

### 4.1 Chatbot Specification

For this project, the chatbot specifications as follows:

- Chatbot name: Hotel Reservation Bot.
- Programming language: Java programming language.
- Additional NLP tool: Apache OpenNLP, Java WordNet Library, and WordNet.
- NLP capability: Part-of-Speech (POS) tagging, WordNet checking, Levenshtein distances, and regular expression.

### 4.2 Conversation Model

The conversation model will be separated into some parts. Each part has its own kind of conversation. Some parts are:

- Greeting.
- Introduction.
- Reservation.
- Final checking.

Here is the detail of conversation of each part:

- Greeting

User: Hi

Bot: Hello too

Bot: My name is Hotel Bot

Bot: What is your name?



- Introduction

User: My name is Ferdy

Bot: Hello there, Ferdy! :D

Bot: Would you like to stay at hotel or hostel, Ferdy?

- Reservation

User: I would like to stay at hotel

Bot: Which city you want to stay in?

User: I would like to stay at Tokyo

Bot: So, you are going to travel to Tokyo

Bot: We have three options for you, which one do you prefer?

1) Hotel 1 --- XXX USD

2) Hotel 2 --- XXX USD

3) Hotel 3 --- XXX USD

User: I think, I would like to stay at Hotel 2

Bot: What date for the check-in and how many nights?

User: My arrival date is 02/04/2019 and I will stay for 2 nights

Bot: It's done! Your information has been recorded. Please check it

Name: Ferdy

Accommodation: Hotel 2

Check-in: Feb 04 2019

Check-out: Feb 06 2019

###Price: XXX USD###

Bot: Is it correct?

- Final checking

User: Correct

Bot: Thank you very much for using our service, Ferdy

Bot: Please make a payment in our website within 1 hour (Booking ID: XXX) Have a nice day!!

### 4.3 Result

This chatbot's engine is a combination between Java, OpenNLP, and WordNet. Java has some libraries for string process (ex. regular expression), but for the NLP capabilities, the additional libraries have to be added in Java, such as OpenNLP and JWNL for connection between Java and WordNet. The output of this chatbot can be seen below.

```
===== Welcome to hotel/hostel reservation application =====
===== Created by: Ferdy Mulyadi 60070701701 =====
===== Please say 'Hi' to Hotel Bot to call the bot =====

User: Hello
Bot: Howdy?
Bot: My name is Hotel Bot
Bot: What is your name?

User: My name is Ferdy
Bot: Hello there, Ferdy! :D
Bot: Would you like to stay at hotel or hostel, Ferdy?

User: I would like to stay at hotel
Bot: Which city you want to stay in?

User: I will travel to Bangkok
Bot: So, you are going to travel to Bangkok
Bot: We have three options for you, which one do you prefer?
1) Hotel 1 --- XXX USD
2) Hotel 2 --- XXX USD
3) Hotel 3 --- XXX USD

User: I think hotel 2 is suitable for me
Bot: What date for the check-in and how many nights? (For example, maybe 3/31/2019 and 5 nights)

User: I arrive around 2/18/2019 and i will take 3 nights
Bot: It's done! Your information has been recorded. Please check it
Name: Ferdy
Accommodation: Hotel 2
Check-in: Feb 18 2019
Check-out: Feb 21 2019
###Price: XXX USD###
Bot: Is it correct?

User: Yes, it's correct. Thank you very much
Bot: Thank you very much for using our service, Ferdy
Bot: Please make a payment in our website within 1 hour (Booking ID: 1526620441429)
Have a nice day!!
```

## **CHAPTER 5 CONCLUSION**

Chatbots is an effective way if implemented carefully. Chatbots are the forms of automation and they would replace human agents because chatbots are really fast and never get tired compared to human agents can get fatigued or overwhelmed. Thus, many people try to make chatbot for their company for customer service purpose in order to serve their customers 24 hours with zero mistake, while human agents cannot do. The most important is chatbots make it easier to get closer to customers and they will serve the customers with fast response.

From this simple hotel reservation bot, the required NLP capabilities are less than the complex one. Complex chatbot need more NLP capabilities in order to make it reliable and works as same as human agent. For future works, this hotel reservation bot should be able to connect to online data in order to query the name and price of accommodation according to the location.

## REFERENCE

- [1] “The Complete Guide to Chatbots in 2018,” Sprout Social. [Online]. Available: Sprout Social, <https://sproutsocial.com/insights/topics/chatbots/> [Accessed: 16 May 2018].
- [2] E. Devaney, “The 2018 State of Chatbots Report: How Chatbots are Reshaping Online Experiences,” Drift, 23 Jan 2018. [Online]. Available: Drift, <https://blog.drift.com/chatbots-report/> [Accessed: 16 May 2018].
- [3] M. Marcus, B. Santorini and M.A. Marcinkiewicz. “Building a large annotated corpus of English: The Penn Treebank.” *Computational Linguistics*, vol. 19, no. 2, 1993, pp. 313–330.
- [4] C. Fellbaum. “WordNet and wordnets,” in *Encyclopedia of Language and Linguistics*, 2nd ed. K. Brown, et al., Oxford: Elsevier, January 2005, pp. 665-670.