Journal of
Information Systems Engineering
and Business Intelligence

Vol.6, No.2, October 2020
Available online at: http://e-journal.unair.ac.id/index.php/JISEBI

Smart Dissemination by Using Natural Language Processing Technology

Tora Fahrudin[1)*], Kastaman[2)], Sherin Nadya Meideni[3)], Padma Edhitya Chairunnisafa Priyono[4)], Muhammad Galang Fathirkina[5)], Samira[6)]

[1)2)3)4)5)6)]*Telkom University, Indonesia*

Jl. Telekomunikasi No. 1 Dayeuhkolot, Bandung

[1)]torafahrudin@telkomuniversity.ac.id, [2)]kastaman@telkomuniversity.ac.id, [3)]sherinadyaaa@student.telkomuniversity.ac.id,
[4)]padmaecp@student.telkomuniversity.ac.id, [5)]fathirkinagalg@student.telkomuniversity.ac.id, [6)]samira@student.telkomuniversity.ac.id

Article history:

Received 5 June 2020
Revised 8 September 2020
Accepted 21 September 2020
Available online 28 October 2020

Keywords:

Dynamic filter
Dynamic content
Natural language processing
Smart dissemination
Whatsapp

Abstract

Background: Recently, WhatsApp has become the world's most popular text and voice messaging application with 1.5 billion users. A lot of WhatsApp Application Programming Interface (API) has also been established to be connected to other applications. On the other hand, the development of natural language processing (NLP) for WhatsApp messages has snowballed. There are extensive studies on the dissemination information using WhatsApp but the study on NLP involving data from WhatsApp is lacking.
Objective: This study aims to implement NLP in smart dissemination applications by using WhatsApp API.
Methods: We build a framework that embeds an intelligent system based on the NLP in WhatsApp API to disseminate a dynamic message. Some of the sentences are used to evaluate the accuracy of this application.
Results: Smart dissemination consists of dynamic filter and dynamic content. Dynamic filter was conducted by using the POS tagger and clause statement. Meanwhile, dynamic content was built by using the replace MySQL function. There are twofold limitation: the application could not transform a message that matches rule <3> with conjunction "dan"; has the same attribute before and after <CC> tag; and the maximum of the logical operator is one type for coordinating conjunction (AND/OR) in one sentence.
Conclusion: Our framework can be used for dynamic dissemination of messages using dynamic message content and dynamic message recipient with an accuracy of 95% from twenty sample messages.

I. INTRODUCTION

WhatsApp is a pun of the phrase "What's up?" [1]. It is an alternative messaging service to Short Message Services (SMS) with many benefits such as low-cost [2]; supporting group conversation [3]; supporting rich multimodal medium of communication (text, photos, videos, emoticon, documents and also location) [4]; and more secure with end-to-end encryption [5]. Recently WhatsApp has dominated the messaging app market with over 1.5 billion monthly users worldwide [6].

An Application Programming Interface (API) provides a programmatic interface to a software component or service [7]. A lot of WhatsApp API has also been established to be connected to other applications. Wablas is one example of a WhatsApp API gateway service for sending and receiving messages, notification, scheduler, reminder, and tracking with a simple integration system to another service [8]. By using Wablas, we can broadcast messages to a particular receiver quickly and easily. There are two programing languages that can support Wablas: PHP and Python.

On the other hand, the development of natural language processing (NLP) to process WhatsApp messages has developed rapidly, including a Chabot in Indonesian university admission [9]; WhatsApp auto replay Chabot in English language [10]; WhatsApp bot for the COVID-19 statistical services data [11]; and professional Chabot

[*] Corresponding author

doi: http://dx.doi.org/10.20473/jisebi.6.2.133-142

application to find inappropriate terms in messages [12]. However, the development of NLP to disseminate study-program information among university students has not been conducted.

This study will implement Part-of-Speech tagger (POS tagger) method in NLP to extract parts of speech of each word, such as noun, verb, adjective, etc. Some POS tagger in Indonesian language has been conducted in a rule-based model, a probabilistic model, hidden markov model and neural network model [13] such as INA-NLP POS tagger [14], POS tag Indonesia [15], deep neural network POS tagger [16], etc. Some research used POS tagger in Indonesian language directly such as document subjectivity and target detection in opinion mining [17] to extract public opinions [18]; modified opinion mining rules to extract multilabel students complaints [19]; and also Indonesian POS tagging system for computer-aided independent language learning [20].

The current research aims to use POS tagger result to identify students' attributes in the dynamic filtering of WhatsApp messages. To get the attribute list and send a dynamic message to students, metadata of student attributes are combined with the replace function in MySQL. Wablas API, MySQL database, Python, Apache, and PHP are used to implement the smart dissemination applications. The objective of this research is twofold: 1) implementing the POS tagger module as a part of NLP technology to support a dynamic filter process; and 2) building a prototype application for disseminating WhatsApp messages (which may include dynamic message) to particular recipients (which may include the dynamic ones). This research is useful as a basis for developing information dissemination applications with dynamic content and objects such as information dissemination within an organization i.e. from management to employees.

The remainder of this paper is divided into four sections. In Section 2, we provide a brief review of the method. In Section 3, we show the experimental study and simulation results. In Section 4, we discuss the results. Finally, in Section 5, we conclude the paper with some future research directions.

II. Methods

In general, the system consists of four parts of processes, which can be seen in Fig. 1. The first one is the WhatsApp messages receiver. First, we use Wablas API to forward the incoming messages that have been received by using a webhook URL. Second, we save to database. After that, the messages can be processed by another module. The third process consists of a filter and a content module. A filter means how we can extract the objects from the messages, namely to whom the message is sent. Content means what types of message being sent, static or dynamic. A static message means that the message sent to all receivers contain the same words. A dynamic content means the message sent to students contain a different message from one receiver to another. Lastly, the fourth process is sending out the message by using Wablas API.

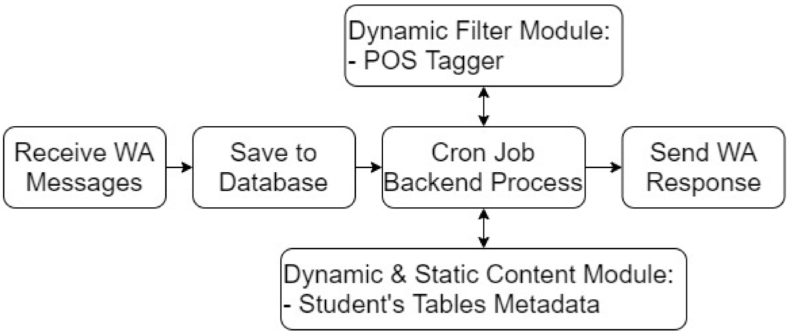
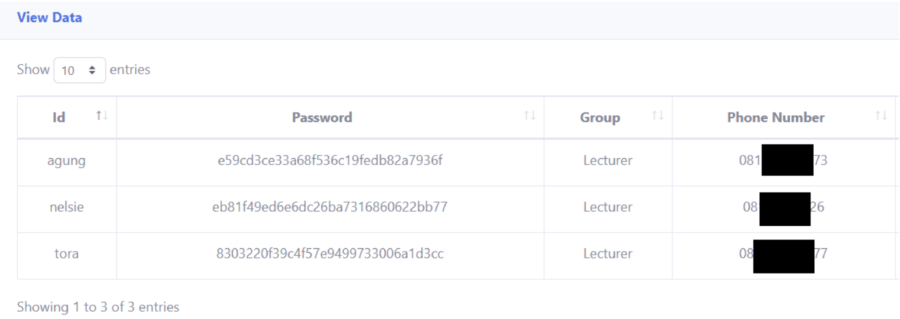


Fig. 1 The Proposed Method

A. Receive WA Messages

The dissemination process starts when a user sends a WhatsApp message by using a dissemination application number. The user who was registered as an administrator to the application (as shown in Fig. 2) can send a broadcast message to students. Each user can have more than one number registered to the application. The broadcast message will be identified by the system by keyword "B <message>".



View Data

Show 10 entries

Id	Password	Group	Phone Number
agung	e59cd3ce33a68f536c19fedb82a7936f	Lecturer	081[redacted]73
nelsie	eb81f49ed6e6dc26ba7316860622bb77	Lecturer	08[redacted]26
tora	8303220f39c4f57e9499733006a1d3cc	Lecturer	08[redacted]77

Showing 1 to 3 of 3 entries

Fig. 2 Users' registered Phone Number

B. *Save To Database*

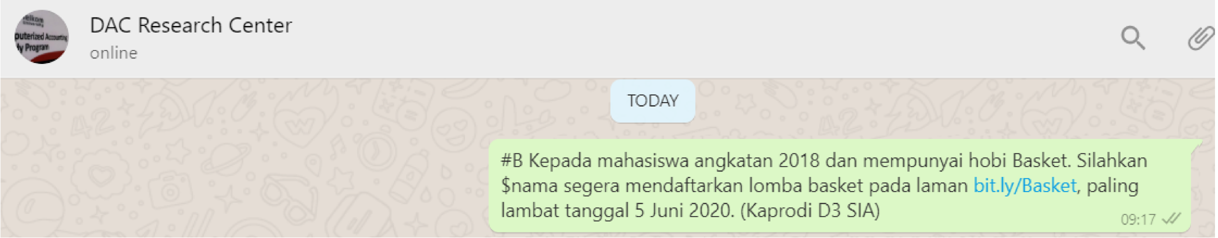
The broadcast message sent by users will be saved in the database with status 0. Another module will process that broadcasted message. That module will read the broadcast message periodically by using cronjob facilities.

C. *Backend Process (Dynamic Filter Module and Dynamic Content Module)*

The objective of the backend process is to receive a broadcast message from the user, identify and replace keyword in a message with a dynamic content and lastly filter the recipients. Those processes will be executed periodically by a cronjob facility. Dummy data set was used to test the backend process for the dynamic filter module and the dynamic content module.

By using a cronjob facility, a message will be checked if it has been read and proceeded or not by the system. This process is repeated every 1 minute. After the message is read by the system, the status of the message will be set to 1.

After a message is read by the system, then the object and content of the message will be identified. The system must decide who the object of that message is, either every student or a specific student. Keyword "Kepada" is used to simplify the object identification process. Fig. 3 shows a sample of a message which contains keyword "Kepada", which means the message will be sent only to the specific students whose GPA >= 3 and < 3.9. Another message does not have a specific object (does not contain keyword "Kepada"), which means that it will be sent to all of the students. The illustration is provided in Fig. 4.

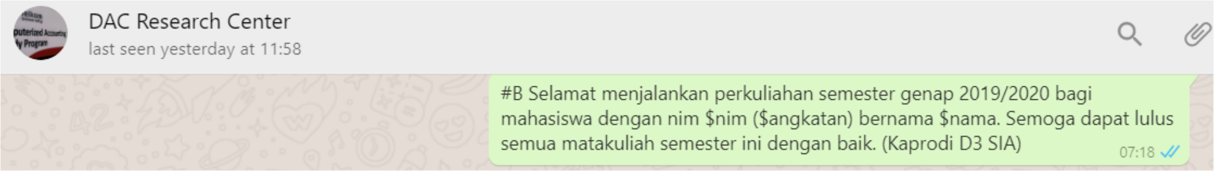


DAC Research Center
online

TODAY

#B Kepada mahasiswa angkatan 2018 dan mempunyai hobi Basket. Silahkan $nama segera mendaftarkan lomba basket pada laman bit.ly/Basket, paling lambat tanggal 5 Juni 2020. (Kaprodi D3 SIA) 09:17

Fig. 3 Sample of a message with a specific object keyword

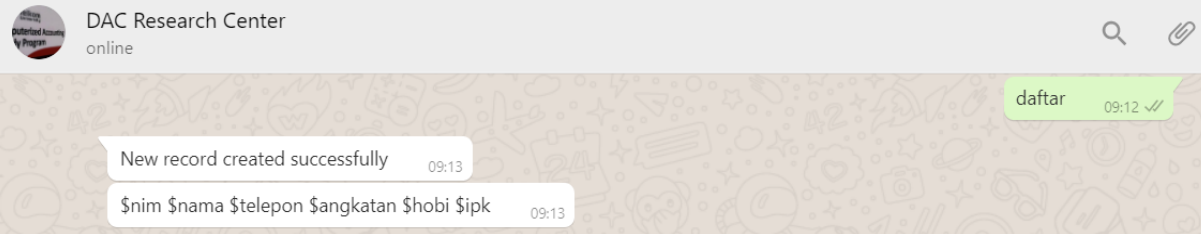


DAC Research Center
last seen yesterday at 11:58

#B Selamat menjalankan perkuliahan semester genap 2019/2020 bagi mahasiswa dengan nim $nim ($angkatan) bernama $nama. Semoga dapat lulus semua matakuliah semester ini dengan baik. (Kaprodi D3 SIA) 07:18

Fig. 4 Sample of a message without a specific object keyword

The object identification process by using a dynamic filter module is built by using a POS tagger function in NLP. By using POS tagger, we can extract part of speech tag for every single or multi-word token. We use nlp-id [21], a Python library that provides various functions for NLP for Indonesian language. Detailed Indonesian POS tagging list can be seen in [22], [17] and [23]. POS tagger result of a broadcast message as shown in Fig. 3 can be seen in Table I. The dynamic filter module extracts the student attributes by using POS tagger result, whose noun phrases is a match to our student attributes. The attribute of student data can be achieved by using "daftar" command which can be seen in Fig. 5.



daftar 09:12

New record created successfully 09:13

$nim $nama $telepon $angkatan $hobi $ipk 09:13

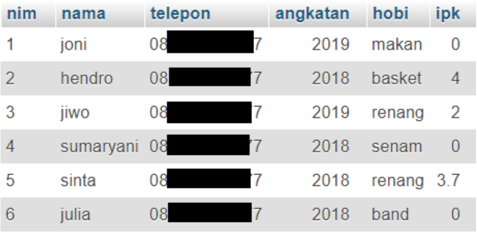
Fig. 5 List of an attribute in student attributes

The last process in the dynamic filter module is to create a filter condition in "where clause" SQL statement to get a specific student who matches with the desired criteria. Therefore, the where clause SQL will be "WHERE Angkatan IN (2018) and hobi IN ('Basket')".

TABLE 1
PART OF SPEECH OF BROADCAST MESSAGE IN FIG. 3

Word	Tag	Description	Student Attributes
kepada <to>	IN	Preposition	-
mahasiswa <student>	NN	Noun	-
angkatan <degree>	NN	Noun	angkatan
2018	NUM	Number	-
dan <and>	CC	Coordinating Conjunction	-
mempunyai <have>	VP	Verb Phrase	-
hobi <hobby>	NN	Noun	hobi
basket <basketball>	NN	Noun	-
silahkan <please>	VP	Verb Phrase	-
$nama <$name>	NN	Noun	nama
segera <immediately>	ADV	Adverbs	-
mendaftarkan <register>	VP	Verb Phrase	-
lomba <contest>	NN	Noun	-
basket <basketball>	NN	Noun	-
pada <on>	IN	Preposition	-
laman <the page>	NN	Noun	-
bit.ly/Basket	NN	Noun	-
paling lambat <no later than>	ADJP	Adjective Phrase	-
tanggal <date>	NN	Noun	-
5	NUM	Number	-
juni <june>	VP	Verb Phrase	-
2020	NUM	Number	-

The dynamic content module replaces all student attribute keywords with value for each student dynamically. To distinguish static and dynamic content, a user should use character "$" that is followed by attribute names in the database like shown in Fig. 6 and Table 2. Table 2 shows an example of a broadcasted dynamic and static content message. We use REPLACE function in MySQL to replace the attributes keyword with appropriate values of each student's data.



nim	nama	telepon	angkatan	hobi	ipk
1	joni	08[redacted]7	2019	makan	0
2	hendro	08[redacted]7	2018	basket	4
3	jiwo	08[redacted]7	2019	renang	2
4	sumaryani	08[redacted]7	2018	senam	0
5	sinta	08[redacted]7	2018	renang	3.7
6	julia	08[redacted]7	2018	band	0

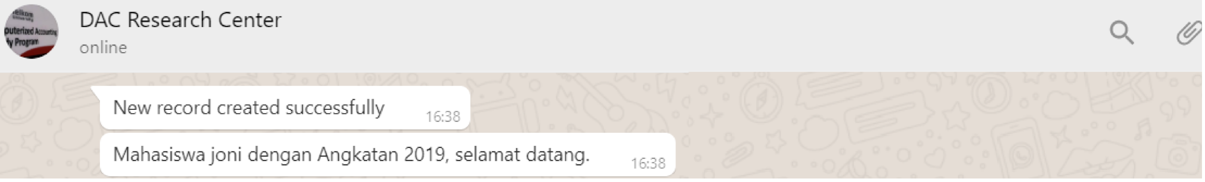
Fig. 6 Students attribute and data

TABLE 2
BROADCAST MESSAGE FOR EACH STUDENT

Content	Message	Joni	….	Julia
Dynamic	Mahasiswa $nama dengan Angkatan $angkatan, selamat datang. <Student $name with has degree $degree, welcome>	Mahasiswa joni dengan Angkatan 2019, selamat datang. <Student joni with has degree 2019, welcome>	…	Mahasiswa julia dengan Angkatan 2018, selamat datang <Student julia with has degree 2018, welcome>
Static	Selamat menjalankan perkuliahan semester Genap 2019/2020. <Welcome to 2019/2020 even course >	Selamat menjalankan perkuliahan semester Genap 2019/2020. <Welcome to 2019/2020 even course >	…	Selamat menjalankan perkuliahan semester Genap 2019/2020. <Welcome to 2019/2020 even course >

D. Send WA Response

The broadcast message which has been proceeded by using a dynamic filter and dynamic content module will be sent to an appropriate student. Wablas API sends the message one by one. Fig. 7 shows an example of broadcasted messages received by Joni.



New record created successfully 16:38

Mahasiswa joni dengan Angkatan 2019, selamat datang. 16:38

Fig. 7 One of a broadcast message which was received by student Joni

III. RESULTS

A. Result of Dynamic Filter Module

Because the dynamic filter module is created by using "where clause" SQL statement, we test the application by using operators in the where clause [24], as shown in Table 3. There are two operator in "where clause" SQL statement, logical operator such as "AND" "OR" and comparison operator such as "=", "<", ">", "<=", ">=".

B. System Architecture

Fig. 8 shows the smart dissemination application architecture. It contains Wablas API, web server, a mobile, a laptop and the Internet connection. Wablas is the API service provider to receive and sent broadcast messages. The web server is for registering the lecturers' phone numbers. A mobile and laptop are the devices for accessing the applications. From thirty attempts, the performance of delivering a message from this system architecture is achieved on average of 2 minutes.

TABLE 3
DYNAMIC FILTER TESTED

Operator	Keyword	Filter Message	Where Clause Result	Status
Logical	AND <DAN>	Kepada Angkatan 2019 dan mempunyai Hobi Renang.	WHERE angkatan IN ('2019') AND hobi IN ('renang')	Ok
Logical	AND <DAN>	Kepada Angkatan 2018;2019;2020 dan mempunyai Hobi Renang.	WHERE angkatan IN ('2019','2018','2020') AND hobi IN ('renang')	Ok
Logical	OR <ATAU>	Kepada Mahasiswa dengan ipk > 3.5 atau mempunyai Hobi Renang.	WHERE ipk >3.5 OR hobi IN ('renang')	Ok
Logical	OR <ATAU>	Kepada Mahasiswa dengan ipk > 3.5 atau mempunyai Hobi Renang;Basket;Senam	WHERE ipk >3.5 OR hobi IN ('renang','basket','senam')	Ok
Comparison	=	Kepada mahasiswa dengan ipk = 4.	WHERE ipk =4	Ok
Comparison	<	Kepada mahasiswa dengan ipk < 2.	WHERE ipk <2	Ok
Comparison	>	Kepada mahasiswa dengan ipk > 3.	WHERE ipk >3	Ok
Comparison	<=	Kepada mahasiswa dengan ipk <= 1.5.	WHERE ipk <1.5	Ok
Comparison	>=	Kepada mahasiswa dengan ipk >= 3.	WHERE ipk >=3	Ok

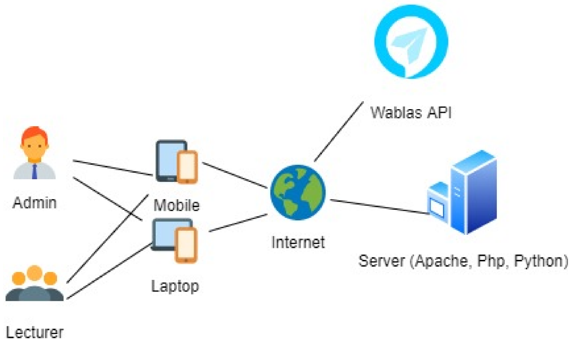
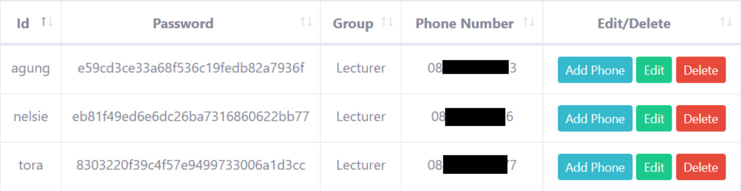


Fig. 8 System Architecture

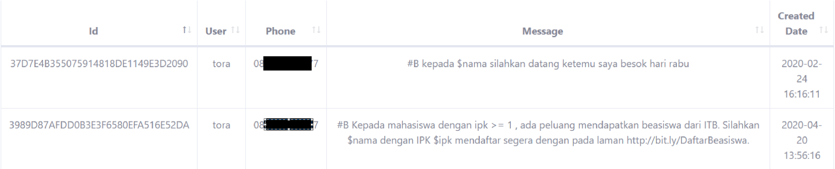
C. Application

There are two kinds of applications: web-based and WhatsApp. The web-based application is used to register the lecturers' phone numbers to the system, which is necessary to be able to broadcast messages to students. Fig. 9 shows the lecturers' phone number, which are registered as an administrator (admin) into the application. One lecturer may have more than one phone number registered to the system. The history of every broadcast message can be seen in "View Sent Messages" menu like shown in Fig. 10.



Id ↑↓	Password ↑↓	Group ↑↓	Phone Number ↑↓	Edit/Delete ↑↓
agung	e59cd3ce33a68f536c19fedb82a7936f	Lecturer	08[redacted]3	Add Phone Edit Delete
nelsie	eb81f49ed6e6dc26ba7316860622bb77	Lecturer	08[redacted]6	Add Phone Edit Delete
tora	8303220f39c4f57e9499733006a1d3cc	Lecturer	08[redacted]7	Add Phone Edit Delete

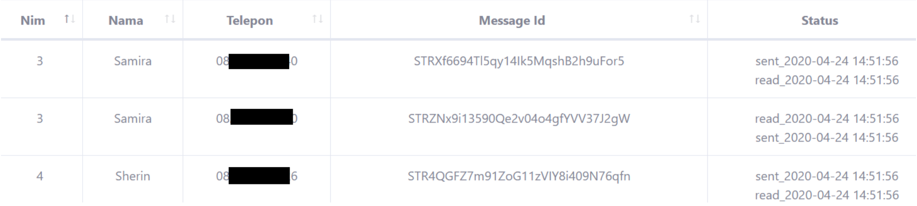
Fig. 9 Menu to view lecturers' phone numbers



Id	User	Phone	Message	Created Date
37D7E4B355075914818DE1149E3D2090	tora	08[redacted]7	#B kepada $nama silahkan datang ketemu saya besok hari rabu	2020-02-24 16:16:11
3989D87AFDD0B3E3F6580EFA516E52DA	tora	08[redacted]7	#B Kepada mahasiswa dengan ipk >= 1 , ada peluang mendapatkan beasiswa dari ITB. Silahkan $nama dengan IPK $ipk mendaftar segera dengan pada laman http://bit.ly/DaftarBeasiswa.	2020-04-20 13:56:16

Fig. 10 Menu to view broadcasted messages history

Tracking each message can be seen in "View Tracking Messages" menu, as shown in Fig. 11. By using the tracking messages status module of Wablas API, we can see the detailed status for each student.



Nim	Nama	Telepon	Message Id	Status
3	Samira	08[redacted]0	STRXf6694Tl5qy14Ik5MqshB2h9uFor5	sent_2020-04-24 14:51:56 read_2020-04-24 14:51:56
3	Samira	08[redacted]0	STRZNx9i13590Qe2v04o4gfYVV37J2gW	read_2020-04-24 14:51:56 sent_2020-04-24 14:51:56
4	Sherin	08[redacted]6	STR4QGFZ7m91ZoG11zVIY8i409N76qfn	sent_2020-04-24 14:51:56 read_2020-04-24 14:51:56

Fig. 11 Menu to view broadcast messages status for each student

TABLE 4
SAMPLE MESSAGES

Number	Message	Where Clause Result	Status
1	Kepada nim 1;2;3 dan hobi makan;renang.	WHERE nim IN ('1','2','3') AND hobi IN ('makan','renang')	True Positive
2	Kepada nim 1;2;3 atau hobi makan;renang.	WHERE nim IN ('1','2','3') OR hobi IN ('makan','renang')	True Positive
3	Kepada nim > 2 dan nim < 3 dan hobi renang.	WHERE nim > '2' AND nim < '3' AND hobi IN ('renang')	True Positive
4	Kepada Angkatan > 2019 dan ipk > 3 dan ipk <= 4.	WHERE angkatan > '2019' AND ipk >3 AND ipk <=4	True Positive
5	Kepada Angkatan 2017;2018;2020 atau nama joni;hendro;jiwo.	WHERE angkatan IN ('2017','2018','2020') OR nama IN ('joni','hendro','jiwo')	True Positive
6	Kepada Angkatan 2017;2018;2020 dan nama joni;hendro;jiwo.	WHERE angkatan IN ('2017','2018','2020') AND nama IN ('joni','hendro','jiwo')	True Positive
7	Kepada siswa dengan telepon 08123;0812321;8123123.	WHERE telepon IN ('08123','08321','08123123')	True Positive
8	Kepada siswa dengan telepon 091823123.	WHERE telepon IN ('091823123')	True Positive
9	Kepada siswa dengan telepon 091823123 atau 0812312312 atau 3128319.	WHERE telepon IN ('091823123','0812312312','3128319')	True Positive
10	Kepada siswa dengan nama joni.	WHERE nama IN ('joni')	True Positive
11	Kepada siswa dengan nama joni;hendro.	WHERE nama IN ('joni','hendro')	True Positive
12	Kepada siswa dengan nama joni atau hendro.	WHERE nama IN ('joni','hendro')	True Positive
13	Kepada siswa dengan nama joni atau hendro atau ipk > 3.5	WHERE nama IN ('joni','hendro') OR ipk >3.5	True Positive
14	Kepada Angkatan 2018 dan Angkatan 2019.	WHERE angkatan IN ('2018') AND angkatan IN ('2019')	False Negative
15	Kepada Angkatan 2018 dan 2019.	WHERE angkatan IN ('2018',’2019’)	True Positive
16	Kepada Angkatan 2018 atau Angkatan 2019.	WHERE angkatan IN ('2018') OR angkatan IN ('2019')	True Positive
17	Kepada Angkatan 2018 atau 2019.	WHERE angkatan IN ('2018','2019')	True Positive
18	Kepada Angkatan 2019 dan ipk < 3 dan ipk > 4.	WHERE angkatan IN ('2019') AND ipk <3 AND ipk >4	True Postive
19	Kepada Angkatan 2019 dan ipk > 3.5 dan hobi renang;senam dan basket.	WHERE angkatan IN ('2019','basket') AND ipk >3.5 AND hobi IN ('renang','basket','senam')	True Postive
20	Kepada Angkatan 2019 dan ipk > 3.5 dan hobi renang;senam;basket dan nim 2.	WHERE angkatan IN ('2019') AND ipk >3.5 AND hobi IN ('renang','senam','basket') AND nim IN ('2')	True Postive

D. Testing Results

To measure the effectiveness of our application, especially how the dynamic filter module can transform the message to "where clause" result, we create twenty sample messages that contain a combination of a number of attributes, a value of each attribute, and also a logical and comparison operator at Table 4. Recall, precision, and accuracy metrics are used to measure the effectiveness. True positive (TP) means the transformation of "where clause" of that message is correct and the receivers are appropriate. False positive (FP) means the transformation of "where clause" of that message is incorrect and the receivers are appropriate. True negative (TN) means the transformation of "where clause" of that message is incorrect and the receivers are inappropriate. False negative (FN) means the transformation of "where clause" of that message is correct and the receivers are inappropriate.

TABLE 5
PRECISION, RECALL AND ACCURACY RESULTS

	Positive (Transformation of 'Where Clause' is correct)	Negative (Transformation of 'Where Clause' is incorrect)
Positive (Receiver is appropriate)	19	0
Negative (Receiver is inappropriate)	1	0

From Table 5, we get recall (+) value of 0.95, precision (+) value of 1, and also accuracy value of 0.95. We found error in "Kepada Angkatan 2018 dan Angkatan 2019" message. That error is because the sender wants to broadcast message to all students with student entry years of ('2018','2019'), but because the message uses conjunction "dan" and mention the name of attribute again after the conjunction, the filter process will add conjunction "AND" which make the "where clause" result get "WHERE angkatan IN ('2018') AND angkatan IN ('2019')". We classify that error to be FN because the transformation of "where clause" of that message is correct but the receivers are inappropriate.

Four combination rules for dynamic filter processing as shown in Table 6 are derived from twenty sample messages from Table 4. Symbol <ATTRIB> denotes the list of student attributes. <LIST OF VALUES> denotes the list of values for attribute <ATTRIB>. <CC> denotes coordinating conjunction, which can contain "AND <DAN>" or "OR <ATAU>".

TABLE 6
DYNAMIC FILTER RULES

Rules Code	Rules	Message Number
<1>	<ATTRIB> <LIST OF VALUES>	7, 8, 10, 11
<2>	<1> <CC> <LIST OF VALUES>	9, 12, 15, 17
<3>	<1> <CC> <1>	1, 2, 3, 4, 5, 6, 14, 16, 18, 19, 20
<4>	<2> <CC> <1>	13

IV. DISCUSSION

We have created a smart dissemination application by using a NLP technology. We use a POS tagger function to extract the tag label for each word and process it to get the "where clause" SQL statement. This SQL statement is needed to build a dynamic filter of the messages before sending them to students. Lastly, the replace function is also used to replace the dynamic keyword in the messages by the appropriate value for each student.

Some sentences have been tested to check the "where clause" SQL statement scenario results based on logical and comparison operators. From the tested results in Table 3, we have found that a dynamic filtering module is successful in some scenarios. The limitation of our study here is twofold. The first is the maximum of the logical operator is one type for Coordinating Conjunction (AND/OR) i.e atau in sentences "Kepada Mahasiswa dengan IPK > 3.5 atau mempunyai Hobi Renang; Basket; Senam atau mempunyai angkatan 2019; 2018." (To students with GPA> 3.5 or have a swimming hobby; Basketball; Gymnastics or have a class of 2019; 2018.). A sentence that has more than one Coordinating Conjunction i.e "Kepada Mahasiswa dengan IPK > 3.5 atau mempunyai Hobi Renang; Basket; Senam dan mempunyai angkatan 2019; 2018." (To students with GPA> 3.5 or have a swimming hobby; Basketball; Gymnastics and have a class of 2019; 2018.) cannot be proceeded by our systems. The second is the message that uses conjunction "dan" and mentions the name of the attribute again after the conjunction. To deal with this problem, a semantic approach can be used. This will be considered as our future works.

With our systems, a user can send a message to a dissemination center and the system will automatically send the broadcast message to the student who meets the relevant criteria. A lecturer can track the sending message status for each student by loggin in to the web-based application. A user can only send a broadcast message only if registered as an adminisrator.

Our smart dissemination applications can be implemented not only for academic purposes but also for any situations that require dissemination of static or dynamic information from an authorized sender such as when: (1) A gead of a housing complex (*rukun tetangga*) needs to send a message to the people in the neighborhood. For example: A citizen aged 40 and above is suggested to exercise every week; (2) A division head wants to encourage employees with a salary of less than 5 million and lives in Bandung to take the opportunity as a marketing at PT XYZ on Saturday; (3) A committee member wants to send a message to all participants from Jogjakarta majoring in science to enroll to Jogja Data Sciences Course before 27 August 2020.

TABLE 7
MESSAGE EXAMPLES IN OTHER CASE

Case	Message	Where Clause Result	Status
1	Kepada Warga umur > 40, silahkan pergunakan hari minggu untuk olahraga.	WHERE umur > 40	Ok
2	Kepada karyawan dengan gaji < 5000000 dan asal bandung. Ada tawaran sebagai tenaga marketing pada hari sabtu dari PT XYZ.	WHERE gaji < 5000000 AND asal IN ('bandung')	Ok
2	Kepada peserta asal jogja dan jurusan ipa. Silahkan mendaftarkan sebagai peserta kursus Jogja Data Science sebelum 27 Agustus 2020.	WHERE asal IN ('jogja') AND jurusan IN ('ipa')	Ok

Although those twenty samples in Table 4 and another sample in Table 7 contain a combination of some attributes, value of each attribute, and also logical and comparison operator, the system can cover the combinations. Our proposed method has a limitation because it can only cover four rules as mentioned in Table 6. We believe that the other message similar our rules in Table 6 may give the same accuracy results compared to our experiment results.

We can see that sample message data in Table 4 which fall into FN is 1 and for FP and TN are 0. There are some reasons. First, the FN case is rare, because people usually write the message "Kepada Mahasiswa Angkatan 2019 dan 2020 dan 2021" not "Kepada Mahasiswa Angkatan 2019 dan Angkatan 2020 dan Angkatan 2021" (they do not repeat "Angkatan"). In our opinion, the number of sample data which represent FP message is still relevant. Second, to the best of our knowledge, there are no sample messages that match the FP criteria because of the incorrect "where clause" result could not make the query result become true and make the database server give the correct answer. Third, one of the examples of TN message is "Kepada peserta asal jogja dan berasal dari IPA. Silahkan mendaftarkan sebagai peserta Jogja Data Science!". That message will be transformed by server to "WHERE asal IN ('jogja','berasal')". That incorrect “where clause” result is caused by the attribute "berasal" is not defined in the database, so that makes the receivers not appropriate.

So, in our opinion, those twenty sample messages already represented the actual message that contain less sample of FN, zero FP, and also zero TN. Eventhough we add more samples that meet the TN criteria and our rules, the accuracy, precision and the recall would be the same. That is because the wrong message will fall into the wrong "where clause" result and makes the receivers not appropriate (TN criteria), and TN criteria did not affect the accuracy because the formula of the accuracy is TP+TN/(TP+TN+FN+FP).

V. Conclusions

This paper has successfully developed a smart dissemination application broadcast messages to a particular student who meets certain criteria by using Wablas API and NLP. The smart dissemination contains two parts: dynamic filtering and dynamic content. The former is built by using POS tagger and "where clause" SQL statement, while the latter is built by using replace function in MySQL.

Certain sentences for testing the dynamic filter module show Recall score of 0.95, Precision 1, and also Accuracy 0.95. Nevertheless, there still twofold limitation in our works: the application could not transform a message that matches rule <3> with conjunction “dan” and has the same attribute before and after <CC> tag, and also maximum of a logical operator is only one type for coordinating conjunction in one sentence. We recommend that this study will be improved by handling more than one type of coordinating conjunction.

Overall this application can help lecturers to broadcast messages to particular students easily by using WhatsApp application.

REFERENCES

[1] WhatsApp Inc, “WhatsApp About,” 2020. [Online]. Available: https://www.whatsapp.com/about/. [Accessed: 30-Apr-2020].

[2] A. Schwind and M. Seufert, “WhatsAnalyzer: A tool for collecting and analyzing whatsapp mobile messaging communication data,” in *Proceedings of the 30th International Teletraffic Congress, ITC 2018*, 2018, vol. 01, pp. 85–88, doi: 10.1109/ITC30.2018.00020.

[3] M. Koparal, H. Y. Ünsal, H. Alan, F. Üçkardeş, and B. Gülsün, “WhatsApp messaging improves communication in an oral and maxillofacial surgery team,” *Int. J. Med. Inform.*, vol. 132, 2019, doi: 10.1016/j.ijmedinf.2019.103987.

[4] A. Sánchez-Moya and O. Cruz-Moya, “‘Hey there! I am using WhatsApp’: A Preliminary Study of Recurrent Discursive Realisations in a Corpus of WhatsApp Statuses,” in *Procedia - Social and Behavioral Sciences*, 2015, vol. 212, pp. 52–60, doi: 10.1016/j.sbspro.2015.11.298.

[5] S. Dechand, A. Naiakshina, A. Danilova, and M. Smith, “In encryption we don’t trust: The effect of end-to-end encryption to the masses on user perception,” in *Proceedings - 4th IEEE European Symposium on Security and Privacy, EURO S and P 2019*, 2019, pp. 401–415, doi: 10.1109/EuroSP.2019.00037.

[6] “Number of daily active WhatsApp Status users from 1st quarter 2017 to 1st quarter 2019(in millions),” *statista*, 2020. [Online]. Available: https://www.statista.com/statistics/730306/whatsapp-status-dau/. [Accessed: 30-Apr-2020].

[7] I. Rauf, E. Troubitsyna, and I. Porres, “Systematic mapping study of API usability evaluation methods,” *Comput. Sci. Rev.*, vol. 33, pp. 49–68, 2019, doi: 10.1016/j.cosrev.2019.05.001.

[8] T. Wablas, “Wablas API Gateway Service for Business,” 2020. [Online]. Available: https://wablas.com/.

[9] Y. W. Chandra and S. Suyanto, “Indonesian chatbot of university admission using a question answering system based on sequence-to-sequence model,” *Procedia Comput. Sci.*, vol. 157, pp. 367–374, 2019, doi: 10.1016/j.procs.2019.08.179.

[10] Y. Nagender and K. H. Patil, “Whatsapp auto responder using natural language processing and AI,” *Int. J. Comput. Eng. Technol.*, vol. 8, no. 5, pp. 15–22, 2017.

[11] D. A. N. Mysql, “Bot Whatsapp Sebagai Pemberi Data Statistik,” *J. Inform. dan Sist. Inf.*, vol. 1, no. 2, pp. 282–293, 2020.

[12] K. Bock and S. M. Garnsey, “Professional chat application based on natural language processing,” in *A Companion to Cognitive Science*, 2018, pp. 226–234, doi: 10.1002/9781405164535.ch14.

[13] M. Kamayani, “Perkembangan Part-of-Speech Tagger Bahasa Indonesia,” *J. Linguist. Komputasional*, vol. 2, no. 2, p. 34, 2019, doi: 10.26418/jlk.v2i2.20.

[14] A. Purwarianti, A. Andhika, A. F. Wicaksono, I. Afif, and F. Ferdian, “InaNLP : Indonesia Natural Language Processing Toolkit Case study : Complaint Tweet Classification,” in *2016 International Conference On Advanced Informatics: Concepts, Theory And Application (ICAICTA)*, 2016, pp. 5–9.

[15] A. Dinakaramani, F. Rashel, A. Luthfi, and R. Manurung, “Designing an Indonesian part of speech tagset and manually tagged Indonesian corpus,” in *Proceedings of the International Conference on Asian Language Processing 2014, IALP 2014*, 2014, pp. 66–69, doi: 10.1109/IALP.2014.6973519.

[16] R. S. Yuwana, E. Suryawati, and H. F. Pardede, “On Empirical Evaluation of Deep Architectures for Indonesian POS Tagging Problem,” *2018 Int. Conf. Comput. Control. Informatics its Appl. Recent Challenges Mach. Learn. Comput. Appl. IC3INA 2018 - Proceeding*, pp. 204–208, 2019, doi: 10.1109/IC3INA.2018.8629531.

[17] A. Hamzah and N. Widyastuti, “Document subjectivity and target detection in opinion mining using HMM POS-Tagger,” in *Proceedings of 2015 International Conference on Information and Communication Technology and Systems, ICTS 2015*, 2016, pp. 83–87, doi: 10.1109/ICTS.2015.7379876.

[18] L. R. Atmarani, M. Sudarma, and I. D. Giriantari, “Sistem Opinion Mining dengan Metode Pos Tagging dan SVM Untuk Ekstraksi Data Opini Publik pada Layanan JKBM,” *Maj. Ilm. Teknol. Elektro*, vol. 16, no. 1, p. 91, 2016, doi: 10.24843/mite.1601.13.

[19] T. Fahrudin, J. L. Buliali, and C. Fatichah, “Ina-BWR: Indonesian bigram word rule for multi-label student complaints,” *Egypt. Informatics J.*, vol. 20, no. 3, pp. 151–161, 2019, doi: 10.1016/j.eij.2019.03.001.

[20] Muljono, U. Afini, C. Supriyanto, and R. A. Nugroho, “The development of Indonesian POS tagging system for computer-aided independent language learning,” *Int. J. Emerg. Technol. Learn.*, vol. 12, no. 11, pp. 138–150, 2017, doi: 10.3991/ijet.v12.i11.7383.

[21] F. Eddy, “nlp-id 0.1.16,” *python software foundation*, 2020. [Online]. Available: https://pypi.org/project/nlp-id/0.1.6/. [Accessed: 28-May-2020].

[22] A. F. Wicaksono and A. Purwarianti, “HMM Based Part-of-Speech Tagger for Bahasa Indonesia,” in *4th International MALINDO (Malay and Indonesian Language) Workshop*, 2010, no. March 2017.

[23] A. Purwarianti, A. Saelan, I. Afif, F. Ferdian, and A. F. Wicaksono, “Natural language understanding tools with low language resource in building automatic indonesian mind map generator,” *Int. J. Electr. Eng. Informatics*, vol. 5, no. 3, pp. 256–269, 2013.

[24] Anonymous, “SQL Where Clause,” *w3schools*, 2020. [Online]. Available: https://www.w3schools.com/sql/sql_where.asp. [Accessed: 02-Jun-2020].