Text-based LSTM Networks for Automatic Thai Love Quotes Generation on Twitter

Orathai Khongtum*, Nuttachot Promrit*, and Sajjaporn Waijanya*

Abstract

In this article, we propose the model for automatic Thai love quotes generation on Twitter using Text-based LSTM (Long Short-Term Memory) Network. This model is designed to learn the relationship of words in sentences from Twitter tweets with the hashtag "love quotes" and love songs 3,097 sentences 28,749 words. This approach differs from other previous research about sentences generation. The process of model training, we compare Loss from 2 input formats including with 1) Integer value 2) word2vec. The experimental has 4 approaches including 1) LSTM+Integer value with 2 words input 2) LSTM+Integer value with 3 words input 3) LSTM+ Word2Vec with 2 words input and 4) LSTM+Word2Vec with 3 words input. The LSTM+word2vec showed the lowest Loss. The evaluation using Human-targeted Translation Edit Rate (HTER). The average of HTER rate of LSTM+Word2vec model with 2 input words is 0.29 and for 3 input words is 0.26.

Keywords: Text-based LSTM, LSTM, RNN, Thai Love Quotes, Thai Love Quotes Generation, Automatic Generation.

1. Introduction

The automatic message generation to be similar to the message from the human for conveying emotions and thoughts is interesting and challenging work in Natural Language Processing and Artificial Intelligence. The goal of automatic message generation is to generate readable messages with understanding and right the grammar of the language. The automatic message generation has used in many tasks including automatic documentation generation, automatic weather reports from raw data, explanations in expert systems, medical informatics, machine translation, intelligence chat bot, robotic command, automatic question-answer. The automatic message is very challenging for arts in the message such as music composing, poetry composing and quotes generation. We've found that researchers are working to automatically generate Thai messages about question & answering [1], [2] and another work is mathematical expressions conversion into Thai text [3]. There are using template and conjunction words pattern to automatically generation.

Quotes are a short message that represents the art from the writer. Thai dictionary of the Royal Institute gives the meaning of the word quote is sharp message and thought-provoking [4]. The authors write quotes for teaching, encourage and it is the reflection of author's thoughts. The characteristic of the quotes in Thai language [5] including message with rhyme, message whit out rhyme. The quotes have showed in term of short message, phrase, repetition and short poem. There have the grammatical flexibility and many times the authors will write quote by incomplete sentence especially in Thai love quotes as in the Table 1.

Table 1. *The example of Thai quotes.*

Thai Quotes	Phonetic	English (Without Grammar)
ฤดูฝนเปียกปอน	rv^4du^1fon^5piak^2 p@n^1	rainy season-wet
ฤดูร้อนหวั่นไหว	rv^4du;^1r@;n^4 wan^2waj^5	summer-sensitive
ฤดูหนาวคิดถึงใคร	rv^4du;^1na;w^5 khid^4thvng^5khraj^1	winter-misses-who
ฤดูไหนก็คิดถึงเธอ	rv^4du;^1naj^5khid^4 thvng^5th#;^1	any season-miss-you

* *Department of Computing, Faculty of Science, Silpakorn University.*

As we show in the example, the grammatical flexibility in Thai quotes. If we use template technique we might have to define the huge number of template. Moreover, we never found any works using LSTM (Long Short-Term Memory) to generate Thai message.

In this article, we propose the model for automatic Thai love quotes generation on Twitter using Text-based network without template technique. The network has the design for learning the relation of words in Thai love quotes from Twitter and Thai love song. To measure our machine ability, the results has generated to 2 groups including 1) starting quote by 2 words and 2) starting quote by 3 words. Then we compare by average loss value and human evaluation.

2. Related Works

The automatic message generation is one important task in Natural Language Processing (NLP) especially art in message such as novel, music, quotes, poem. We found researchers use the Hidden Markov Models for prediction the sequence of words [6] by using probability from the model to learn the sequence of alphabet possibilities. They have generated the Polish text by use Polish novel trilogy of Henryk Sienkiewicz. Their machine can generate short text but cannot generate complete sentence. The automatic song has been generated by using Long Short-Term Memory networks (LSTMs). LSTMs is a special kind of Recurrent neural networks (RNN) which works for sequential data such as sequence of images or sequence of words. LSTMs has use for Rap lyric generation [7] and Music composition [8]. Data set of music composition compare the results between word-level-RNN and char-level-RNN. The experimental result of word-level-RNN is better.

Creating short stories automatically using LSTMs is another research [9]. The dataset is the concatenation of Conan O'Brien monologue jokes over the last 5 years including 258,443 words and bag of word size is 13,773 words. Those words are transformed by GloVe vector. The joke story starting with selecting the main word by entry word or random from the dictionary. The length of the sentence depends on the number of words entered or ending with a configuration value at the end of the sentence. The automatic joke story generation has evaluated by human evaluation score including 4 levels following score 0: Illogical, score 1: not funny, score 2: Quite funny and score 3: funny.

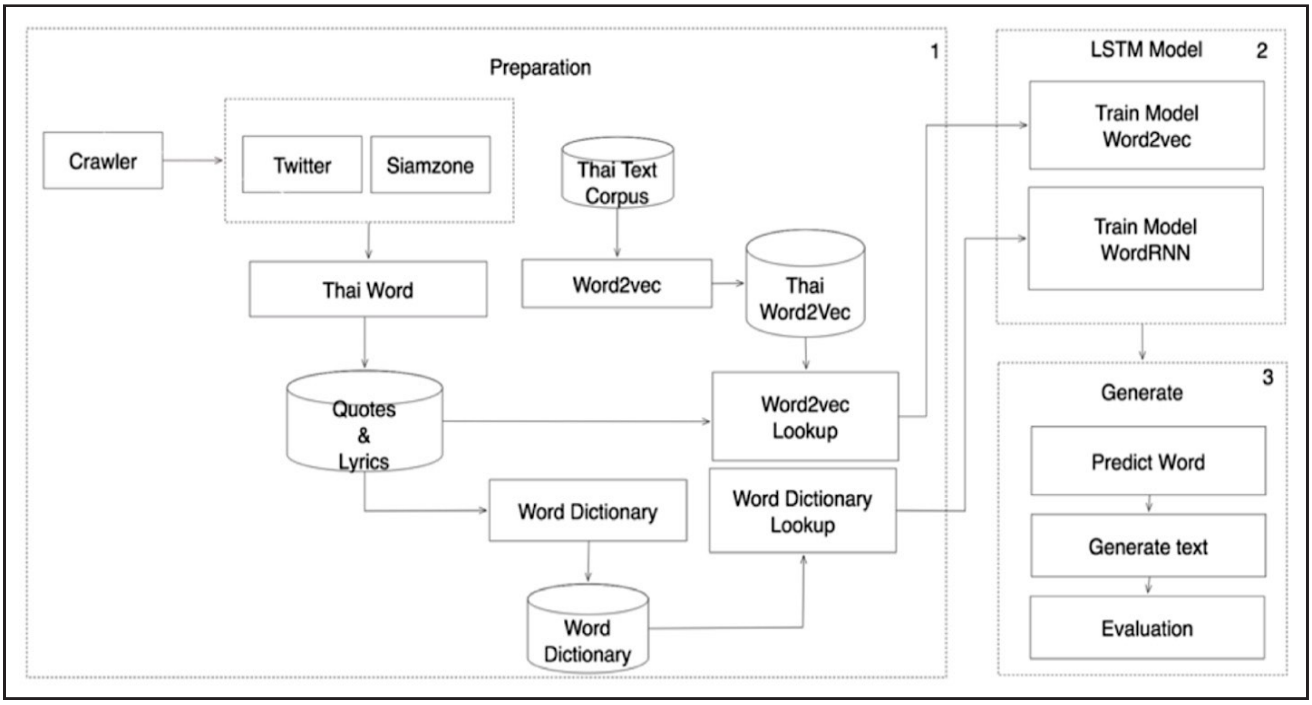


Figure 1. *Process overview of Thai love quotes generation on Twitter using Text-based LSTMs.*

3. Model and Methodology

Thai language used to be input of process in this article. We had 3 main process groups including with 1) the preparation process 2) creating LSTM models process and 3) Thai message generation process. The process overview of Thai love quotes generation on Twitter using Text-based LSTMs has shown in Figure 1

3.1 The Preparation

The Dataset of this article collect by crawl message from twitter with hashtag #lovequotes and crawl lyrics from www.siamzone.com. The dataset including 3,073 sentences. We used Thai word segmentation PhlongTaIam API [10] to segment words from all sentence to 25,242 words. We prepare 2 formats of dataset for compare loss values from Long Short-Term Memory Model training. The 1st format is transformed Thai word to be the unique integer value and each integer value represented by word frequency. Then create word list as shown the example in table 2.

Table 2. *The example of word list with integer values.*

Words	Integer Values
เขา (khaw^5, he)	127
มอบ (m@;b^3, give)	474
พอแล้ว (ph@;^1lx;w^4, enough)	928
ไม่อยู่ (maj^3ju;^2, away)	1,690
ปลอบ (pl@;b^2, soothe)	1,689
หนุ่ม (num^2, boy)	576

We use the same dataset prepare to 2nd format is Word2vec. We have created Word2Vec by defining size of word vectors 128 dimension and train Word2Vec by skip-gram model use Thai words from ThaiText corpus [11].

The result of Word2Vec from ThaiText corpus in Figure 2 shows a vector that represents semantic attributes of the words. The example of words with semantic relation in Thai are “หนุ่ม” = boy, “สาว” = girl, “หญิง” = woman, “ชาย” = man. These words are shown meaning in the group of gender. However, if we search “สาว, “หญิง, “ชาย” to get the integer value from word list the return value will be “0” since there are no those word in love quote dataset.

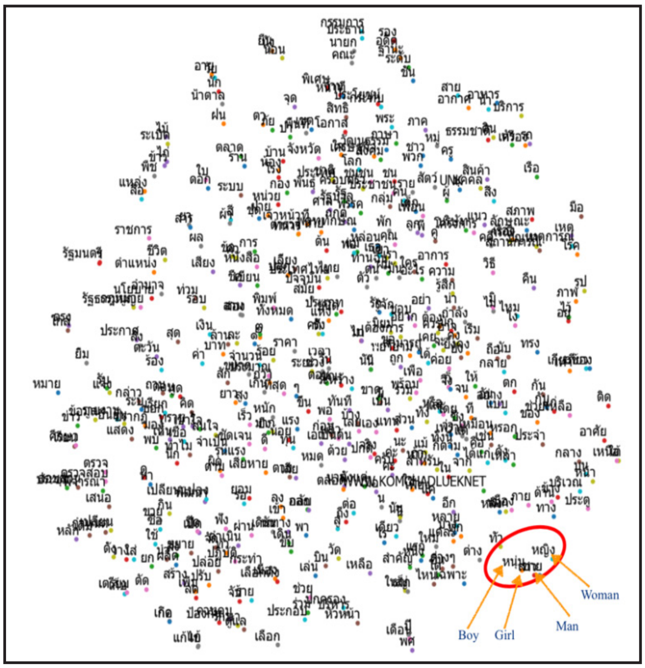


Figure 2. *Word2Vec from ThaiText Corpus.*

To present the value of the vector 128 dimension from Figure 2. We lookup vector value of each word and show the example words list with vector values in table 3.

Table 3. *The example of words list with vector values.*

Words	Vector [1 x 128]
เขา (khaw^5, he)	[-0.01 -0.08 0.11 .. 0.02 0.11 -0.03]
มอบ (m@;b^3, give)	[0.03 0.01 0.04 .. -0.02 0.10 -0.07]
พอแล้ว (ph@;^1lx;w^4, enough)	[0.02 -0.03 0.05 .. 0.07 0.15 -0.08]
ไม่อยู่ (maj^3ju;^2, away)	[-0.02 0.02 0.12 .. 0.17 0.03 -0.11]
ปลอบ (pl@;b^2, soothe)	[-0.01 -0.01 -0.04 .. 0.02 0.12 -0.23]
หนุ่ม (num^2, boy)	[0.04 0.04 0.01 .. 0.10 -0.06 -0.21]
สาว (sa;w^5, girl)	[-0.14 -0.01 -0.04 .. 0.11 -0.04 -0.17]
หญิง (cha;j^1, woman)	[0.01 -0.01 0.05 .. -0.01 -0.08 -0.05]
ชาย (jing^5, men)	[-0.01 -0.02 -0.06 .. -0.03 -0.00 -0.16]

3.2 Creating LSTM Models

In this article, we implement the model by using TensorFlow library for python and we use LSTM 1 layer as show in equation (1). We define LSTM unit as following. i_t is input gate, f_t is forget gate, o_t is output gate, c_t is memory cell and h_t is hidden state. When x_t is input at current time, σ is activate function sigmoid, W and U is weight and b is bias vector.

$$
\begin{aligned}
i_t &= \sigma\,(W^i x_t + U^i h_{t-1} + b^i)\\
f_t &= \sigma\,(W^f x_t + U^f h_{t-1} + b^f)\\
o_t &= \sigma\,(W^o x_t + U^o\, h_{t-1} + b^o)\\
g_t &= tanh\,(W^g x_t + U^g\, h_{t-1} + b^g)\\
c_t &= f_t . c_{t-1} + i_t\, . g_t\\
h_t &= o_t\ \ .tahn(c)_t
\end{aligned}
\qquad (1)
$$

LSTM 1 layer including hidden layer 512 dimension and loss function is cross-entropy. To optimize parameters, we select the RMSProp-Optimizer function which is an advanced form of gradient descent. Input data of LSTM model is random text from Twitter and Thai lyric. The process of LSTM with input 3 Thai words show in Figure 3.

We design 2 experiments. The 1st experiment, we use 2 Thai words by the matrix shape is 2x1 and 3 Thai words by matrix shape is 3x1 to be input. The value of the matrix is value transformed from Thai word to be the unique integer value. The 2nd experiment, we use 2 Thai words by the matrix shape is 2x128 and 3 Thai words by matrix shape is 3x128 to be input. The number 128 is size of Word2Vec vector dimension. We use Word2Vec model to transform words to vector values. The results of the LSTM model are generated by predicting of the probabilities that were normalized with the softmax function and select the maximum value to be answer.

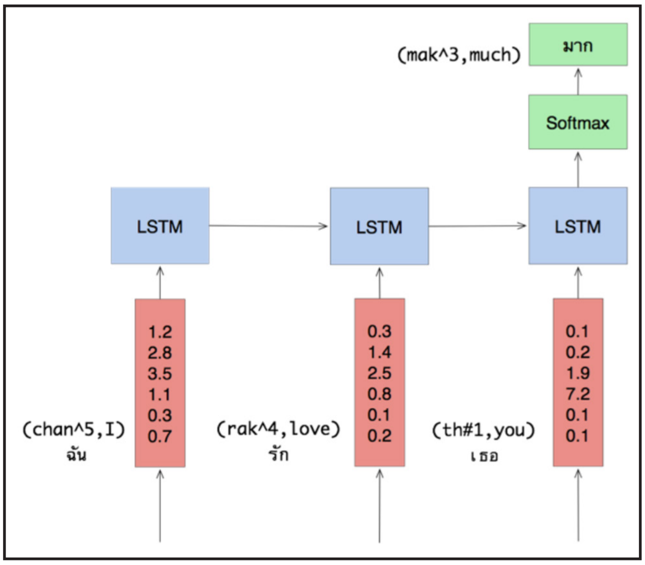


Figure 3. *The process of LSTM model with input 3 Thai words.*

3.3 Thai Message Generation

We use the model to generate Thai message (love quote). The input of model can be Thai words 2 or 3 words. Our model will predict the next word and it is repeated until finding the stop message symbol (in this article we use "."). Then the love quote will show by automatically generated from our system.

4. Experimental and Result

In this article, we experimented to measure the performance of model by compare the Average Loss of 2 data formats including 1) word as unique integer value (word-rnn) and 2) word as Word2Vec value (word2vec).

In each data format, we have adjusted input parameter words to be 2 and 3 words for training model. The comparison of average loss shows in Figure 4.

The experiment, we used 300,000th training iterations. The comparison of average loss in LSTM training was showed in Figure4. The loss value of Word2Vec with input 3 words has decreased obviously near 200,000th training iterations. Therefore, we can use the model with training less than 300,000th iterations.

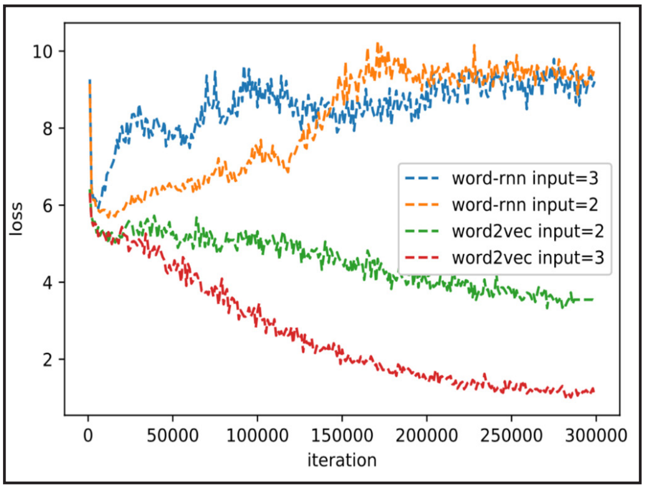


Figure 4. *The comparison of average loss in LSTM model training.*

Table 4. *The example messages from LSTM model with word as unique integer value.*

Input Words	Love Quotes
การมีรัก (To have love) ka;n^1mi;^1 rak^4	**การมีรัก**เธอรู้โลกจะโลกจะโลกจะโลกจะ (**To have love**, you know it, the world will be the world will the world will be) **ka;n^1mi;^1rak^4**th#;^1ru;^4 lo;k^3ca^2lo;k^3ca^2lo;k^3ca^2lo;k^3ca^2
คำว่ารัก (The word loves) kham^1wa;^3 rak^4	**คำว่ารัก**เธอได้โลกจะโลกจะโลกจะโลกจะ (**The word loves**, you got it, the world will be the world will the world will be) **kham^1wa;^3rak^4**th#;^1daj^3 lo;k^3ca^2lo;k^3ca^2lo;k^3ca^2lo;k^3ca^2
แต่ยิ่ง (But more) tx;^2jing^3	**แต่ยิ่ง**หน่อยเจอะกันทีไรทีไรตอนนี้ชอบชอบชอบชอบชอบ (**But more** when when meet you, now like like like like like) **tx;^2jing^3**n@;j^2c#^2kan^1thi;^1raj^1thi;^1raj ^1t@;n^1ni;^4ch@;b^3 ch@;b^3ch@;b^3ch@;b^3ch@;b^3

From Table 4. The Predictive words come up with many duplicate words and these are the meaningless message.

Therefore, the usable model is LSTM model with Word2Vec. To measure the performance of model, we test the model by 25 dataset of input words. The result of is LSTM model with Word2Vec (Input word = 2 words) show in Table 5. And the result of is LSTM model with Word2Vec (Input word = 3 words) show in Table 6. Both of Table 5 and 6, we show the phonetic alphabet of Thai words and translate the result to be English to evaluate the results as well.

Table 5. *The example messages from LSTM model with Word2Vec (input word = 2 words).*

Input = 2	Love Quotes
ฉันไม่ (I do not) chan^5 maj^3	**ฉันไม่**มีเธอใกล้ๆ กัน (**I do not** have you near) chan^5maj^3 mi;^1th#;^1klaj^3klaj^3kan^1
แต่ฉัน (But I) tx;^2chan^5	**แต่ฉัน**ห้ามมา (**But I** forbid) tx;^2chan^5ha;m^3ma;^1
เรามี (We have) raw^1mi;^1	**เรามี**ให้ใคร มีเธอใกล้ๆ กัน (**We have** for whom? I have you closely) raw^1mi;^1haj^3khraj^1mi;^1th#;^1klaj^3 klaj^3kan^1
แต่ยิ่ง (But more) tx;^2jing^3	**แต่ยิ่ง**นี้ก็รักเธอ (**But more** this is love.) **tx;^2jing^3**ni;^4k@^rak^4th#;^1
ให้ฉัน (Let me) haj^3chan^5	**ให้ฉัน**มีเพื่อใครใคร (**Let me** have for whom?) **haj^3chan^5**mi;^1phv;a^3khraj^1khraj^1

Table 6. *The example messages from LSTM model with Word2Vec (input word = 3 words).*

Input = 3	Love Quotes
คำว่ารัก (The word loves) kham^1wa;^3 rak^4	**คำว่ารัก**ที่สองเรานั้นเข้าใจ (**The word loves** that we understand) **kham^1wa;^3rak^4**thi;^3s@;ng^5raw^1 nan^4khaw^3caj^1
แต่มันคง (But it would) tx;^2man^1 khong^1	**แต่มันคง**รัก (**But it would** love) tx;^2man^1khong^1rak^4
ที่ว่ารัก (That I love) thi;^3wa;^3 rak^4	**ที่ว่ารัก**เธอมากเท่าไรฉันก็รักเธอยิ่งกว่าใคร (**That I love** you so much, I love you more than anyone) **thi;^3wa;^3rak^4**th#;^1ma;k^3thaw^3raj^1 chan^5k@^3rak^4th#;^1jing^3kwa;^2khraj^1
ให้ฉันดูแล (Let me take care) haj^3chan^5 du;^1lx;^1	**ให้ฉันดูแล**เธอมา (**Let me take care** of you) **haj^3chan^5du;^1lx;^1**th#;^1ma;^1
การมีรัก (To have love) ka;n^1mi;^1 rak^4	**การมีรัก**คือก็รักเธอยิ่งกว่าใคร (**To have love** is to love you more than anyone) **ka;n^1mi;^1rak^4**khv;#^1k@^3rak^4th#;^1 jing^3kwa;^2khraj^1

The evaluation of model we select the Human-targeted Translation Edit Rate (HTER) [12]. This evaluation method will have human in loop of editing message. The value of HTER score will be between 0-1 and the quality of model will be good if HTER score near “0”. All edits including substitutions, insertions, deletions, shifts of any number of words will be count and divide by number of words of post-edit versions (reference words) as show in equation (2).

$$HTER = \frac{\#\ of\ edits}{\#\ of\ reference\ words} \tag{2}$$

When number (#) of edit is #substitutions+
#insertions+
#deletions+
#shifts

To create the post-edit version, we invite 5 participants to read and edit the love quotes from 25 datasets. They are students in Computer Science Program and Information Technology Program. They are following favorite twitter user who always tweets love quotes and they usually re-tweet those messages. The example of post-edit version from LSTM+ Word-RNN (word as unique integer value) model shows in

Table 7. *The example messages by human post-edit and HTER score of LSTM+Word-RNN (word as unique integer value) model.*

Input Words	Quote Generated by Model	Post-edit Version	HTER
การมีรัก (To have love) ka;n^1mi;^1 rak^4	**การมีรัก**เธอรู้โลกจะโลกจะโลกจะโลกจะ (**To have love**, you know it, the world will be the world will the world will be) **ka;n^1mi;^1rak^4**th#;^1ru;^4 lo;k^3ca^2lo;k^3ca^2lo;k^3ca^2lo;k^3ca^2	**การมีรัก**เธอรู้โลกจะยิ้ม~~โลกจะโลกจะ~~ (**To have love**, you know it, the world will be smile) **ka;n^1mi;^1rak^4**th#;^1ru;^4 lo;k^3ca^2jim^4~~lo;k^3ca^2lo;k^3ca^2lo;k^3ca^2~~	0.58
คำว่ารัก (The word loves) kham^1wa;^3 rak^4	**คำว่ารัก**เธอได้โลกจะโลกจะโลกจะโลกจะ (**The word loves**, you got it, the world will be the world will the world will be) **kham^1wa;^3rak^4**th#;^1daj^3 lo;k^3ca^2lo;k^3ca^2lo;k^3ca^2lo;k^3ca^2	**คำว่ารัก**เธอได้รับโลกจะสดใส~~โลกจะโลกจะโลกจะ~~ (**The word loves**, you got it, the world will be bright) **kham^1wa;^3rak^4**th#;^1daj^3rab^4lo;k^3ca^2 sod^2saj^5~~lo;k^3ca^2lo;k^3ca^2lo;k^3ca^2~~	0.62
แต่ยิ่ง (But more) tx;^2jing^3	**แต่ยิ่ง**หน่อยเจอะกันทีไรทีไรตอนนี้ชอบชอบชอบชอบชอบ (**But more** when when meet you, now like like like like like) **tx;^2jing^3**n@;j^2c#^2kan^1thi;^1raj^1thi;^1raj^ 1t@;n^1ni;^4ch@;b^3 ch@;b^3ch@;b^3ch@;b^3ch@;b^3	**แต่ยิ่ง**หน่อยเจอะกันจะยิ่ง~~ทีไรทีไรตอนนี้ชอบชอบชอบชอบ~~ชอบ (**But more** we meet, we will like more) **tx;^2jing^3**n@;j^2c#^2kan^1 **ca^2 jing^3**~~thi;^1raj^1thi;^ 1raj^1t@;n^1ni;^4ch@;b^3 ch@;b^3ch@;b^3ch@;b^3~~ch@;b^3	0.57

Table 8. *The example messages by human post-edit and HTER score of LSTM+Word2vec model.*

Input Words	Quote Generated by Model	Post-edit Version	HTER
เรามี (We have) raw^1mi;^1	**เรามี**ให้ใคร มีเธอใกล้ๆ กัน (**We have** for whom? I have you closely) **raw^1mi;^1**haj^3khraj^1mi;^1th#;^1klaj^3 klaj^3kan^1	เรามี~~ให้~~ใคร แค่มีเธอใกล้ๆ กัน (**We have** whom? I have you closely) **raw^1mi;^1**~~haj^3~~khraj^1 kh;^3mi;^1th#;^1klaj^3 klaj^3kan^1	0.20
แต่ยิ่ง (But more) tx;^2jing^3	**แต่ยิ่ง**นี้ก็รักเธอ (**But more** this is love.) **tx;^2jing^3**ni;^4k@^3rak^4th#;^1	แต่ยิ่ง~~นี้~~นานก็ยิ่งรักเธอ (**But more** time I love you more) **tx;^2jing^3**~~ni;^4~~n@;n^1k@^3jing^3rak^4th#;^1	0.30
คำว่ารัก (The word loves) kham^1wa;^3 rak^4	**คำว่ารัก**ที่สองเรานั้นเข้าใจ (**The word loves** that we understand) **kham^1wa;^3rak^4**thi;^3s@;ng^5raw^1 nan^4khaw^3caj^1	**คำว่ารัก**คือคำที่สองเรานั้นเข้าใจ (The word loves that the word we understand) **kham^1wa;^3rak^4** khv;#^1kham^1thi;^3s@;ng^5raw^1 nan^4khaw^3caj^1	0.20
ที่ว่ารัก (That I love) thi;^3wa;^3 rak^4	**ที่ว่ารัก**เธอมากเท่าไรฉันก็รักเธอยิ่งกว่าใคร (**That I love** you so much, I love you more than anyone) **thi;^3wa;^3rak^4**th#;^1ma;k^3thaw^3raj^1 chan^5k@^3rak^4th#;^1jing^3kwa;^2khraj^1	**ที่ว่ารัก**เธอมากเท่าไร คือฉัน~~ก็~~รักเธอยิ่งกว่าใคร (**That I love** you so much, that is I love you more than anyone) **thi;^3wa;^3rak^4**th#;^1ma;k^3thaw^3raj^1 khv;#^1chan^5~~k@^3~~rak^4th#;^1jing^3kwa;^2khraj^1	0.17

Table 7 and example of post-edit version from LSTM+Word2Vec model show in Table 8. The word with underline is new words and the word with strike line is deleted.

The HTER score from Table 7 show 0.58, 0.62 and 0.57. The scores are near larger than 0.5 that means the quality of LSTM+Word-RNN model using word unique integer is not quite good. The HTER score from Table 8 show 0.20, 0.30, 0.20 and 0.17. The scores near “0” that means the quality of LSTM+Word2Vec model is quite good. From table 7 and 8 we can compare the example of HTER score of 2 models with same the start words in Table 9.

Table 9. *The comparison of 2 models with same start words.*

Start Words	HTER of LSTM+Word-RNN	HTER of LSTM+Word2Vec
คำว่ารัก (The word loves) kham^1wa;^3rak^4	0.62	0.20
แต่ยิ่ง (But more) tx;^2jing^3	0.57	0.30

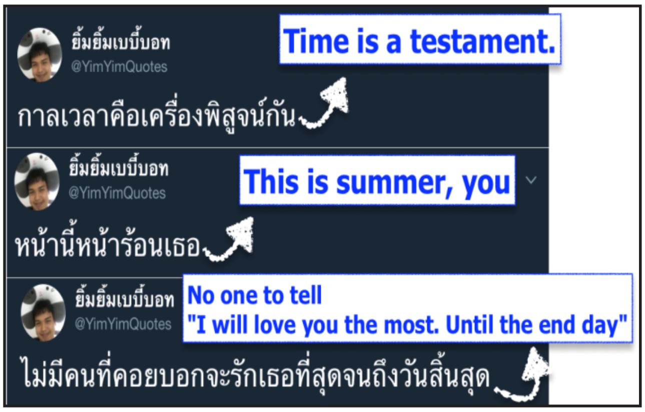
The result of the examples in Table 8 (LSTM +Word2Vec model) is better than the result of the examples in Table 7 (LSTM+Word-RNN model). Therefore we use the example of post-edit in Table 8 by correct from 5 participants and calculate the average score by person then average from all. The HTER score show in Table 10.

Table 10. *The LSTM+Word2vec model HTER score of 25 Love Quotes.*

Person	Input = 2 Words	Input = 3 Words
	HTER (%)	HTER (%)
1	0.28	0.26
2	0.27	0.26
3	0.28	0.25
4	0.36	0.27
5	0.28	0.25
Average	**0.29**	**0.26**

The average of HTER score of LSTM+Word2vec model with 2 input words is 0.29 and for 3 input words is 0.26. The result of Tables 4 to 10, we can confirm that relate with the average loss value in Figure 4. The average loss of LSTM+Word-RNN never drop in the same way with HTER score, if we use the Word-RNN model the HTER score will be near "1".

In additional, we use this model for implement the automatic love quote generation on Twitter and tweet the message as bot. The twitter account name is @YimYimQuotes. Our bot will be finding twitter status with Thai hash-tag #คำคมรัก (love quote). Then the bot will random 1 status and select 3 start words to be input and send to model. Model will predict the next word and it is repeated until finding the stop message symbol. The final process, the model will generate the love quote and tweet as we show in Figure 5.



ยิ้มยิ้มเบบี้บอท
@YimYimQuotes

Time is a testament.

กาลเวลาคือเครื่องพิสูจน์กัน

ยิ้มยิ้มเบบี้บอท
@YimYimQuotes

This is summer, you

หน้านี้หน้าร้อนเธอ

ยิ้มยิ้มเบบี้บอท
@YimYimQuotes

No one to tell
"I will love you the most. Until the end day"

ไม่มีคนที่คอยบอกจะรักเธอที่สุดจนถึงวันสิ้นสุด

Figure 5. *The example automatic tweet message by LSTM+Word2Vec Model.*

5. Conclusion and Future Work

In this article, we propose LSTM model and adjusted parameters to predict words and select words to automatic generate message on Twitter. In the process of model training we experiment with 2 formats of datasets including word transform to integer and word transform to vector. LSTM+word2vec with 3 words input can show low average loss and Human-targeted Translation Edit Rate can show low average edit rate as well. Even though the word corpus that use for this article is not big, but when we analyze the result of the evaluation it represents the acceptable score. The love quotes generated by our model are readable and understandable by the human.

The research on automatic Thai message generation is still small especially message in art area such as music composition and poem composition. Since those messages often reflect the emotion of author and meaning often create the impression. Next, we will apply LSTM and Word2Vec to compose Thai short poem in the future.

6. References

[1] H. Decha and K Patanukhom. "Development of Thai Question Answering System." *In Proceedings of the 3rd International Conference on Communication and Information Processing*. pp. 124-128. ACM, New York, NY, USA, 2017.

[2] C. Kwankajornkiet, A.Suchato, and P.Punyabukkana. "Automatic multiple-choice ques-tion generation from Thai text." *In 2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE)*. pp. 1-6, 2016.

[3] Available Online at https://gsbooks.gs.kku.ac.th/58/the34th/pdf/PMO7.pdf.

[4] Thai dictionary of the Royal Institute, Available Online at http://www.royin.go.th/dictionary/.

[5] W. Yooyen. *คำคมในภาษาไทย*. Available Online at http://anchan.lib.ku.ac.th/kukr/handle/003/20786.

[6] G. Szymanski and Z. Ciota. *Hidden Markov Models Suitable for Text Generation*, 2018.

[7] P. Potash, A. Romanov, and A. Rumshisky. *GhostWriter: Using an LSTM for Automatic Rap Lyric Generation*. Presented at the 2015.

[8] K. Choi, G. Fazekas, and M. Sandler. *Text-based LSTM networks for Automatic Music Composition*. ArXiv160405358 Cs. 2016.

[9] H. Ren and Q. Yang. *Neural Joke Generation*, 2017.

[10] Veer Sattayamas. *GitHub*, Available Online at https://github.com/veer66/PhlongTaIam: PHP Thai word breaker, 2014.

[11] S. Waijanya and N. Promrit. “The poet identification using convolutional neural net-works.” *Advances in Intelligent Systems and Computing*. Vol. 566, pp. 179-187, 2018.

[12] S. Waijanya and A. Mingkhwan. *Thai Poetry Machine Translation to English Automate Evaluation VS Human Post-Edit*. Presented at the 6 November 2014.