2017 1st International Conference on Informatics and Computational Sciences (ICICoS)

POS-Tagging for Non-English Tweets: An Automatic Approach

(Study in Bahasa Indonesia)

Devi Munandar, Endang Suryawati, Dianadewi Riswantini,
Achmad Fatchuttamam Abka, Rini Wijayanti, Andria Arisal
Research Center for Informatics, Indonesian Institute of Sciences, Bandung, Indonesia
devi@informatika.lipi.go.id, endang@informatika.lipi.go.id, diana@informatika.lipi.go.id
achmad.fatchuttamam.abka@lipi.go.id, rini.wijayanti@lipi.go.id, andria.arisal@informatika.lipi.go.id

***Abstract*—the studied approach to part-of-speech tagging for tweets in Bahasa Indonesia. Bahasa Indonesia, as well as many other non-English languages, are lacking in language processing resources. This is mainly due to the lack of usable local corpus. This paper describes our work on building tweet corpus in Bahasa Indonesia that have been tagged manually using tag set from previous work with the addition of new tags specifically for tweet data. This corpus then used to train neural network based tagger. The experiment results for training process obtained Skip-Gram model with 66.23% testing accuracy and 66.33% validation accuracy, f1 score for all model close 60% accuracy in POS tagger testing model and shows that the tagger can assign tags with the acceptable result using word vector as features.**

Keywords—part-of-speech; corpus; word embedding; artificial neural network

I. INTRODUCTION

Part-of-Speech (POS) tagging is one of the fundamental tasks in natural language processing (NLP) because it is used in most of NLP application, such as question answering, sentiment analysis, word sense disambiguation, etc. A POS tagger is employed by reading an input text and assigns a part of speech tag to each one of the words. Since manually tagging is very laborious and time consuming, some researches have been attempted to do this process automatically. The approaches that commonly used are rule based, probabilistic, and transformational based approach. Rule based POS tagger assigns a POS tag based on manually created linguistic rules [1]. Probabilistic approach finds the highest probability of tag which given by its surrounding context [2], while the latest approach is combination of rule based and probabilistic approach to automatically obtain symbolic rules from corpus [3]. However, a POS Tagger is created only for specific language and most of them are applied for English language, while others are very limited.

Bahasa Indonesia is a local language which also suffers from NLP resources although it is spoken by about 250 million of people. Besides being used in formal events, it is also widely used in social media context. According to We Are Social compendium of world digital stats of 2017, Indonesia now has 106 million active social media users or 40% by total population and it makes Indonesian as one of the biggest social media user for many years. Global Web Index also reveals that Indonesian user typically spend about three hours per day on social networking services. The report shows that there would be huge amount of textual Indonesian language resources already available and will be generated. These textual resources will be more beneficial if processed by natural language processing techniques.

The previous researches have been conducted to build Indonesian POS Tagging. Reference [4] develop POS Tagger by using Hidden Markov Model while [5] employs a rule-based approach. However, the POS tagger is only well performed on texts that mostly contains a formal language, but not on informal language text. Therefore, this research will focus on creating an automatic POS tagging for social media messages in Bahasa Indonesia. User-generated text in social media has more challenges since user often use informal language for expression, a lot of misspelling, non-standard abbreviation, out-of-vocabulary, etc. We employ word embedding and Artificial Neural Network (ANN) method to create a model for tag assignment. We add five new POS Tagset as well to accommodate the undefined words in social media context.

The paper is organized as follows. Some related researches in POS Tagging methods are given in Sec. II. The methodology, material and experiment design are described in Sec. III. Sec. IV shows experiment results and discussion. The paper is ended with a conclusion and future works.

II. RELATED WORKS

One of the techniques to perform POS Tagging in identifying meaning every word in a sentence is shown by analyzing the sentence in Odia language using Support Vector Machine technique. Various Part-Of-Speech are developed using five tagsets in very small corpus training. In their technique testing step, each word will be labeled POS as it does for training data. Uncertain words are tagged with supporting of lexicon, NER system and word suffix. Results of experiment are computed with precision, recall, and f1_score. Individual results are computed for noun, verb, adjective, pronoun and, adverb. These results are used to compare with the ANN method [6].

978-1-5386-0903-3/17/$31.00 © 2017 IEEE

The placement of a phrase method related to syntax by requiring grammar and post tagger is a chunking method used in the classification of sentence text input. The Subject category, Predicate, Object, Description and Complement are previously applied rules, while phrase is based on POS information, they use Hidden Markov Model based on POS tagger to support Indonesian language. Then use NLTK Regex Parser function for chunking text sentences, with a set of grammar rules that construct of regular expression [7].

In use of Hindi language, the Artificial Neural Network (ANN) approach uses speech tagger to analyze the effectiveness of sentences to overcome disambiguation. In the experiment they are divided into two phases; first, learning phase with manually-tuned training corpus input with output Part-Of-Speech Tagging according rules learned; second, tagging phase with Untagged corpus inputs by dividing the sentence with tokenize and indexes it as token, then labeling process using lexicon in accordance with the rules based POS Tagger. The next process uses ANN based parts of Speech tagger. The result of this second phase is accurate POS tagged corpus [8].

In addition, the use of word embedding to determining the model before the annotation is done to enrich the corpus, or in doing research on sentiment analysis. Its use by appearing twitter words from corpus as vectors are arranged into sentiment classes employ machine learning. Word label or token is provided by the seed lexicon. Representation vector is used according to the centroid of the bag-of-word vector of the tweets. Embedded word, which is a low-dimensional solid word training vector from a corpus document, the embedding of cutting-edge words used as a feature in the regression model to determine the relationship between Twitter words and positive sentiments [9].

III. Experiment Design, Material and Methodology

A. Data Collection

We use 1,000 manually annotated tweets for training reference and validation, and other 100,000 tweets for building embedding vectors. Those tweets were collected over ten days period in April 2017, from 10 Indonesian trending topics of those days with 1,000 tweets for each topic. There are 42,492 hashtags; 66,301 mentions; 44,291 URLs; 3,001 emotion symbols; and 40,056 discourse markers (or retweet).

B. Part-of-Speech (POS)

Part-of-speech (POS) tag defines word class of every word in a sentence. Collection of POS tagged words is called corpus, which is one of major requirement for various techniques in natural language processing such as language generation, information retrieval, text summarization, question and answering, machine translation, etc.

There are two major techniques in POS tagging. They are rule-based and probability-based tagging. Rule-based tagging is conducted in a top-down approach, where human linguist experts define various rules to be consulted for tagging process. While probability-based tagging is evaluated in a bottom-up manner, using a corpus as training data to determine probabilistically the best tag for every word within a context. In this experiment, we are looking for methods for generating a good and optimal corpus using ANN which is part of probability-based techniques for POS tagging.

C. Data Pre-Processing

Pre-Processing is a crucial process to produce an optimum model for training, testing, validation and POS tagging. There are several phases for pre-processing data:

1) The first phase: data collection in data storage using social media (raw text twitter) is formed into text file. Tweet text file is then cleaned by removing blank line, html code, Unicode, repeat punctuation and unnecessary trash tweets.

2) The second phase: create word embedding is tokenizing process and assigning POS tag for every token. Since there is no available annotated Indonesian corpus for social media (especially twitter) message, the POS tag assignment is carried out manually by experts.

Pre-processing is implemented to facilitate in tokenizing and avoid unexpected tweet and facilitate in doing the coding at application phase. This process is a customization that approaches the standard Indonesian language.

Indonesian language have 5 main parts of speech: adjective (*kata sifat*), noun (*kata benda*), verb (*kata kerja*), adverb (*kata keterangan*), and function words (*kata tugas*). There are many other characteristics of Indonesian language, which can be derived into 23 part-of-speech tags [10]. In order to accommodate various terms in social media, additional 5 POS tags that not provided in formal document tag [11] are incorporated. In the previous experiment without adding 5 POS tags for support social media document, it's resulted in 64.97% accuracy in training phase. Although the experiment results most similar number accuracy but we get more information about the distinction between the words of a formal document with specific words from social media twitter in words labeling. Those 28 POS tags is described more detail in Table 1. IOBES tagging scheme [12] is also used because of its expressiveness for handling two or more continuous words tagging assignment. It is simple by attaching additional symbol (S for single, B for beginning, I for inside, E for end, and O for other) in front of the original POS tag. For example, "buku" which has "NN" as its POS tag, will have "S-NN" as its tag with IOBES. Meanwhile the phrase "buku tamu" will have B-NN as tag for word "buku" and E-NN as tag word "tamu" [13].

Mention (**AT**) is used when a user wants to refer a message to other twitter users to engage them in a communication. It is identified from '**@**' usage before a twitter user name. **Hashtag** (**HASH**) is the symbol '**#**' that used before the relevant keywords to facilitate topic search and categorization.

Discourse marker (**DISC**) is identified with '**RT**' which is used when the user re-send (re-tweet) a message from another user's tweets. **Emoticon** (**EMO**) is the expression of emotion or feeling in the tweet that is represented by graphical or simpler with punctuation. **URL** (**Uniform Resource Locator**) is used when user wants to refer to a specific web address by adding its URL to the tweet.

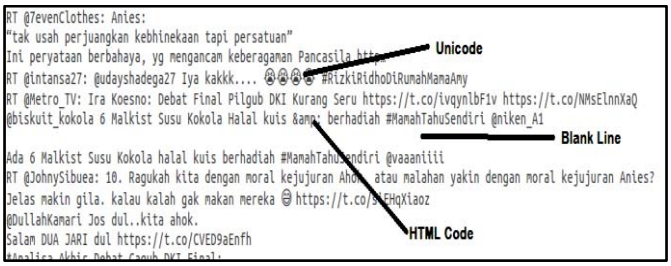

Fig 1. Stream for Indonesia tweet

TABLE I. POS TAGSETS IN TWITTER BAHASA INDONESIA [10][13]

Tag	Description	Example Document (in Bahasa Indonesia)	Example Twitter
CC	Coordinate Conjunction	dan, tapi, lalu	
CD	Cardinal Numeral	1938, tiga, seribu	
DT	Article / Determine	para, sang, si	
FW	Foreign Words	stress, gadget	
IN	Prepositions	kepada, untuk, tentang, dari	kpd, utk, dr
JJ	Adjective	kecil, cepat, kecewa	
MD	Modal or Auxiliaries Verbs	boleh, bisa, dapat	
NEG	Negations	Tidak, bukan,jangan	
NN	Common Noun	kelas, rahasia, bank	
NND	Classifier, Partitive, and Measurent Nouns	macam, kali, liter	
NNP	Proper Nouns	bapak, ibu, nyonya	
OD	Ordinal Numbers	pertama, kedua	
PR	Common/Demonstrative Pronouns	ini, itu, sana	
PRP	Personal Pronouns	kami, saya, dia, aku	
RB	Adverbs	hanya, selalu, juga	
RP	Particles	lah, kah, pun	
SC	Subordinate Conjunction	untuk, yang, agar	
SYM	Symbols	%, $	
UH	Interjection	hehehe, hai, wah	
VB	Verbs	melihat, menolak	
WH	Question	apa, siapa, kemana	
X	Unknown	ekuasi, ekivalen	
Z	Punctuation	, . ” – ( ) [ ] { }? !	
AT	***Mention***		@
DISC	***Discourse Maker***		RT :
HASH	***Hastag***		#
URL	***Uniform Resource Locator***		http:// https://
EMO	***Emoticons***		:) :(:))) :P

D. Word Embedding

Word embedding is the collective name for a set of language modeling and feature learning techniques in natural language processing (NLP) where words or phrases from the vocabulary are mapped to vectors of real numbers. Conceptually it involves a mathematical embedding from a space with one dimension per word to a continuous vector space with much lower dimension. Word embedding is required for natural language processing to create a vector representation of words. It is also very efficient for training with a large-scale data with very large vocabulary.

Word2vec is word embedding model that computationally efficient predictive model for learning word embedding from row text with uses the word vectors based representation to get better and less complicated NLP applications [14]. This model divided in two criteria, the Continuous Bag-of-Word model (CBOW) and the Skip-gram model.

Continuous Bag-of-Word model (CBOW) foresees the missing word, and arranges a window of context words with proof-missing word in a given sentence or phrase, or target prediction.

$$p(w_I) = y_j = \frac{exp(u_j)}{\Sigma_{j'=1}^{V} exp(u'_j)} \quad (1)$$

Skip-gram is probabilistic language model for information retrieval for searching nearest words in sequential, semantically or logical relationship, window of size ‘K’ words [15]. The definition of Skip-gram soft-max function is

$$p(w_I) = y_{c,j} = \frac{exp(u_{c,j})}{\Sigma_{j'=1}^{V} exp(u'_j)} \quad (2)$$

$$u_{c,j} = u_j = v'^{T}_{wj}.h \text{ for c = 1, 2, ..., } C \quad (3)$$

$$\mathrm{h} = \frac{1}{C}\mathrm{W}.(\sum_{i=1}^{C} \mathrm{x_i}) \quad (4)$$

Where v'^{T}_{wj} is the j^{th} column of the output matrix W'**.** The output y_j is obtained by passing the input u_j through the soft-max function of equation (1). Finally, could compute the output of j^{th} node of the c^{th} output word via soft-max function of (2).

Another word embedding model is GloVe. It generates word vectors by evaluating word co-occurrences within a corpus (twitter text). In order to train GloVe model, it is necessary to construct a co-occurrence matrix ***X.*** Element of $\boldsymbol{X_{ij}}$ represents how many times word *i* come up in context of word *j* [16].

$$J = \sum_{i=1}^{V}\sum_{j=1}^{V} f(X_{ij})(w_i^T w_j + b_i + b_j - log\ X_{ij})^2 \quad (5)$$

In (5), w_i is main word of vector, w_j is context word of vector, $b_i, b_j$are scalar biases for the main and context words.

Function f in (6) is a weighting function to prevent learning from extremely common word pairs. The following function:

$$f(X_{ij}) = \{(\frac{X_{ij}}{X_{max}})^{\alpha} 1\ if\ X_{ij} < X_{max} \quad (6)$$

E. Artificial Neural Network

Artificial neural network (ANN) is calculation model for information processing, defining, prediction and clustering. Just like nature neural networks, artificial neural networks have neurons to process inputs and outputs [17].

In recent years, artificial neural network has become very popular among modern researchers for solving the ill-defined or non-linear problems like engineering, medical, business and cyber security etc. A simple architecture of artificial neural network is presented in Fig. 2. This architecture consists of the nodes in hidden layers, network connections, initial weight controls and selection of activation functions, appears a very important role in the ANN modeling and their values usually depends on the characteristic of problem [17].

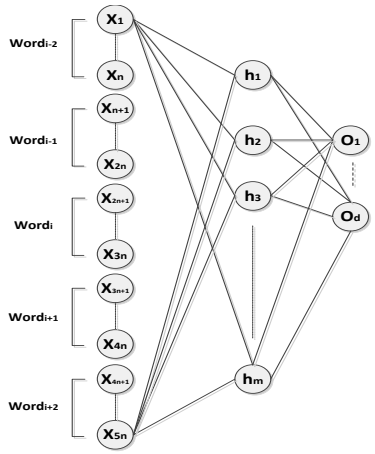

Fig 2. Architecture of artificial neural network [12]

IV. RESULTS AND DISCUSSION

Fig. 3 shows the POS tagging process for data input of Indonesian twitter data which is divided into 3 phases. *The first phase*, the word embedding vector dimension and corpus as the input component for the training process. Information embedding size, number of vocab and words index is part of embedding vector. The data label is defined in a function that consists of a class tag set for single word and multi word. The process of checking each word in the corpus matches with the index label of the tag class in module. *The second phase* of the process create the window data to determine the value of the target word using the model method defined. Models are loaded sequentially (Random, CBOW, SkipGram, and GloVe). The result of the matric vector value between the word index and the label index becomes input for the construction of the training data using ANN. *The third phase* process calculates precession, recall, f1 scores and accuracy values. Furthermore, the model is stored for use in the testing phase of POS tagging automatically.

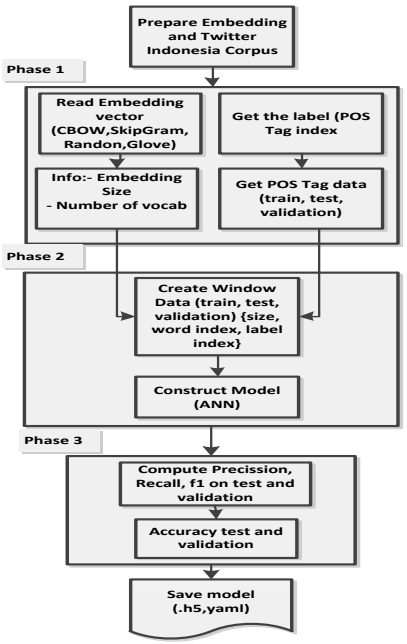

Fig 3. Training process POS tagging

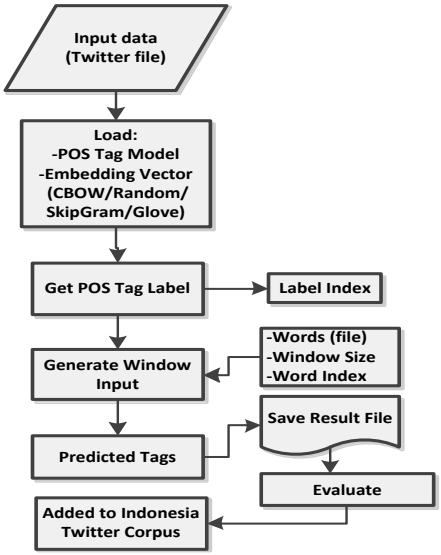

Fig 4. Automatic Testing Model POS Tagger

TABLE II. MODEL OF POS TAGGING TRAINING DATA WITH 28 TAG SETS

Model	Word Window	Context Win. size	Vector Dim.	Accuracy (%)	
				Tes.	Valid.
Random	5	2	500	63.98	**65.19**
	7	3	**500**	**64.12**	64.33
CBOW	7	3	100	64.46	**65.96**
	7	3	**400**	**66.16**	65.60
Skip-Gram	**7**	**3**	**200**	**66.23**	**66.33**
Glove	5	2	**100**	**61.40**	**62.19**
	7	3	200	60.75	61.78

In Fig. 4, it is shown automatic POS tagger with using twitter data input file that has been through pre-processing phase. POS tagging model that has built in previous experiment is loaded along with Embedding vector model. Generate window input needed label index from POS tag label then input words of twitter file, window size, and word index. Predicted process calls predicted class in model POS tagging to generate output input word and tag predict. Result of automatic POS tagging saved into text file. We needed human annotators to evaluate automatic POS tagging result to add into corpus.

In Table 2, it is presented POS-Tagging accuracy calculation results from training data using 4 word embedding models (Random, CBOW, Skip-gram and GloVe). Based on experiment training using SkipGram model has optimal accuracy in words window 7, context window size 3, vector dimension 200 with testing accuracy value is 66.23%, and validating accuracy value 66.33%.

Based on the training results in Table 2, the optimal value of each model is obtained based on the vector word embedding category tested. There are 6 word embedding vectors that give the optimal model; CBOW model with vector 50 and word context 3, Skip-gram model with vector 100 and word context 2, Skip-gram model with vector 200 and word context 3, CBOW model with vector 300 and word context 2, CBOW model with vector 400 and word context 3, Skip-gram model with vector 500 and word context 2. Tags with 25 samples are not presented that have Precision, Recall, and F1 Score because

they cannot be calculated. The data is spread on single word and multi-word using IOBES tagging scheme. From word embedding vector experiments, word context and in training model, with base vector using CBOW and Skip-gram word embedding models and word context 2 or 3 produce the optimal accuracy.

Automatic tagging is evaluated by automatically tagged 300 tweets with generated models to predict POS tag of every word (token). The predicted results are compared with the expert manual tagging result that has been properly checked. The calculation uses Precision-Recall metric to evaluate classifier output quality. Precision-Recall is a useful measure of success of prediction when the classes are very imbalanced.

$$P = \frac{T_p}{T_p + F_p} \tag{7}$$

$$R = \frac{T_p}{T_p + F_n} \tag{8}$$

$$F = 2\frac{PxR}{P+R} \tag{9}$$

$$Accuracy(y, \hat{y}) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} 1(\hat{y}_i = y_i) \tag{10}$$

TABLE III. PRECISION, RECALL, AND F1 SCORE OF MULTI EMBEDDING VECTOR DIMENSION TESTING DATA WITH 300 TWEETS

Tags	Samples	CBOW->Vector 50->Word Context 3			Skip-gram->Vector 100->Word Context 2			Skip-gram->Vector 200->Word Context 3		
		precision	recall	f1-score	precision	recall	f1-score	precision	recall	f1-score
B-DISC	183	100%	100%	100%	100%	100%	100%	100%	100%	100%
B-JJ	9	100%	33%	50%	0%	0%	0%	0%	0%	0%
B-NN	69	8%	1%	2%	100%	1%	3%	100%	1%	3%
B-NNP	41	28%	20%	23%	36%	20%	25%	29%	15%	19%
E-DISC	183	98%	100%	99%	100%	100%	100%	100%	100%	100%
E-JJ	9	100%	11%	20%	0%	0%	0%	0%	0%	0%
E-NN	71	50%	3%	5%	100%	1%	3%	50%	1%	3%
E-NNP	40	21%	17%	19%	0%	0%	0%	27%	7%	12%
I-AT	183	99%	100%	100%	99%	100%	99%	100%	100%	100%
S-AT	61	14%	31%	19%	11%	16%	13%	13%	25%	17%
S-CC	58	92%	76%	83%	100%	74%	85%	93%	71%	80%
S-CD	118	50%	25%	33%	87%	45%	59%	73%	44%	55%
S-EMO	2	100%	50%	67%	100%	50%	67%	100%	50%	67%
S-FW	182	56%	40%	46%	63%	48%	55%	66%	45%	54%
S-HASH	116	23%	20%	21%	20%	18%	19%	20%	19%	20%
S-IN	216	86%	78%	82%	90%	81%	85%	87%	81%	84%
S-JJ	133	21%	20%	20%	37%	22%	27%	31%	22%	26%
S-MD	81	82%	74%	78%	86%	79%	83%	95%	74%	83%
S-NEG	54	88%	70%	78%	86%	69%	76%	93%	70%	80%
S-NN	683	39%	59%	47%	39%	68%	50%	44%	49%	46%
S-NNP	566	40%	45%	42%	41%	57%	48%	43%	59%	50%
S-PR	74	92%	80%	86%	95%	81%	88%	100%	81%	90%
S-PRP	82	85%	62%	72%	74%	62%	68%	76%	59%	66%
S-RB	81	38%	31%	34%	44%	40%	42%	45%	47%	46%
S-RP	9	40%	22%	29%	100%	11%	20%	100%	11%	20%
S-SC	132	88%	66%	75%	74%	69%	71%	80%	71%	76%
S-UH	40	24%	23%	23%	21%	12%	16%	8%	5%	6%
S-URL	162	60%	69%	64%	67%	72%	70%	67%	76%	71%
S-VB	483	43%	42%	43%	53%	37%	44%	37%	55%	44%
S-WH	13	73%	62%	67%	100%	54%	70%	100%	54%	70%
S-X	31	27%	10%	14%	11%	3%	5%	27%	10%	14%
S-Z	585	98%	96%	97%	99%	99%	99%	99%	99%	99%
avg / total		60%	58%	58%	64%	62%	60%	63%	61%	60%

TABLE IV. PRECISION, RECALL, AND F1 SCORE OF MULTI EMBEDDING VECTOR DIMENSION TESTING DATA (CONTINUE)

Tags	Samples	CBOW->Vector 300->Word Context 2			CBOW->Vector 400->Word Context 3			Skip-gram->Vector 500->Word Context 2		
		precision	recall	f1-score	precision	recall	f1-score	precision	recall	f1-score
B-DISC	183	99%	100%	100%	100%	100%	100%	100%	100%	100%
B-JJ	9	100%	33%	50%	100%	22%	36%	0%	0%	0%
B-NN	69	40%	9%	14%	38%	9%	14%	100%	1%	3%
B-NNP	41	30%	27%	28%	36%	24%	29%	35%	17%	23%
E-DISC	183	99%	99%	99%	100%	100%	100%	100%	100%	100%
E-JJ	9	0%	0%	0%	100%	22%	36%	0%	0%	0%
E-NN	71	25%	1%	3%	20%	1%	3%	100%	1%	3%
E-NNP	40	24%	17%	20%	22%	10%	14%	25%	5%	8%
I-AT	183	100%	100%	100%	99%	100%	100%	99%	100%	99%
S-AT	61	9%	8%	8%	14%	23%	18%	12%	16%	14%
S-CC	58	98%	81%	89%	86%	74%	80%	94%	76%	84%
S-CD	118	76%	30%	43%	56%	26%	36%	86%	41%	55%
S-EMO	2	100%	50%	67%	100%	50%	67%	100%	50%	67%
S-FW	182	56%	41%	47%	50%	42%	46%	63%	43%	51%
S-HASH	116	24%	28%	26%	20%	15%	17%	29%	21%	24%
S-IN	216	91%	82%	86%	88%	81%	84%	92%	80%	85%
S-JJ	133	29%	17%	21%	26%	23%	25%	41%	20%	27%
S-MD	81	93%	83%	88%	86%	78%	82%	88%	81%	85%
S-NEG	54	91%	74%	82%	85%	65%	74%	89%	72%	80%
S-NN	683	38%	58%	46%	39%	57%	46%	40%	67%	50%
S-NNP	566	38%	50%	43%	37%	56%	44%	40%	57%	47%
S-PR	74	97%	81%	88%	98%	81%	89%	97%	81%	88%
S-PRP	82	86%	62%	72%	89%	62%	73%	88%	62%	73%
S-RB	81	47%	42%	44%	40%	41%	40%	49%	41%	45%
S-RP	9	50%	11%	18%	33%	22%	27%	100%	11%	20%
S-SC	132	95%	71%	81%	82%	73%	77%	84%	65%	74%
S-UH	40	21%	30%	24%	23%	15%	18%	19%	7%	11%
S-URL	162	58%	64%	61%	58%	72%	64%	67%	67%	67%
S-VB	483	46%	45%	45%	51%	34%	41%	47%	45%	46%
S-WH	13	100%	38%	56%	89%	62%	73%	100%	31%	47%
S-X	31	22%	6%	10%	25%	10%	14%	10%	3%	5%
S-Z	585	97%	96%	96%	98%	95%	97%	99%	98%	99%
avg / total		61%	60%	59%	60%	59%	58%	65%	62%	60%

Precision (P) is defined as the number of true positives (T_p) over the number of true positives plus the number of false positives (F_p).

Recall (R) is defined as the number of true positives (T_p) over the number of true positives plus the number of false negatives (F_n).

These quantities are also related to the (F1) score, which is defined as the harmonic mean of precision and recall.

Accuracy classification set score is computes of tag set labels predicted for a sample must exactly match the corresponding set of tag labels in true.

Where $\hat{y}_i$ is predicted value of the i^{-th} sample and y_i is the corresponding true value, then the fraction of correct predictions over $n_{samples}$

Table 3 and Table 4 show experiment results that compare the results of tagging using a model that has been done using tagging results with manual tagging that is considered to be accurate. On both tables that the results of Precision, Recall and F1 Score look almost similar, showing that in using Indonesian twitter data, the optimization using word context 2. In the Recall measurement can be seen that model success with context word and selected word embedding retrieval the correct tags result with actual number. F1 Score views mean value from Precision and Recall generated 60% for optimal value.Some tags set have a good precision on obvious words in Bahasa Indonesia twitter as an example; Such as discourse maker, mention, punctuation, preposition, emoticon, common / demonstrative have precision above 90%. However, the Recall and F1 have decreased in value in comparison. This relates to the amount of using train data and the multi vocabulary differences in Indonesian twitter data are not standardized, so the model generated when performing test data produces a fairly complex learning.

V. CONCLUSIONS

Part-of-speech tagging is not only usable for formal document analysis, but also can be used for analyzing microblogging messages.This paper triesto automate POS tagging for tweets in Bahasa Indonesia. The first phase is extracting twitter message in Bahasa Indonesia through the streaming API. Collected data is pre-processed by filtering and tokenizing as well as assigning various vectors for every word with word embedding techniques. Some documents are manually labeled. The training experiment was conducted with 1000 tweets that divided into 640 tweets for training, 160 tweets for validation and 200 tweets for testing. The training process is to generate models using word embedding vectors 50, 100, 200, 300, 400, 500 and word context 2, 3, 4. The results of each model are stored in the repository for testing. In testing phase and based on each vector model is taken with the most optimal value. The next step of testing is using Precision-Recall to find the optimal model for testing 300 data tweets. Tagging results using the generated model compared to manual tagging will give a value up to the accuracy for each model which is the first step of making the Indonesian twitter corpus.

In the next study, the experiment will be equipped with the addition of training data to produce a better model in the accuracy to recognize each word based on the tagging model used. And then will try to experiment with others model such as convolution neural network and recurrent neural network with connections form cycle between units.

REFERENCES

[1] G. Rubin. B. Greene. Automatic Grammatical Tagging of English. Technical Report, Department of Linguistics, Brown University, Providence, Rhode Island. 1971

[2] FemphyPisceldo, Manurung, R., Adriani, Mirna. Probabilistic Part-ofSpeech Tagging for bahasa Indonesia. Third International MALINDO Workshop, colocated event ACL-IJCNLP 2009, Singapore, August 1, 2009.

[3] Brill, E.A simple rule-based part-of-speech taggCher. In: Proceedings of the Third Conference on Applied Natural Language Processing (ANLP '92), Trento, Italy (1992) 152–155.

[4] Wicaksono, AlfanFarizki, and AyuPurwarianti. HMM based part-of-speech tagger for Bahasa Indonesia." Fourth International MALINDO Workshop, Jakarta. 2010.

[5] Rashel, Fam, et al. Building an Indonesian rule-based part-of-speech tagger. Asian Language Processing (IALP), 2014 International Conference on. IEEE, 2014.

[6] BishwaRanjan Das, SmrutirekhaSahoo, Chandra Sekhar Panda, SrikantaPatnaik, Part of Speech Tagging in Odia Using Support Vector Machine, Procedia Computer Science, Volume 48, 2015, Pages 507-512, ISSN 1877-0509.

[7] ArryAkhmadArman, Arif B. Putra N, AyuPurwarianti, Kuspriyanto, Syntactic Phrase Chunking for Indonesian Language, Procedia Technology, Volume 11, 2013, Pages 635-640, ISSN 2212-0173.

[8] Ravi Narayan, S. Chakraverty, V.P. Singh, Neural Network based Parts of Speech Tagger for Hindi, IFAC Proceedings Volumes, Volume 47, Issue 1, 2014, Pages 519-524, ISSN 1474-6670.

[9] Felipe Bravo-Marquez, Eibe Frank, Bernhard Pfahringer, Building a Twitter opinion lexicon from automatically-annotated tweets, Knowledge-Based Systems, Volume 108, 2016, Pages 65-78, ISSN 0950-7051.

[10] A. Dinakaramani, F. Rashel, A. Luthfi and R. Manurung, "Designing an Indonesian part of speech tagset and manually tagged Indonesian corpus," *2014 International Conference on Asian Language Processing (IALP)*, Kuching, 2014, pp. 66-69.

[11] Avontuur, T, Balemans, I, Elshof, L, van Noord, N, van Zaanen, M, "Developing a part-of-speech tagger for Dutch tweets," 2012 Computational Linguistics in the Netherlands Journal, 2012, pp. 34-51.

[12] Ronan Collobert, Jason Weston, Leon Bottou, Michael Karlen, KorayKavukcuoglu, PavelKuksa," Natural Language Processing (Almost) from Scratch", Journal of Machine Learning Research , vol 12 no. Aug, pp. 2493-2537, 2011.

[13] A. F. Abka, "Evaluating the use of word embeddings for part-of-speech tagging in Bahasa Indonesia," *2016 International Conference on Computer, Control, Informatics and its Applications (IC3INA)*, Tangerang, 2016, pp. 209-214.

[14] YafengRen, Ruimin Wang, DonghongJi, A topic-enhanced word embedding for Twitter sentiment classification, Information Sciences, Volume 369, 10 November 2016, Pages 188-198, ISSN 0020-0255.

[15] G. Remmiya Devi, P.V. Veena, M. Anand Kumar, K.P. Soman, Entity Extraction for Malayalam Social Media Text Using Structured Skip-gram Based Embedding Features from Unlabeled Data, Procedia Computer Science, Volume 93, 2016, Pages 547-553, ISSN 1877-0509.

[16] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global Vectors for Word Representation.", in EMNLP, Volume 14, 2014, pp. 1532-43.

[17] CaferMertYeşilkanat, YaşarKobya, HalimTaşkın, UğurÇevik, Spatial interpolation and radiological mapping of ambient gamma dose rate by using artificial neural networks and fuzzy logic methods, Journal of Environmental Radioactivity, Volumes 175–176, September 2017, Pages 78-93, ISSN 0265-931X.