Thai Variable-Length Question Classification for E-Commerce Platform Using Machine Learning with Topic Modeling Feature

Mr. Wasu Chunhasomboon

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Computer Science and Information Technology
Department of Mathematics and Computer Science
FACULTY OF SCIENCE
Chulalongkorn University
Academic Year 2021

การจำแนกคำถามภาษาไทยที่ความยาวแปรผันได้สำหรับแพลตฟอร์มอีคอมเมิร์ซโดยใช้การเรียนรู้ของเครื่องด้วยคุณลักษณะการสร้างตัวแบบหัวข้อ

นายวสุ ชุณหสมบูรณ์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2564

Thesis Title	Thai Variable-Length Question Classification for E-Commerce Platform Using Machine Learning with Topic Modeling Feature
By	Mr. Wasu Chunhasomboon
Field of Study	Computer Science and Information Technology
Thesis Advisor	Associate Professor SUPHAKANT PHIMOLTARES, Ph.D.

Accepted by the FACULTY OF SCIENCE, Chulalongkorn University in Partial Fulfillment of the Requirement for the Master of Science

.... Dean of the FACULTY OF SCIENCE
(Professor POLKIT SANGVANICH, Ph.D.)

THESIS COMMITTEE

.... Chairman
(Professor CHIDCHANOK LURSINSAP, Ph.D.)

.... Thesis Advisor
(Associate Professor SUPHAKANT PHIMOLTARES, Ph.D.)

.... External Examiner
(Prem Junsawang, Ph.D.)

วสุ ชุณหสมบูรณ์ : การจำแนกคำถามภาษาไทยที่ความยาวแปรผันได้สำหรับแพลตฟอร์มอีคอมเมิร์ซโดยใช้การเรียนรู้ของเครื่องด้วยคุณลักษณะการสร้างตัวแบบหัวข้อ. (Thai Variable-Length Question Classification for E-Commerce Platform Using Machine Learning with Topic Modeling Feature) อ.ที่ปรึกษาหลัก : รศ. ดร.ศุภกานต์ พิมลธเรศ

ในปัจจุบันแพลตฟอร์มอีคอมเมิร์ชมีการเติบโตอย่างต่อเนื่องในทุกปี และกลายเป็นส่วนนึงของชีวิตประจำวัน อย่างไรก็ตามแอปพลิเคชันมีการเปลี่ยนแปลงเป็นระยะ ผู้ใช้งานใหม่หรือผู้ใช้งานที่มีประสบการณ์จะพบปัญหาได้ ซึ่งทางแพลตฟอร์มอีคอมเมิร์ซได้จัดช่องทางในการแก้ปัญหาไว้หลายช่องทางได้แก่ เอฟเอคิว อีเมล การคุยสด และการโทรศัพท์ โดยปกติเอฟเอคิวจะถูกเมินเนื่องจากความยากในการค้นหาคำตอบที่ต้องการ ช่องทางที่เหลือใช้ได้ อย่างไรก็ตามจำนวนผู้ใช้ที่สูงเป็นเหตุให้เกิดคอคอดโดยเฉพาะในเหตุการณ์พิเศษทำให้ผู้ใช้ได้รับการช่วยเหลือล่าช้าเนื่องจากตัวแทนบริการลูกค้าสามารถตอบผู้ใช้ได้ทีละครั้งในเวลาหนึ่ง ดังนั้นวิทยานิพนธ์ฉบับนี้เสนอการจำแนกคำถามภาษาไทยที่ความยาวแปรผันได้สำหรับแพลตฟอร์มอีคอมเมิร์ช โมเดลที่นำเสนออยู่บนฐานของการรวมตัวของสถาปัตยกรรมสองแบบเข้าด้วยกันได้แก่การจัดสรรของดีรีเคลแฝง และ โครงข่ายหน่วยความจำระยะสั้นยาวแบบสองทิศทางเพื่อใช้เป็นกระบวนการสกัดคุณลักษณะ ผลลัพธ์ที่ได้จะนำมาต่อกันและป้อนสู่เพอร์เซปตรอนแบบหลายชั้นที่ใช้ฟังก์ชันกระตุ้นซอฟต์แม็กเพื่อจำแนกคำถามที่เข้ามา ผลการทดลองได้ชี้ให้เห็นว่าโมเดลที่นำเสนอให้ผลดีกว่าโมเดลการจำแนกข้อความที่มีอยู่ด้วยความแม่น 84.43%

สาขาวิชา	วิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ	ลายมือชื่อนิสิต ....
ปีการศึกษา	2564	ลายมือชื่อ อ.ที่ปรึกษาหลัก ....

6278510623 : MAJOR COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

KEYWORD: Variable-length text classification, Thai natural language processing, E-commerce platform

Wasu Chunhasomboon : Thai Variable-Length Question Classification for E-Commerce Platform Using Machine Learning with Topic Modeling Feature. Advisor: Assoc. Prof. SUPHAKANT PHIMOLTARES, Ph.D.

Nowadays, e-commerce platform continuously grows every year and becomes a part of our daily life. However, the application changes from time to time. Either new users or experienced users could face a problem. Several channels, which are FAQ, email, live chat, and call, are provided by e-commerce platform to cope with the problem. FAQ is usually ignored because it is hard to search for the desired answer. The rest channels are applicable. However, the huge number of users causes a bottleneck especially in the special events which delays the users to receive help because customer service agent can reply to the user once at a time. Therefore, this thesis proposed Thai variable-length question classification for e-commerce platform. The proposed model is based on a fusion of two architectures, Latent Dirichlet Allocation (LDA) and Bidirectional Long Short-Term Memory (Bi-LSTM), as a feature extraction process. Then, the results are concatenated and fed into a multilayer perceptron (MLP) network with a softmax as an activation function to classify an incoming question. The experimental results indicated that the proposed model outperforms the existing classification models with an accuracy of 84.43%.

Field of Study:	Computer Science and Information Technology	Student's Signature ....
Academic Year:	2021	Advisor's Signature ....

ACKNOWLEDGEMENTS

I would like to express my deep gratitude to everyone who has helped me to complete this thesis

This thesis could not be completed without advice, suggestion and dedication by my advisor, Associate Professor Dr. Suphakant Phimoltares. I would like to say that I am grateful to be under his advice.

I would like to express my gratitude to all thesis commitee: Professor Chidchanok Lursinsap and Dr. Prem Junsawangfor their insightful comments and time.

Furthermore, I would like to express my thankful to my family and friend for their encouragement and support.

Wasu Chunhasomboon

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER I
INTRODUCTION

1.1 Background and Rationale

At present, the e-commerce platform grows dramatically every year [1]. It becomes an essential part of our life. The reason is its convenience. Buyers and sellers can interact across the globe using simple electronic devices. However, there are several existing e-commerce platforms in the market. The features and policies are dependent on each e-commerce platform. Thus, either new users or experienced users could have questions regarding an application. To handle with the problem, e-commerce platform provides several channels such as call, email, frequently asked questions (FAQ) and live chat. However, during the special events, users must wait for a long time to receive help due to a number of incoming cases. This is an unsatisfied experience that could lead to losing users to competitors.

Frequently Asked Questions (FAQ) page is usually provided by e-commerce platform. It is a list of questions and answers intended to provide useful information to users. However, there are plenty of topics and subtopics in the FAQ. Users are required to search with the correct keywords in order to obtain the answer. Many users might find that this requirement is difficult. Thus, they prefer the other channels.

Live chat is an online service allowing users to ask questions regarding an application with the customer service agent directly. It is a user-friendly approach because it does not require users to use the specific keywords. However, this approach requires a person to reply to the users. The problem arises during the special events. The need of customer services increases drastically, whereas the customer service agent is able to serve a customer once at a time. Thus, users have to wait longer for help.

To cope with the problem, chat bot is one of the most efficient approaches. There are three supporting reasons. Firstly, users are not required to use specific keywords. Secondly, it can answer user's question automatically and does not require a person to monitor all day. Thirdly, it is scalable to handle incoming-questions efficiently.

To create a practical chatbot is a challenging task. A writing style and a word choice may be different for users who have the same question. The first challenge relates to quality of information. Some users might describe questions with insufficient information. It causes a difficulty for classification due to its ambiguousness. Others might describe questions with too many details which are specific to their cases. The supporting details do not help a model distinguish topics easier but act like a noise resulting in difficulty for classification. The second challenge is the number of classes that a model can classify. The more the number of classes a model can classify, the more practical a model can achieve. Therefore, it requires a model that can distinguish unique patterns among classes. Lastly, there is no blank space in Thai writing structure causing a tokenization becomes a difficult task. Thus, it is required to use a proper tokenizer to achieve a high performance.

From many prior works in a text classification, there are several machine learning techniques proposed to handle this kind of problem. For instance, Topic Similarity K-Nearest Neighbor (TS-KNN), Recurrent Neural Network with Latent Dirichlet Allocation (RCL) and Bi-LSTM with Convolutional Neural Network (Bi-LSTM-CNN). However, these models are trained to handle specific tasks, either to classify a short text or a document, whereas the questions obtained from users have different characteristics in which the length depends on the individual writing style. Thus, creating the classification model that is performed well on any text length is still a challenging task.

1.2 Research Objectives

To create a proper model for variable-length question classification in order to provide a correct answer to the users.

1.3 Scope of the work

1. The domain of the study is to classify Thai-FAQ for the E-commerce platforms.
2. The variable-length questions are synthesized from a survey. The number of words in each question is between 2 words to 83 words.
3. The questions can be classified into 31 classes. They are derived from 4 main classes of the E-commerce FAQ pages which are account, payment, logistic, and refund.
4. A question is considered to be a short one when it contains less than 15 words [2].
5. There are only Thai questions used in the study.
6. English words are translated into Thai due to insufficient data.
7. Abbreviations and contractions are rewritten into a full form due to insufficient data.
8. The word tokenizer used in the study is the Maximal Matching.

1.4 Expected Outcomes

This research aims to propose a proper model for classifying questions into 31 classes. Both local features and global features will be used to design the proposed model, which is able to provide an informative answer to the users.

1.5 Organization

In this thesis, there are 5 chapters which are introduction, literature review, proposed model, experiments and results and conclusion. The current chapter, chapter I, is an introduction. It gives an overview information about the thesis. Then, in chapter II, literature review, there are 7 related studies described. They are different in terms of domains and model frameworks in order to find the proper models that can handle the variable-length text. Subsequently, the chapter III is proposed model. This chapter describes and visualize the proposed model step-by-step. Then, the chapter IV represents experiments and results. In this chapters, there are serval experiments were conducted to achieve the best performance of the proposed model and to ensure that the proposed is better than the existing models. Finally, the chapter IV, conclusion, represents the core information from the previous chapters.

CHAPTER II

LITERATURE REVIEW

As described earlier, many researchers proposed a methodology to handle the text classification in the recent decades. In this chapter, there are seven related studies provided as follows.

The first study aims to classify an emotion in Thai YouTube comments [3]. It was proposed by Phakhawat Sarakit et al. The study focuses on six emotions which are happiness, surprise, sadness, fear, disgust and anger. The comments used to classify an emotion are from two different sources which are music video (MV) and commercial advertisement video (AD). The proposed framework consists of four processes. Firstly, there are four pre-processing techniques applied on the YouTube comment which are tokenization, stop word removal, HTML tags removal and word stem. Secondly, the feature selection applied by discarding word that does not appear more than the minimum threshold. Thirdly, term frequency (TF) and term frequency-inverse document frequency (TF-IDF) are used to transform comment into a vector. Lastly, the experiment was conducted on three classification methods which are Support Vector Machine (SVM), Multinomial Naïve Bayes (MNB) and Decision Tree. Figure 1 refers to these four processes of the proposed framework.

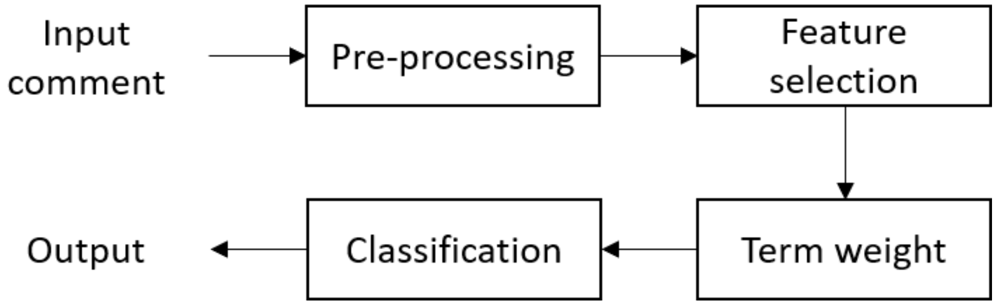


Figure 1: The emotion classification framework

The experimental result indicated that SVM can achieve the best performance on comments on the music video with the accuracy of 82.28%, whereas MNB can achieve the best performance on comments on the advertisement video with the accuracy of 76.14%. Therefore, the data characteristic and data distribution are important factors for using a proper classification method. Moreover, the study indicated that using TF or TF-IDF cannot handle ambiguous comments because they consider only a word frequency but ignore a semantic.

The second study is a model based on SVM-KNN for English document classification [4]. It was proposed by Yun Jin and Jian Wang. It aims to classify the documents into ten classes. The proposed model is a fusion of the Support Vector Machine (SVM) and the K-Nearest Neighbor (KNN) in order to eliminate the downside of both algorithms. The KNN is sensitive to a local feature, whereas the SVM sometimes allows some misclassifications on the boundary. The proposed framework is as follows. Firstly, for the document to be classified, the SVM classifier is used to calculate the probability of the document for every topic. Secondly, it calculates the difference between two highest probabilities. If the difference is higher than a threshold, the topic that has the highest probability is used as the classification results. Otherwise, the classification result is obtained from the KNN method. The experimental result indicated that F1 score of the proposed model is more than 1% higher than the stand alone SVM or KNN.

The third study is the English document classification based on sparse topic model [5]. It was proposed by Tao Liu. The author claimed that using a bag-of-words to represent documents cannot reveal semantic information, such as polysemy and synonym. Thus, a document-topic feature was proposed as a document representation. It consists of a set of probabilities that a document belongs to each topic. It is constructed using the Latent Dirichlet Allocation (LDA). The main concept of LDA is that a word can represent several topics and is highly correlated to other words within the same topic. After that, SVM is used as a classifier. The data used in this study is Reuter-21578 corpus. The experimental results indicated that the proposed model is able to achieve a higher F1 score than the traditional method which is SVM based on bag-of-words.

In the fourth study, Qiuxing Cheng et al. proposed the Chinese short text classification based on LDA model [6]. The challenge arises from the characteristic of the short texts. They rarely have common words within the same topic, causing the traditional classification methods which are solely based on the bag-of-words (BoW) model cannot perform well. However, they found that the distinct words between two short texts can be used to reveal the hidden relationship. They combined the BoW model with the Latent Dirichlet Allocation (LDA). LDA was used to obtain the probability of each word belonging to the topics. Then, the modified K-Nearest Neighbor (KNN) algorithm was used as a classifier. They used the Cosine similarity method as a distance metric of KNN algorithm. The data used in this study include the posts from Sina news containing 6 categories. The experimental result indicated a significant improvement. The proposed model had around 25%-47% increase for all evaluation matrices which are precision, recall, and F1-score.

The fifth study is the Automated Thai-FAQ Chatbot using RNN-LSTM [7] proposed by Panitan Muangkammuen et al. The authors claimed that the existing chatbots are not efficient, such as Dual Encoder. Each question was compared with all possible outcomes. Thus, the authors proposed Long-Short Term Memory (LSTM) to extract a feature from a question. LSTM can carry an important feature over a long period which is suitable for a data characteristic of text questions. Then, they used Softmax function as a classifier to classify the given question. There are 2,636 questions used in this study. They are categorized into 80 classes. Moreover, the authors split the questions into three sets. Firstly, the training set had 60% of questions. Secondly, the validation set had 20% of questions. Thirdly, the test set data had 20% of questions. During the experiment, the authors found a significant pattern that the average probability of correct prediction is 0.92 and incorrect prediction is 0.48. Thus, they set a threshold. If the probability is less than 0.5, the given question will be discarded. As a result, the proposed model ignored 13.64% of questions. However, the accuracy increased from 83.9% to 93.2%.

In the sixth study, Yongchao Yan and Kai Zheng proposed the Chinese Document Classification Model Based on Multi-level Topic Feature Extraction [8]. The authors fused two different type of classification models which are a discriminative model, Bi-Directional Long-Short Term Memory (Bi-LSTM) and a generative model, Latent Dirichlet Allocation (LDA). Bi-LSTM is capable to extract a local feature of a document, whereas LDA is capable to extract a global feature of a document. Furthermore, the dimensionality reduction technique, max pooling layer, was applied to an output of Bi-LSTM in order to reduce the computational cost and emphasize important features. Then, a linear activation function was used to classify the given document. In this study, the documents were obtained from two sources. 100,000 news documents were obtained from Sina News containing 10 news categories and 25,000 documents were obtained from takeaway review containing 2 emotional categories. The evaluation matrices used in this study are precision, recall, F1-score and accuracy. The experimental result indicated that the proposed model achieves the highest score among all comparing models for all evaluation matrices.

The last paper was proposed by Chenbin Li et al. named Chinese News Text Classification based on Improved Bi-LSTM-CNN [9]. The authors tried to improve the accuracy of a text classification using an existing deep learning architecture. The proposed model consists of four processes. Firstly, each word is converted to a vector by Continuous Bag-of-Word method. Secondly, Bi-Directional Long-Short Term (Bi-LSTM) is applied to extract a local feature. Thirdly, the Convolutional Neural Network (CNN) is applied in order to obtain a global feature. Lastly, Softmax function is applied with a global feature to classify the given document. The experiment was conducted on 65,000 documents from THUCNews containing 10 categories. The experimental result revealed that the proposed model achieves 99% of F1-score.

According to the abovementioned studies, there are several fusion techniques proposed such as TS-KNN, RCL and Bi-LSTM-CNN. These models are constructed to handle a specific data characteristic, either short or long text. As a result, they cannot perform well when dealing with a variable-length text which is a data characteristic of text questions.

CHAPTER III

PROPOSED MODEL

In this thesis, the methodology consists of two sections including data pre-processing and classification model. Firstly, data pre-processing is a word tokenization. It is a process of splitting sentence into words. Secondly, a classification model is a fusion of two machine learning techniques, a generative model and a discriminative model. They are used to extract different levels of features. Then, these features are used to obtain the answer that matches with the given question. The proposed workflow can be visualized as shown in Figure 2.

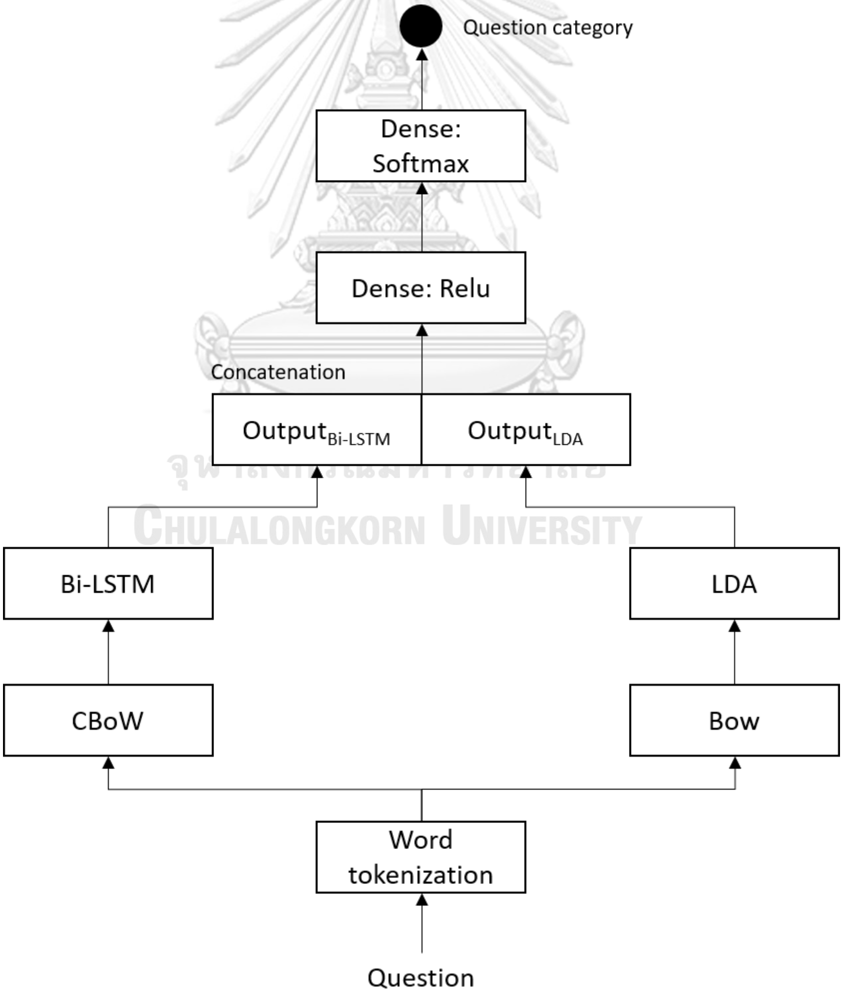


Figure 2: The proposed model structure

3.1 Data pre-processing

To reveal a latent pattern of a text question, a text question has to be converted from a text to a vector or a matrix. There are several methods which can be implemented by considering the whole question, words or characters. In this study, a text question is converted by considering words. The reason is that a word carries more information than a character and also avoids a sparseness when considering the whole question. However, there are challenges. Thai writing structure does not have a blank space between words which makes it difficult to split a text question into a list of words. Furthermore, a live chat is a semi-formal place. Inquirers sometime intentionally add additional characters of the last character of a question to make it friendly. Therefore, the maximum matching might be one of the most suitable methods, described as follows:

3.1.1 Maximum Matching

The maximum matching is a dictionary-based word tokenization approach. It is constructed to solve the drawback of longest matching which is a greedy approach. Typically, the longest matching scans characters from left to right and check those characters whether they are in a dictionary or not. If they are found, the scan will continue. If not, the latest series of characters that found in a dictionary is tokenized. As a result, the longest matching typically makes an incorrect tokenization by overextending words. On the other hand, the maximum matching is not a greedy approach. It tries to tokenize words in several perspectives with two sequential rules. Firstly, it prioritizes lists of tokenized words with the fewer unknown words. Then, it selects a list of tokenized words that has the minimal number of words. An example of maximum matching is shown in Figure 3.

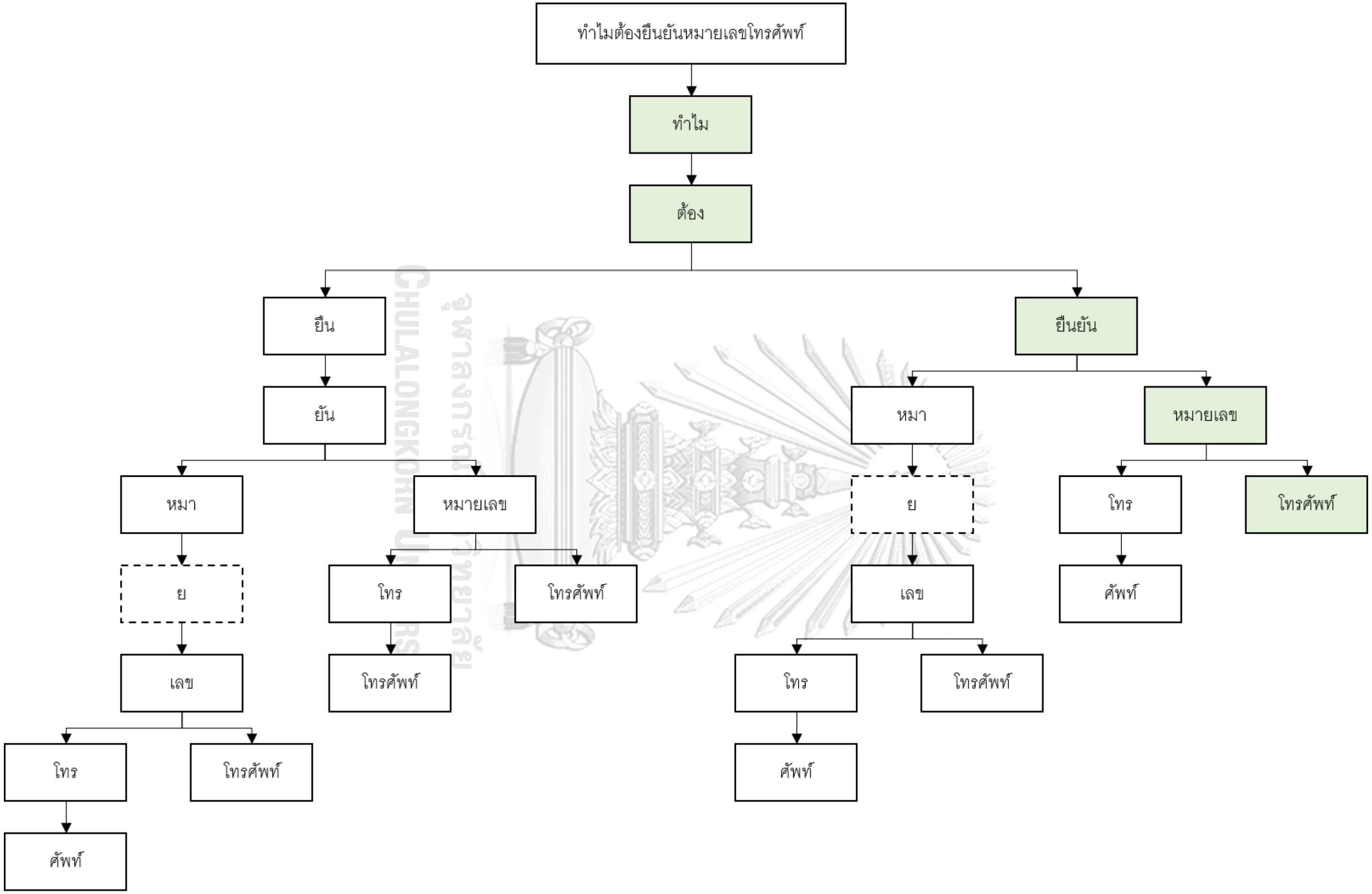


Figure 3: An example of maximum matching

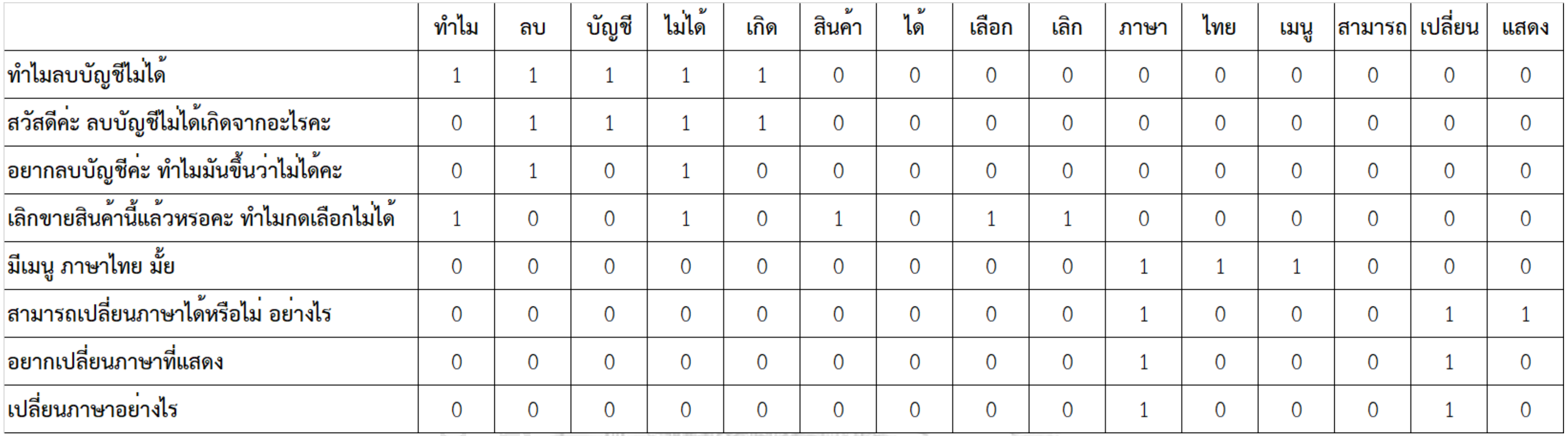
As shown in Figure 3, Maximal matching generates several lists of tokenized words. It can be visualized by a tree where a root node is the first word and a leaf node is the last word. A list of tokenized words can be obtained by traversing though a tree. Moreover, because Maximal matching is a dictionary-based method, there are two types of nodes. A solid boarder represents a word found in dictionary, whereas a dot boarder represents a word that cannot be found in dictionary or an unknown word. The result from Figure 3 is the path on the right most because it matched Maximal matching's rule. It is a list of tokenized words that has a minimal number of both unknow words and words.

3.2 Classification model

The existing text classification models are constructed for a data with single data length, either short or long. However, text questions obtained from a live chat have a variable-length which depends on an individual writing style. This study aims to solve this issue by applying existing techniques and models. The proposed classification model is a fusion of two different classification techniques which are based on generative model and discriminative model. A generative model is the Latent Dirichlet Allocation (LDA). It is constructed based on Bag-of-Words (BoW) as a representation of text questions. It is used to extract a global feature of a text question. On the other side, a discriminative model is the Long-Short Term Memory (LSTM) based on Continuous Bag-of-Words (CBoW) as a representation of text questions. It is used to extract a local feature of a text question. These two features, a local feature and a global feature, are combined by concatenation. Finally, the multilayer perceptron (MLP) is applied for feature extraction and dimensionality reduction. There are two activation functions, Relu and Softmax, are implemented in this study. Softmax function is used to classify the topic that each text question belongs to. A process of the proposed model can be described by the following sections.

3.2.1 Bag-of-Words (BoW)

BoW is a simple approach to transform each text question into a vector. Although it ignores a grammar and word order, it is able to draw a statistical meaning by considering only the occurrence of each word. The process begins with the dictionary. Then, it counts the occurrence of each word that can be found in the dictionary. Therefore, each text question is transformed into a vector with d dimension where d is the number of words in a dictionary. Each cell represents the number of occurrences of each word. An example of BoW can be depicted as Fig 4. The table consists rows and columns which represent tokenized text questions and words in the dictionary respectively. Each cell is the number of word occurrences in a text question. However, words that is not in the pre-defined dictionary are discarded. The output from LDA is a vector with d dimension where d is the number of words in the dictionary. In this study, there are 683 words included in the dictionary.



	ทำไม	ลบ	บัญชี	ไม่ได้	เกิด	สินค้า	ได้	เลือก	เลิก	ภาษา	ไทย	เมนู	สามารถ	เปลี่ยน	แสดง
ทำไมลบบัญชีไม่ได้	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
สวัสดีค่ะ ลบบัญชีไม่ได้เกิดจากอะไรคะ	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0
อยากลบบัญชีค่ะ ทำไมมันขึ้นว่าไม่ได้คะ	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0
เลิกขายสินค้านี้แล้วหรอคะ ทำไมกดเลือกไม่ได้	1	0	0	1	0	1	0	1	1	0	0	0	0	0	0
มีเมนู ภาษาไทย มั้ย	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0
สามารถเปลี่ยนภาษาได้หรือไม่ อย่างไร	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
อยากเปลี่ยนภาษาที่แสดง	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0
เปลี่ยนภาษาอย่างไร	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0

Figure 4: An example of Bag-of-Words

3.2.2 Latent Dirichlet Allocation (LDA)

LDA is a concept of how to generate documents. Each document consists of two components which are topic and word. A series of words characterizes a topic. Several topics are combined to generate a document. LDA structure can be depicted as Fig 3.4. LDA is categorized as generative model because it is based on the data distribution and probability. In practice, three components are applicable. However, a mixture of words for each topic and a mixture of topics for each document are unknown. Thus, the concept of LDA can be adopted to derive those unknow values. It uses a BoW as a representation of a document because the word co-occurrence can be used to draw a statistical meaning between words and topics.

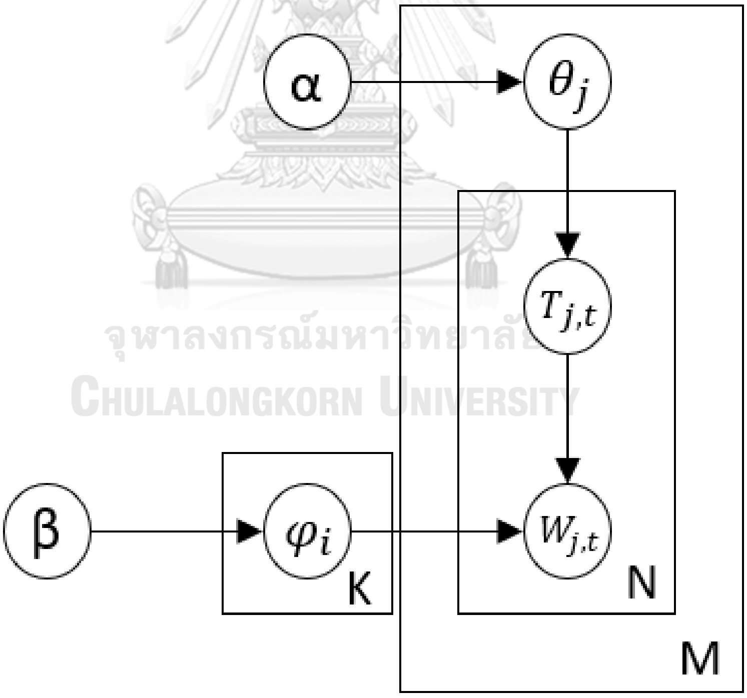


Figure 5: LDA structure

As stated earlier, a generative model is based on the data distribution. LDA is no exception. It is based on the Dirichlet distribution with two hyperparameters, α and β. They control the area of probability density in which α controls the topic distribution over document and β controls the word distribution over topic. When the hyperparameter has a value less than 1, the considered component is more likely to belong to a specific category. On the contrary, when the hyperparameter has a value more than 1, the considered component is more likely to be a mixture of several categories. LDA can be expressed in mathematical formula shown in equation (1). Moreover, the parameters are defined as in table 3.1.

$$\begin{aligned} P(\boldsymbol{W}, \boldsymbol{T}, \boldsymbol{\theta}, \boldsymbol{\varphi}; \alpha, \beta) = \prod_{i=1}^{K} P(\varphi_i; \beta) \prod_{j=1}^{M} P\left(\theta_j; \alpha\right) \\ \times \prod_{t=1}^{N} P\left(T_{j,t} | \theta_j\right) P\left(W_{j,t} | \varphi_{T_{j,t}}\right) \end{aligned} \tag{1}$$

Table 1: The definition of LDA parameters

Parameters	Definitions
M	The number of documents
K	The number of topics
N	The number of words
α, β	Hyperparameters of the Dirichlet distribution
θ_j	The topic distribution of document j
φ_i	The word distribution of topic i
$T_{j,t}$	The topic of word t of document j
$W_{j,t}$	The word t of document j

In this study, the topic probability distribution of each document, θ, is used as a global feature of a text question. However, according to equation (1), it is difficult to derive the value directly from the equation. It requires an integration which is computational expensive. Therefore, the approximate estimation techniques, Gibbs Sampling, is applied to estimate the parameter, θ.

Gibbs samplings is a Markov chain Monte Carlo (MCMC) algorithm used to approximate the marginal distribution by sampling a sequence of observation from a multivariate probability density. In this study, the process begins by assigning each word in every document to a random topic. Then, it reassigns each word to a new topic based on two criteria. Firstly, it considers a topic that this word belongs to the most. Secondly, it considers a topic that this document belongs to the most. These two criteria are added noised in order to keep a randomness. After several iterations, each word belongs to a single topic. This process is used to approximate the topic probability distribution of each document, θ.

3.2.3 Continuous Bag-of-Words (CBoW)

CBoW is a process to transform a text question to a matrix. This process is required for using the Long Short-Term Memory. It reveals a latent semantic of each word $[w_1, w_2, \dots, w_n]$ in text questions. The concept of CBoW is to use surrounding word to predict the targeted word based on the Multilayer Perceptron (MLP). The process begins with a predefined dictionary. Then, it randomly selects a word as a target word, w_i. The surrounding words $[w_{i-2}, w_{i-1}, w_{i+1}, w_{i+2}]$ are fed into the Multilayer Perceptron to predict a target word. The weights in hidden layers are learned during learning process in order to give the correct prediction as much as possible. Therefore, these weights are considered as a sematic representation of each word. The CBoW model can be visualized as shown in Fig 6.

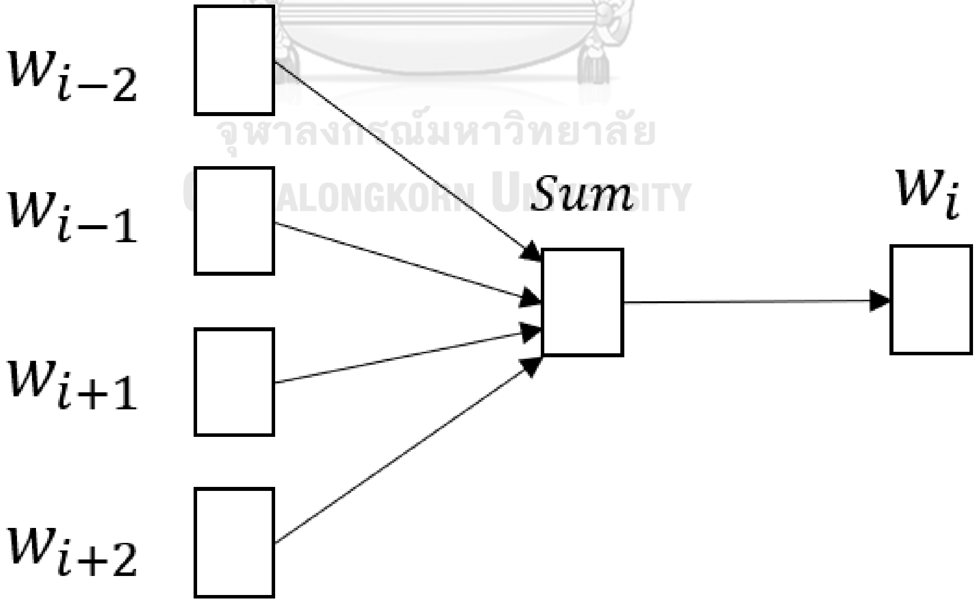


Figure 6: CBoW model structure

3.2.4 Long Short-Term Memory (LSTM)

LSTM inherits from Recurrent Neural Network (RNN) in order to improve the learning efficiency. The problem with RNN is that it suffers from the vanishing gradient problem. This typically happens with the earlier layer of RNN. When LSTM learns from a long sequence, a backpropagation through time makes the gradient shrink. It could be extremely small such that RNN can not learn anything. Therefore, LSTM has a mechanism called gates which are forget gate, input gate, and output gate. They control the amount of information by determining which information should be kept or be discarded. There are three data fed into this cell; h_{t-1} is a hidden state at the previous timestep, c_{t-1} is a cell state at the previous timestep and x_t is an input at the current timestep. These data are calculated to obtained outputs of LSTM cell which are a current hidden state, h_t, and a current cell state, c_t. LSTM cell structure can be visualized as shown in Fig 7.

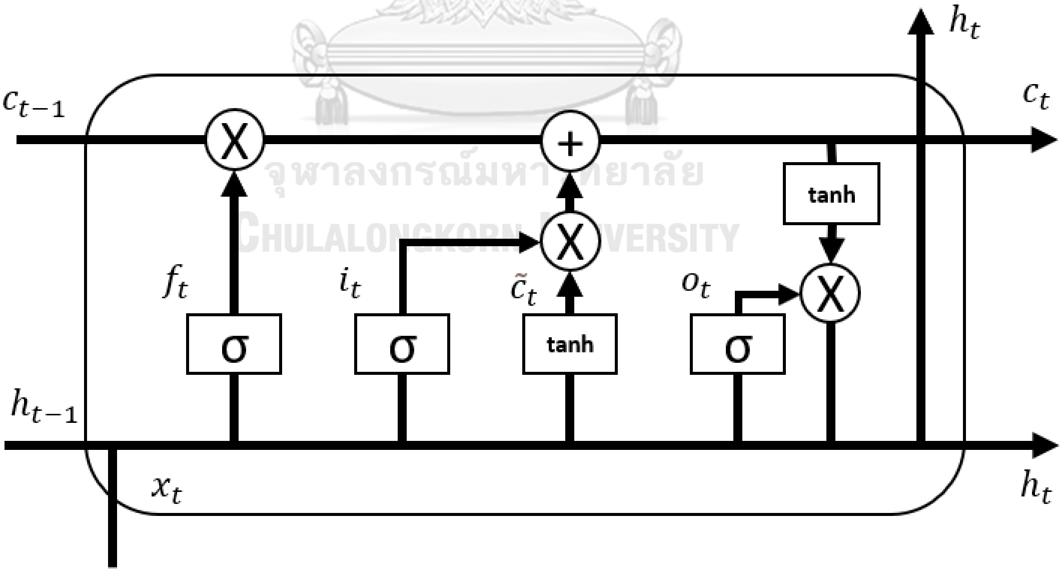


Figure 7: LSTM cell structure

The first gate is a forget gate. It is used to determined how much information from the previous timestep, c_{t-1}, should be kept by using a mechanism of a sigmoid function, $\sigma\,(\cdot)$. Firstly, the inputs, x_t and h_{t-1} are multiplied by their weights separately and are added together with a bias, b_f . Then, a sigmoid function is applied to convert the values between 0 and 1. The result, f_t, is multiplied with c_{t-1} which is the information from the previous timestep. All information will be kept when f_t is 1. On they contrary, all information will be discarded when f_t is 0. Equation (2) represents the forget gate.

$$f_t = \sigma\,(W_f \cdot\ [h_{t-1},\, x_t]\ +\ b_f) \tag{2}$$

The second gate is an input gate. It is used to consider the influence of the inputs x_t and h_{t-1} on the previous cell state, c_{t-1}. It consists of three equations. Firstly, $\tilde{c}_t$ is used to determine the direction which is positive or negative. It is obtained by using a tanh function as shown in equation (3). Secondly, i_t is used to determine the magnitude. It is obtained by using a sigmoid function as shown in equation (4). Lastly, $\tilde{c}_t$ is multiplied with i_t and is added with the result from the forget gate to obtain the current cell state, c_t as shown in equation (5).

$$\tilde{c}_t = \tanh\,(W_{\tilde{c}} \cdot\ [h_{t-1},\, x_t]\ +\ b_{\tilde{c}}) \tag{3}$$

$$i_t = \sigma\,(W_i \cdot\ [h_{t-1},\, x_t]\ +\ b_i) \tag{4}$$

$$c_t =\ f_t\ \cdot\ c_{t-1}\ +\ i_t\ \cdot\ \tilde{c}_t \tag{5}$$

The last gate is an output gate. It determines how much information will be sent out from the LSTM cell. It consists of two equations. Firstly, the inputs, x_t and h_{t-1} are multiplied by their weights separately and are added together with a bias, b_o. Then, a sigmoid function is applied to obtain o_t as described in equation (6). Then, it is multiplied with the adjusted current cell state to obtain the current hidden state, h_t, which represents as the result of LSTM cell as expressed in equation (7).

$$o_t = \sigma\,(W_o \cdot\ [h_{t-1},\, x_t]\ +\ b_o) \qquad (6)$$

$$h_t = \tanh(c_t) \cdot\ o_t \qquad (7)$$

Although, LSTM is able to capture a delay signal from a long text. It extracts features from the first word to the last word sequentially. It could be the case that the last few words have more influence than the first few words. To solve the problem, Bidirectional Long Short-Term Memory (Bi-LSTM) is created by concatenating two LSTM networks. Each network extract features in different perspectives which are forward and backward. In this study, Bi-LSTM is implemented to extract a local feature with 256 dimensions.

3.2.5 Concatenation

Concatenation is a method to concatenate several vectors or matrices into a new one. In this paper, two features which are a global feature with 31 dimensions obtained from LDA and a local feature with 256 dimensions obtained from Bi-LSTM are concatenated. The result is a single vector with 287 dimensions.

3.2.6 Multilayer Perceptron (MLP)

MLP is a simple architecture of artificial neural network. Typically, it is used to extract a feature and also reduce a number of dimensions. The architecture consists of three layers which are input layer, hidden layer, and output layer. It can be expressed in mathematical formula shown in equation (8). The input $[x_1, x_2, \ldots, x_j]$ is fed to the hidden layer where w_{ij} is a weight for each node and w_{i0} is a bias. Then, a non-linear activation function $f(\cdot)$ is applied to the linear combination.

$$o_i = f\left(\sum_j^J w_{ij}x_j + w_{i0}\right) \tag{8}$$

In this study, there are two hidden layers. The first hidden layer is constructed using Relu as the non-linear activation function. The mechanism of Relu can be described with two scenarios. When the input is positive, the output is the same as input. Otherwise, it will be zero. After the first hidden layer is applied, the number of dimensions is reduced to 128. Equation (9) represents Relu activation function.

$$f(z_i) = max(0, z_i) \tag{9}$$

The second hidden layer is constructed using Softmax as the non-linear activation function. In this layer, the input vector is converted to a new vector which all components range between 0 and 1 and must be summed to 1. It can be considered as a probability. Therefore, in this study, Softmax is applied to classify text questions into the correct topic. The result is a vector with 31 dimensions which each component represents a topic number. The topic with the highest probability is selected as a predicted topic. Equation (10) represents Softmax function.

$$f(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} \tag{10}$$

CHAPTER IV
EXPERRIMETAL RESULTS

There are three sections are described in this chapter including dataset preparation, experiments and results.

4.1 Dataset preparation

In this paper, the experiments are conducted on a synthetic data. It was collected from a survey. Respondents were asked to write questions which are suitable with a set of answers from E-commerce FAQ page. To ensure good data quality, data must be obtained from respondents who participated with any E-commerce platform at least once.

The dataset contains 1,796 questions. The questions have length between 2 words and 83 words. The median question length is 11 words. They can be categorized into 5 main classes which are account, logistic, payment, refund and reject. These categories have a total of 31 subclasses. Note that 31 subclasses are equal to 31 answers. Moreover, there are some categories that are marked with a severe case. The questions were separated into training set and test set with a ratio 75:25. However, due to data size, it could not be evaluated with the k-fold cross validation technique. The question was split into ten different versions of training set and test set. The category distribution and the examples of text questions are shown in Table 2 and Table 3 respectively. In Table 2, there is a column named severe case. This column is used to identify whether each case is severe or not. The severe case is marked as 1. The other is marked as 0. Moreover, the reject class represents text questions that are irrelevant to the first four main classes. They were excluded when creating the dictionary and training Latent Dirichlet Allocation model.

Table 2: Class distribution

Main class	Subclass	Severe case	Training set	Test set
Account	ช่องทางที่สามารถใช้ในการสมัครบัญชีผู้ใช้	1	68	16
Account	ไม่สามารถเข้าสู่ระบบได้	1	58	13
Account	การเปลี่ยนชื่อบัญชีและการเปลี่ยนชื่อร้านค้า	0	71	21
Account	การเปลี่ยนเบอร์โทรศัพท์ที่ผู้ไว้กับผู้ให้บริการ	0	45	12
Account	ขั้นตอนและวิธีการเมื่อไม่สามารถจำรหัสผ่านได้	0	40	10
Account	ขั้นตอนและวิธีการเปลี่ยนภาษา	1	46	11
Account	ขั้นตอนและวิธีการยกเลิกบัญชีผู้ใช้	0	34	9
Account	ขั้นตอนและวิธีการขอใบกำกับภาษี	0	65	17
Logistic	โปรโมชั่นค่าขนส่ง	0	57	16
Logistic	วิธีการคำนวณค่าขนส่ง	0	41	11
Logistic	การดำเนินการหากส่งพัสดุไม่สำเร็จ	0	31	8
Logistic	ช่องทางการติดต่อบริษัทขนส่ง	0	43	13
Logistic	การชดเชยร้านค้า กรณีบริษัทขนส่งทำสินค้าสูญหาย	1	40	8
Logistic	การปฏิบัติเมื่อสินค้าที่มาส่งเกิดความเสียหาย	1	29	7
Logistic	วันเวลาทำการของบริษัทขนส่ง	1	61	14
Payment	เงื่อนไขในการเพิ่มบัตรเครดิตหรือบัตรเดบิต	0	42	9
Payment	ช่องทางที่สามารถใช้ในการชำระค่าสินค้า	0	39	10
Payment	การดำเนินการหากลูกค้าไม่ชำระเงินตามเวลาที่กำหนดไว้	0	50	12
Payment	สาเหตุที่สถานะคำสั่งซื้อไม่เปลี่ยนแปลง แม้ลูกค้าจะชำระเงินแล้ว	1	53	13
Payment	ระยะเวลาและเงื่อนไขที่ร้านค้าจะได้รับเงินค่าสินค้า	1	41	8
Payment	ช่องทางชำระเงินปลายทาง จะทำให้ผู้ขายได้รับเงินช้าลงหรือไม่	0	41	10
Payment	สาเหตุและวิธีการดำเนินการเมื่อสินค้าที่ลูกค้า	0	41	11

	ต้องการขาดแคลน			
Payment	ความหมายของค่าธุรกรรมการชำระเงิน	0	59	12
Refund	ขั้นตอนและวิธีการยกเลิกคำสั่งซื้อ	1	61	16
Refund	ขั้นตอนและวิธีการคืนสินค้าประเภทอุปโภคบริโภค	0	46	13
Refund	ระยะเวลาในการคืนเงินค่าสินค้าแก่ผู้ซื้อ	0	37	10
Refund	ค่าใช้จ่ายในการจัดส่งสินค้าคืนร้าค้า	1	34	9
Refund	สาเหตุที่คำสั่งซื้อถูกยกเลิก	1	38	10
Refund	ขั้นตอนและวิธีการคืนสินค้าที่มีขนาดใหญ่	0	36	9
Refund	การขอยืดระยะเวลาในการคืนสินค้า	0	33	9
Reject			63	16
Total			1,443	363

Table 3: The examples of text questions

Main class	Short length	Long length
Account	สมัครยังไง	มีปัญหาเกี่ยวกับการใช้Emailในการลงทะเบียนใช้งาน สามารถลงทะเบียนด้วยวิธีอื่นได้ยังไงบ้างครับ
Account	ลืมรหัสผ่าน	สวัสดีค่ะ พอดีสมัครการใช้งานเกิน1ปีแล้ว ปัจจุบันจำรหัสผ่านไม่ได้ ไม่ทราบว่ามีวิธีไหนที่จะสามารถช่วยแจ้งรหัสผ่านเดิม หรือให้ทางเราเปลี่ยนรหัสผ่านใหม่ได้บ้างคะ
Payment	ทำไมคำสั่งซื้อของฉันจึงถูกยกเลิกอัตโนมัติ	ทำไมคำสั่งซื้อ 20849383 ของผมที่กดสั่งไว้ถึงถูกยกเลิกไปเอง ทั้งๆที่ไม่ได้ทำอะไร ตอนนี้สินค้าที่ผมกดสั่งซื้อไว้เปลี่ยนราคาแล้ว ผมต้องการสินค้าราคาเดิมที่ผมกดสั่งไป
Payment	ทำไมจ่ายแล้วสถานะไม่เปลี่ยน	สวัสดีครับ เมื่อวานผมได้ออร์เดอร์คำสั่งซื้อ AH032559J7B0 ไป กดจ่ายด้วยบัตรเครดิตเรียบร้อยแล้ว มี sms แจ้งมาแล้วว่า เงินผมถูกตัด ทว่าออร์เดอร์ยังไม่เปลี่ยนสถานะเป็นชำระเงินแล้วเลย ช่วยตรวจสอบให้หน่อยครับ
Refund	ของที่ส่งมาบุบ มีนโยบายยังไงบ้างคะ	ดิฉันได้สั่งแยม 1 หีบ เมื่อวันที่ 10 มิถุนายนยน จากร้านดีดี ปรากฏว่ามีแยมแตก 2 ขวด ต้องการให้ทางร้านส่งของมาเปลี่ยนใหม่ให้ด้วย ดิฉันต้องส่งขวดที่แตกคืนไหมคะ
Refund	ถ้าอยากคืนสินค้าต้องทำยังไงบ้าง	สินค้าไม่ตรงตามรายละเอียดที่แจ้งไว้ สามารถส่งคืนสินค้าและขอรับเงินคืนได้มั้ยคะ

4.2 Evaluation Methods

4.2.1 Confusion matrix

To evaluate the performance of proposed model, the confusion matrix is used to depict the results. It compares the prediction results with their actual classes. Typically, the confusion matrix of size nxn can be used to evaluate a test set with n classes. Each column represents the predicted class, whereas each row represents the actual class. The simplest form of confusion matrix consists of four elements corresponding to four values. True Positive (TP) is the value representing the number of correct predictions on a specific class, False Positive (FP) is the value representing the number of wrong predictions on a specific class, True Negative (TN) is the value representing the number of correct predictions on non-specific class, and False Negative is the value representing the wrong prediction on non-specific class. This confusion matrix can be visualized as shown in Table 4.

Table 4: Confusion Matrix

		Predicted value	
		Positive	Negative
Actual value	Positive	True Positive	False Negative
	Negative	False Positive	True Negative

To be more precise, the proposed model is used to obtain the answer regarding the given questions. There are 31 answers or 31 classes. The confusion matrix is used to measures the quality of prediction on the considered class. The considered class is denoted as positive, whereas other classes are denoted as negative. Furthermore, to compute the overall model's performance, the macro-averaging is applied by calculating the performance over each class separately. Then, they are averaged to obtain the overall result. Table 5 represents 4-classes confusion matrix with proxy values. It will be used as a sample of how to calculate each measurement values.

Table 5: 4-classes confusion matrix

	A	B	C	D
A	8	3	3	0
B	2	12	2	1
C	1	4	10	3
D	0	2	5	10

The value in table 5 is a mock up data. There are 4 classes which are A, B, C and D with the total number of 66 samples. The right most column represents the actual classes, whereas the first row represents the prediction classes. For example, it can be interpreted as; the model correctly predicts 8 samples of class A and wrongly predict to class A with 2 samples for class B, 1 sample for class C and 0 for class D.

4.2.2 Accuracy

The basic metric measures the overall correctness by proportionating the number of correct predictions over the number of all predictions. it can be calculated as equation (11).

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions} \tag{11}$$

According to table 5, the number of correct predictions is 40 and the total number of predictions is 66. Therefore, the accuracy is computed by dividing 40 with 66. The accuracy is 60.06%.

4.2.3 Precision

Precision is used to measure the correctness of prediction results with a specific class. The metric describes how likely the prediction of a specific class is correct. It can be formulated as shown in equation (12).

$$Precision = \frac{TP}{TP+FP} \tag{11}$$

According to table 5, the precisions of each class are calculated separately which are 72.73% for class A, 60% for class B, 50% for class C and 71.29% for class D. These 4 values are averaged. Therefore, the overall precision of this model is 63.51%.

4.2.4 Recall

Recall measures the accuracy of a specific class. It considers the number of correct predictions over the total number of records belonging to a specific class. Equation (12) represents the formula.

$$Recall = \frac{TP}{TP+FN} \tag{12}$$

According to table 5, the recalls of each class are calculated separately which are 57.14% for class A, 70.59% for class B, 55.56% for class C and 58.82% for class D. These 4 values are averaged. Therefore, the overall recall of this model is 60.53%.

4.2.3 F1-score

F1-score bases on two existing metrices which are precision and recall. It uses a concept of harmonic mean to combine them into a single value. It can be calculated as shown in (13)

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (12)$$

According to table 5, the F1-scores of each class are calculated separately which are 64% for class A, 64.87% for class B, 52.63% for class C and 64.46% for class D. These 4 values are averaged. Therefore, the overall F1-score of this model is 61.49%.

4.3 Experiments and results

In this study, the experiments were conducted on three perspectives. Firstly, the hyperparameter tuning was to increase the performance of the proposed model. Secondly, the comparative experiment was conducted to verify that the proposed model is able to achieve the better performance than the existing text classification models. Finally, the last perspective involves with the results of individual classes.

4.3.1 Hyperparameter tuning

Hyperparameter tuning is a trial-and-error process to find the most suitable set of hyperparameters that helps a model to reach the best performance. In this study, the process is based on two hyperparameters of Bi-LSTM which are the number of hidden nodes and the dropout rate. Dropout is a technique to avoid the overfitting problem. It adds a randomness to a model by ignoring some nodes during the training process. It is an optional technique that could or could not increase a performance. On the other hand, the number of hidden nodes determines how much complexity the model will be. The more the number of hidden nodes, the more the complexity. Either simple model or complex model has its advantages and disadvantages. A complex model is capable to capture complicate patterns. However, it is more difficult than a simple model to capture simple patterns due to a number of hidden nodes that is need to be adjusted. Furthermore, the more the complexity, the more the computation cost. A complex model requires more time to train and to execute than a simple model. Therefore, to find the proper number of hidden nodes with efficient time consumption, the number of hidden nodes were conducted firstly. Then, it was used to find the proper dropout rate. The evaluation measurements used to find the proper hyperparameters are accuracy, precision, recall, and F1-score. In this study, hyperparameter tuning were conducted on both training and test dataset. The results were presented as shown in table 6, table 7, table 8 and table 9.

In table 6 and table 7, there are 4 different numbers of hidden nodes which are 32, 64, 128 and 256. The experiments were conducted with Bi-LSTM with 0 dropout rate. The results show similar pattern for both training and test datasets. The performance increases as the number of hidden nodes increases. Until it exceeds its peak at 128 hidden nodes, the performance drops. Therefore, the optimal number of hidden nodes is 128.

Table 6: Experimental results with respect to different number of hidden nodes conducted on training dataset (in percentage)

Hidden nodes	Accuracy	Precision	Recall	F1
32	90.25	90.01	89.98	89.95
64	95.55	94.48	95.05	94.76
128	**97.92**	**98.01**	**97.56**	**97.84**
256	92.74	93.12	92.55	92.83

Table 7: Experimental results with respect to different number of hidden nodes conducted on test dataset (in percentage)

Hidden nodes	Accuracy	Precision	Recall	F1
32	79.20	78.88	77.79	78.32
64	79.58	79.79	78.35	79.15
128	**81.05**	**81.02**	**80.08**	**80.68**
256	74.60	74.74	73.58	74.15

According to the table 8 and table 9, to find the proper dropout rate, the experiment was conducted on a Bi-LSTM with 128 hidden nodes. It was ranging from 0 to 0.9 with step size of 0.1. The result indicated the inverse direction between the performance of training and test datasets as expected. The dropout affected the model by introducing a noise. This noise helped the model avoid the overfitting problem. Therefore, when the dropout rate increases, the training performance drops but the test performance increases. The highest performance for test dataset can be obtained when dropout rate is around 0.7. It achieved the highest score on accuracy and F1-score. Moreover, the dropout rates at 0.6 and 0.8 also yielded the similar level of performance but the dropout rate at 0.9 yielded the lowest level of performance. It indicated that the model was difficult to learn any patterns when dropout rate was too high because, for each iteration, there are only 10% of the hidden nodes were adjusted.

Table 8: Experimental results with respect to different number of dropout rates conducted on training dataset (in percentage)

Dropout rate	Accuracy	Precision	Recall	F1
0.0	**97.92**	**98.01**	**97.56**	**97.84**
0.1	97.19	97.11	96.93	97.02
0.2	96.83	96.81	96.39	96.60
0.3	95.48	95.23	94.96	95.09
0.4	93.78	94.35	94.32	94.33
0.5	91.37	91.10	91.74	91.42
0.6	88.98	89.06	89.66	89.36
0.7	87.21	87.72	87.84	87.78
0.8	85.70	85.92	85.88	85.90
0.9	82.75	82.85	83.23	83.04

Table 9 Experimental results with respect to different number of dropout rates conducted on test dataset (in percentage)

Dropout rate	Accuracy	Precision	Recall	F1
0.0	81.05	81.02	80.08	80.68
0.1	82.46	82.45	81.49	81.96
0.2	82.19	82.49	81.06	81.77
0.3	83.74	83.95	82.78	83.35
0.4	83.63	83.46	82.89	83.17
0.5	79.06	79.15	78.35	78.72
0.6	84.07	**84.33**	83.51	83.91
0.7	**84.43**	84.21	83.68	**83.94**
0.8	84.21	84.03	**83.70**	83.86
0.9	78.56	78.05	77.21	77.60

4.3.2 Comparative experiment

To verify the performance of the proposed model, three existing neural network-based models, including traditional LSTM, Recurrent Neural Network with Latent Dirichlet Allocation (RCL) and Bi-LSTM-CNN, were used in this experiment. The experiment was conducted on three scenarios. For the first scenario, the test set was used to evaluate all four models in four metrics. Table 10 presents the overall performance of each model. For the second scenario, the test set was filter to consider only the severe cases, Table 11 represents the model performance on the sever cases. On the last scenario, the test set was separated into several groups based on text length to explore the effect of text length on the model performance. The result can be visualized as shown in Figure 8.

Table 10: Experimental results of the proposed model compared with those of three existing models (in percentage) and average computational time (in milliseconds)

Model	Accuracy	Precision	Recall	F1	Time
RCL [8]	80.08	80.53	78.92	79.71	92.50
Bi-LSTM	81.61	81.26	80.65	80.95	**24.18**
Bi-LSTM-CNN [9]	82.80	82.74	82.04	82.38	77.48
Proposed model	**84.43**	**84.21**	**83.68**	**83.94**	25.87

Table 11: Experimental results of the proposed model compared with those of three existing model (in percentage) on the severe cases

Model	Accuracy	Precision	Recall	F1
RCL [8]	78.70	87.88	77.02	82.05
Bi-LSTM	81.05	88.67	80.02	84.10
Bi-LSTM-CNN [9]	82.63	90.00	81.63	85.56
Proposed model	**84.40**	**90.11**	**83.15**	**86.48**

As shown in Table 10 and Table 11, the experiments were conducted on the whole dataset at once. The proposed model is able to achieve the highest performance on all evaluation measurements. Furthermore, the time required to execute is nearly the same as the baseline model which is Bi-LSTM. On the other hand, the execution time for Bi-LSTM-CNN and RCL is more than baseline around 3 to 4 times. The reason why it takes more time is that they rely on Bi-LSTM's one-to-one architecture. It means that there are outputs from every LSTM cell. Then, they were fed into either CNN or max pooling layer which is time consuming.

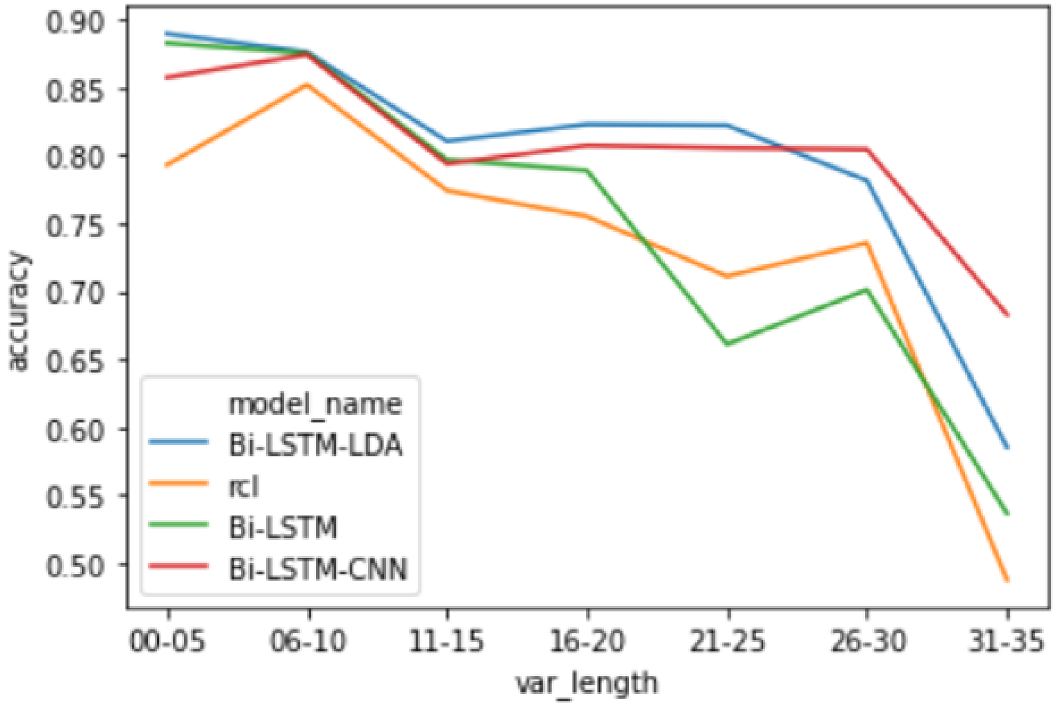


Figure 8: Accuracies based on different data lengths

According to Figure 8, the experiments were conducted on different length of text questions. Each group has the interval length of 5 words starting from 0 words to 35 words. Note that the shortest question length has 2 words. Therefore, there are total of 7 groups. The results provided insightful information. From 2 words to 25 words, the proposed model achieved the highest performance. However, when the question length exceeded 25 words, the performance of the proposed model dropped below Bi-LSTM-CNN. The reason is that LDA is a generative model. It required a number of similar text questions in order to construct a distribution. However, text question with more than 25 words are rare cases.

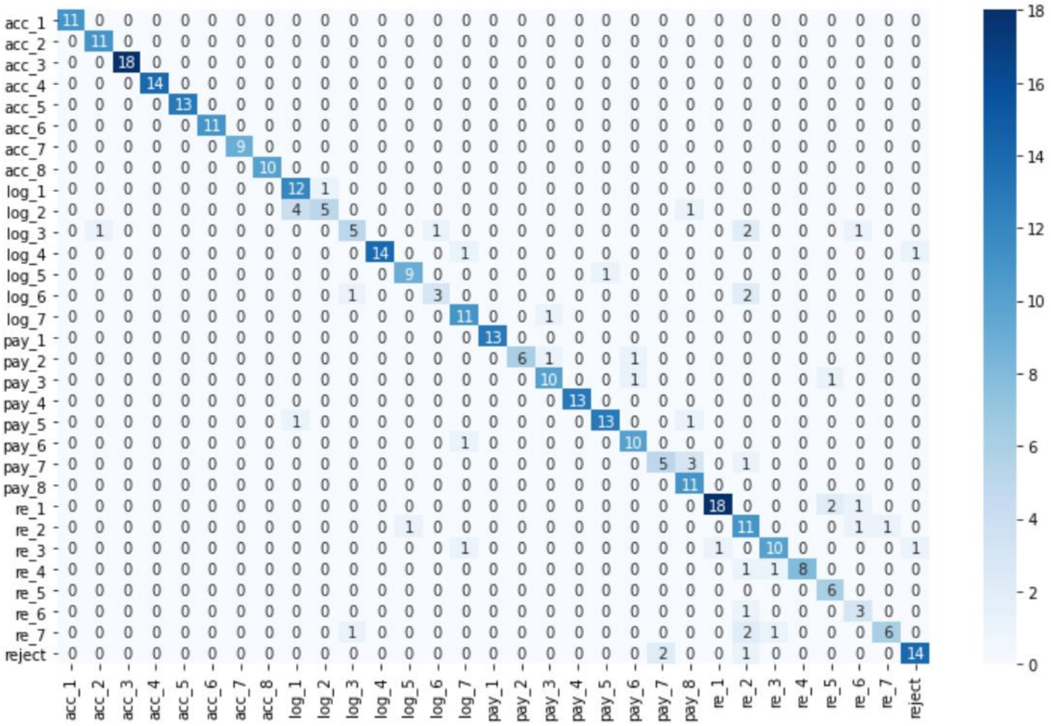
Moreover, it indicated that the proper question length is around 2 to 10 words. This range provided precise information required to obtain the correct answer. After 10 words, it usually consisted of additional information which is specific to each case. Therefore, an additional information did not help the model but it acted like a noise. Samples are provided in Table 12.

Table 12: Samples of proper and improper question length

Main class	Proper length	Improper length
Account	ลืมรหัสผ่าน	สวัสดีครับ ผมเข้าสู่ระบบด้วยแอคเค้าของผมไม่ได้ ทดสอบด้วยพาสเวิร์ดหลายๆแบบแล้วแต่ก็ไม่สำเร็จ ไม่แน่ใจว่าแก้ไขยังไงได้บ้างครับ
Account	ต้องการสมัครสมาชิก	พอดีว่าทีวีมีโปรโมชั่นลดครึ่งหนึ่งเมื่อเป็นสมาชิก แต่ผมยังไม่ได้เป็น ผมสามารถสมัครสมาชิกได้ที่ไหนบ้างครับ
Payment	ทำไมคำสั่งซื้อของฉันจึงถูกยกเลิกอัตโนมัติ	ทำไมคำสั่งซื้อ 20849383 ของผมที่กดสั่งไว้ถึงถูกยกเลิกไปเองทั้งๆที่ไม่ได้ทำอะไร ตอนนี้สินค้าที่ผมกดสั่งซื้อไว้เปลี่ยนราคาแล้ว ผมต้องการสินค้าราคาเดิมที่ผมกดสั่งไป
Payment	ทำไมจ่ายแล้วสถานะไม่เปลี่ยน	สวัสดีครับ เมื่อวานผมได้ออร์เดอร์คำสั่งซื้อ AH032559J7B0 ไป กดจ่ายด้วยบัตรเครดิตเรียบร้อยแล้ว มี sms แจ้งมาแล้วว่าเงินผมถูกตัด ทว่าออร์เดอร์ยังไม่เปลี่ยนสถานะเป็นชำระเงินแล้วเลย ช่วยตรวจสอบให้หน่อยครับ
Logistic	บริษัทขนส่งทำอย่างไรเมื่อส่งสินค้าไปที่อยู่ผู้รับไม่ได้	พอดีสินค้าที่ดิฉันสั่งซื้อไปจะถูกส่งมาให้ดิฉันวันนี้ แต่ วันนี้ดิฉันไม่อยู่บ้าน และ ไม่มีคนรับสินค้าแทนได้ ทางขนส่งจะมาส่งให้ดิฉันใหม่อีกครั้งในภายหลังได้ไหมคะ
Logistic	ทำไมฉันจึงได้รับสินค้าล่าช้า	ผมได้สั่งซื้อ ที่โกนหนวดไฟฟ้า เมื่อวันที่ 1 มิถุนายนยน เวลาผ่านไป 10 วันแล้วยังไม่ได้รับสินค้าเลย
Refund	ของที่ส่งมาบุบ มีนโยบายยังไงบ้างคะ	ดิฉันได้สั่งแยม 1 หีบ เมื่อวันที่ 10 มิถุนายนยน จากร้านดีดีปรากฏว่ามีแยมแตก 2 ขวด ต้องการให้ทางร้านส่งของมาเปลี่ยนใหม่ให้ด้วย ดิฉันต้องส่งขวดที่แตกคืนไหมคะ
Refund	ถ้าจะส่งสินค้าคืนแล้วจะได้ค่าส่งคืนไหมคะ	สวัสดีค่ะ ทางเรามีการซื้อเสื้อจากร้าน XYZ แต่เนื่องจากสินค้าไม่ตรงกับขนาดที่บอกในรายละเอียด ไม่ทราบว่าจะขอค่าขนส่งสินค้าที่คืนไปด้วยได้ไหม

4.3.3 Result of individual classes

The confusion matrix was conducted to understand the outcome clearly on each individual class. Each column represents the predicted class while each row represents the actual class. In this thesis, there are 31 classes as shown in Figure 9. The result indicated that the wrong predictions usually are in the same main class because the word choices are quite similar. Table 13 represents samples of wrong predictions.



	acc_1	acc_2	acc_3	acc_4	acc_5	acc_6	acc_7	acc_8	log_1	log_2	log_3	log_4	log_5	log_6	log_7	pay_1	pay_2	pay_3	pay_4	pay_5	pay_6	pay_7	pay_8	re_1	re_2	re_3	re_4	re_5	re_6	re_7	reject
acc_1	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
acc_2	0	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
acc_3	0	0	18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
acc_4	0	0	0	14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
acc_5	0	0	0	0	13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
acc_6	0	0	0	0	0	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
acc_7	0	0	0	0	0	0	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
acc_8	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
log_1	0	0	0	0	0	0	0	0	12	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
log_2	0	0	0	0	0	0	0	0	4	5	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
log_3	0	1	0	0	0	0	0	0	0	0	5	0	0	1	0	0	0	0	0	0	0	0	0	0	2	0	0	0	1	0	0
log_4	0	0	0	0	0	0	0	0	0	0	0	14	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
log_5	0	0	0	0	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
log_6	0	0	0	0	0	0	0	0	0	0	1	0	0	3	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0
log_7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
pay_1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
pay_2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	1	0	0	1	0	0	0	0	0	0	0	0	0	0
pay_3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	1	0	0	0	0	0	0	1	0	0	0
pay_4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0	0	0	0	0	0	0	0	0	0	0
pay_5	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	13	0	0	1	0	0	0	0	0	0	0	0
pay_6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0
pay_7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	3	0	1	0	0	0	0	0	0
pay_8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0	0	0	0	0
re_1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	18	0	0	0	2	1	0	0
re_2	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	11	0	0	0	1	1	0
re_3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	10	0	0	0	0	1
re_4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	8	0	0	0	0
re_5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0
re_6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	3	0	0
re_7	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	2	1	0	0	0	6	0
reject	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	1	0	0	0	0	0	14

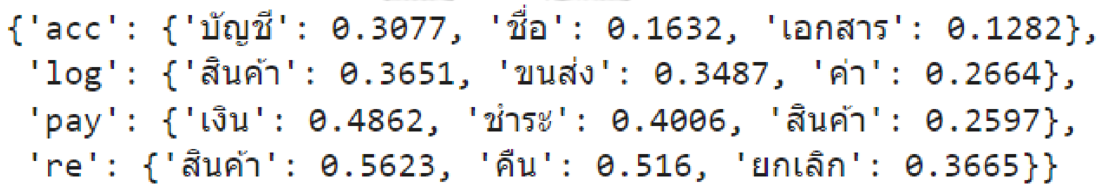
Figure 9: Confusion matrix based on all classes

Table 13: Samples of wrong prediction

Actual class	Predicted class	Question
วิธีการคำนวณค่าขนส่ง	โปรโมชั่นค่าขนส่ง	ทำไมค่าส่งแพง
การดำเนินการหากส่งพัสดุไม่สำเร็จ	ขั้นตอนและวิธีการคืนสินค้าประเภทอุปโภคบริโภค	ของยังไม่มาส่งเลย ทำไมขึ้นว่าคืนไปแล้ว
ช่องทางชำระเงินปลายทาง จะทำให้ผู้ขายได้รับเงินช้าลงหรือไม่	ระยะเวลาและเงื่อนไขที่ร้านค้าจะได้รับเงินค่าสินค้า	อยากทราบว่าการชำระเงินโดยเก็บเงินปลายทางจะมีผลต่อระยะเวลาในการได้รับเงินหรือไม่
ช่องทางชำระเงินปลายทาง จะทำให้ผู้ขายได้รับเงินช้าลงหรือไม่	ระยะเวลาและเงื่อนไขที่ร้านค้าจะได้รับเงินค่าสินค้า	ถ้าลูกค้าต้องการชำระเงินปลายทาง ทางร้านจะได้รับเงินช้ากว่าแบบอื่นไหม
การดำเนินการในกรณีที่ลูกค้าไม่ชำระเงินตามเวลาที่กำหนด	สาเหตุที่คำสั่งซื้อถูกยกเลิกโดยผู้ขาย	ยังไม่ทันได้จ่ายเลย ทำไมคำสั่งซื้อถูกยกเลิกอัตโนมัติ

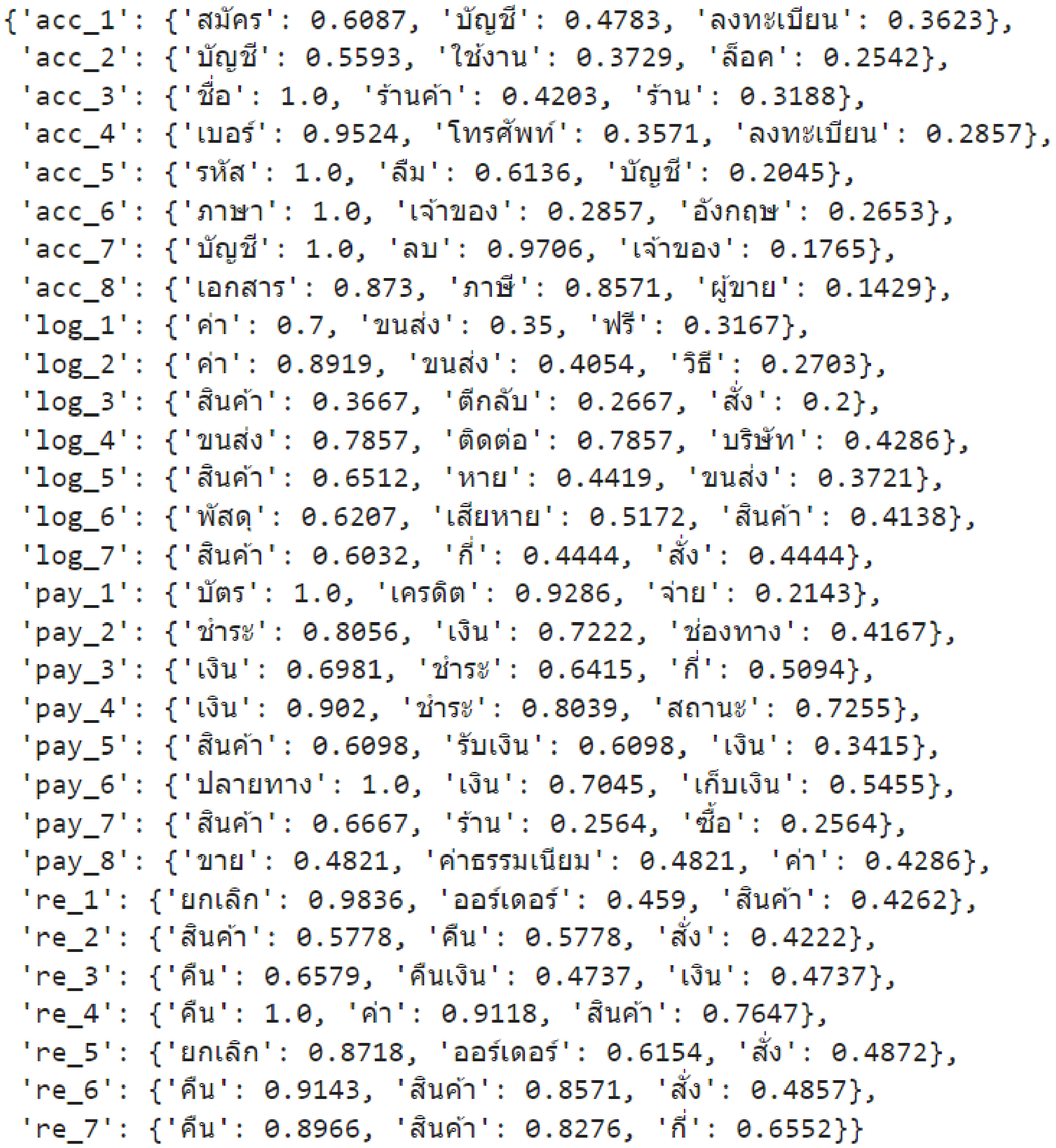
For wrong prediction, table 13 indicated two reasons why the model cannot achieve the accuracy beyond 90%. Firstly, the questions have ambiguity. It cannot clearly distinguish among classes. The second reason is that subclass could be considered as a subset of other subclasses. These two reasons might not be handled by Latent Dirichlet Allocation (LDA). LDA is a generative model which depends on the word occurrence but not on the word semantic.

Furthermore, the next investigation was conducted to find whether there are unique keywords for each class or not. To find the keywords, the number of questions containing a particular word is counted and divided by the total number of questions in that class. The keywords with first three highest ratios were revealed. There are two perspectives provided in this study. Firstly, the investigation was conducted on the main classes. The results indicated that the account class can be clearly distinguish from other classes because there was no keyword co-occurrence with other classes as shown in Figure 10. Nevertheless, when investigated on the subclasses, most of the keywords were interchangeably among subclasses as shown in Figure 11. Therefore, it can be concluded that there were no unique keywords for each subclass. Note that in the figure, the abbreviation is used to represent classes which are acc for account, log for logistic, pay for payment and re for refund.



```
{'acc': {'บัญชี': 0.3077, 'ชื่อ': 0.1632, 'เอกสาร': 0.1282},
 'log': {'สินค้า': 0.3651, 'ขนส่ง': 0.3487, 'ค่า': 0.2664},
 'pay': {'เงิน': 0.4862, 'ชำระ': 0.4006, 'สินค้า': 0.2597},
 're': {'สินค้า': 0.5623, 'คืน': 0.516, 'ยกเลิก': 0.3665}}
```

Figure 10: Keyword in main classes



```
{'acc_1': {'สมัคร': 0.6087, 'บัญชี': 0.4783, 'ลงทะเบียน': 0.3623},
 'acc_2': {'บัญชี': 0.5593, 'ใช้งาน': 0.3729, 'ล็อค': 0.2542},
 'acc_3': {'ชื่อ': 1.0, 'ร้านค้า': 0.4203, 'ร้าน': 0.3188},
 'acc_4': {'เบอร์': 0.9524, 'โทรศัพท์': 0.3571, 'ลงทะเบียน': 0.2857},
 'acc_5': {'รหัส': 1.0, 'ลืม': 0.6136, 'บัญชี': 0.2045},
 'acc_6': {'ภาษา': 1.0, 'เจ้าของ': 0.2857, 'อังกฤษ': 0.2653},
 'acc_7': {'บัญชี': 1.0, 'ลบ': 0.9706, 'เจ้าของ': 0.1765},
 'acc_8': {'เอกสาร': 0.873, 'ภาษี': 0.8571, 'ผู้ขาย': 0.1429},
 'log_1': {'ค่า': 0.7, 'ขนส่ง': 0.35, 'ฟรี': 0.3167},
 'log_2': {'ค่า': 0.8919, 'ขนส่ง': 0.4054, 'วิธี': 0.2703},
 'log_3': {'สินค้า': 0.3667, 'ตีกลับ': 0.2667, 'สั่ง': 0.2},
 'log_4': {'ขนส่ง': 0.7857, 'ติดต่อ': 0.7857, 'บริษัท': 0.4286},
 'log_5': {'สินค้า': 0.6512, 'หาย': 0.4419, 'ขนส่ง': 0.3721},
 'log_6': {'พัสดุ': 0.6207, 'เสียหาย': 0.5172, 'สินค้า': 0.4138},
 'log_7': {'สินค้า': 0.6032, 'กี่': 0.4444, 'สั่ง': 0.4444},
 'pay_1': {'บัตร': 1.0, 'เครดิต': 0.9286, 'จ่าย': 0.2143},
 'pay_2': {'ชำระ': 0.8056, 'เงิน': 0.7222, 'ช่องทาง': 0.4167},
 'pay_3': {'เงิน': 0.6981, 'ชำระ': 0.6415, 'กี่': 0.5094},
 'pay_4': {'เงิน': 0.902, 'ชำระ': 0.8039, 'สถานะ': 0.7255},
 'pay_5': {'สินค้า': 0.6098, 'รับเงิน': 0.6098, 'เงิน': 0.3415},
 'pay_6': {'ปลายทาง': 1.0, 'เงิน': 0.7045, 'เก็บเงิน': 0.5455},
 'pay_7': {'สินค้า': 0.6667, 'ร้าน': 0.2564, 'ซื้อ': 0.2564},
 'pay_8': {'ขาย': 0.4821, 'ค่าธรรมเนียม': 0.4821, 'ค่า': 0.4286},
 're_1': {'ยกเลิก': 0.9836, 'ออร์เดอร์': 0.459, 'สินค้า': 0.4262},
 're_2': {'สินค้า': 0.5778, 'คืน': 0.5778, 'สั่ง': 0.4222},
 're_3': {'คืน': 0.6579, 'คืนเงิน': 0.4737, 'เงิน': 0.4737},
 're_4': {'คืน': 1.0, 'ค่า': 0.9118, 'สินค้า': 0.7647},
 're_5': {'ยกเลิก': 0.8718, 'ออร์เดอร์': 0.6154, 'สั่ง': 0.4872},
 're_6': {'คืน': 0.9143, 'สินค้า': 0.8571, 'สั่ง': 0.4857},
 're_7': {'คืน': 0.8966, 'สินค้า': 0.8276, 'กี่': 0.6552}}
```

Figure 11: Keywords in subclasses

CHAPTER V
CONCLUSION

5.1 Conclusion

In summary, the thesis has proposed a variable-length text classification model. It is a fusion of two existing classification models which are Latent Dirichlet Allocation (LDA) with Bag-of-Words (BoW) and Bidirectional Long Short-Term Memory (Bi-LSTM) network with Continuous Bag-of-Words (CBoW). The reason to combine them together is that they extract different levels of features. LDA extracts a global feature, whereas Bi-LSTM extract a local feature. Two features are concatenated and fed into a multilayer perceptron (MLP) to predict the answer of the text questions.

To construct the proposed model, hyperparameter tuning is required to achieve the highest performance. In this study, two hyperparameters of Bi-LSTM, including the number of hidden node and dropout rate, were tuned. Both hyperparameters indicated that the performance falls in the law of diminishing returns. The optimal values for the number of hidden nodes and dropout rate are 128 and 0.7, respectively.

On the comparative experiments, the proposed model was against three existing model which are Bi-LSTM, RCL, Bi-LSTM-CNN. When experiments were conducted on a whole dataset, it achieved the highest performance on all evaluation matrices which are 84.43% accuracy, 84.21% precision, 83.68% recall, and 83.94% F1-score. On the second experiment, the models were evaluated based on different length intervals. The performance dropped below Bi-LSTM-CNN when a question length exceeds 25 words. The reason is an excessive detail related to a specific case. This excessive detail acts like a noise that makes difficulty in classification. The proposed model relied on LDA which extracts a feature by using every word in a text question all at once, whereas CNN extracts a feature by using filters with a specific word length. Therefore, CNN can deal with the long text questions slightly better than the proposed model. Moreover, the confusion matrix indicates that the wrong predictions usually occur within the same main class due to the similar text characteristic.

5.2 Future work

To improve the performance, future works might be required as follows:

1. Instead of using a synthetic data, the actual data could be explored and used to reveal more semantic.
2. A feature selection which keeps only significant words could be added and proceeded to increase the performance of classification model.

REFERENCES

1. *E-commerce Payment Trends: Thailand*. 2019 [cited 2020 September 18]; Available from: https://www.jpmorgan.com/europe/merchant-services/insights/reports/thailand.
2. *How many words make a sentence*. 2016 [cited 2020 September 18]; Available from: https://techcomm.nz/Story?Action=View&Story_id=106.
3. Sarakit, P., et al., *Classifying Emotion in Thai Youtube Comments*, in *Proceeding of the the 2015 6th International Conference of Information and Communication Technology for Embedded System (IC-TCTES)*. 2015: Hua Hin, Thailand. p. 1-5.
4. Lin, Y. and J. Wang, *Research on Text Classification Based on SVM-KNN*, in *Proceeding of the 2014 IEEE 5th International Conference on Software Engineering and Service Science*. 2014: Beijing, China. p. 842-844.
5. Liu, T., *Sparse Topic Model for Text Classification*, in *Proceeding of the 2013 International Conference on Machine Learning and Cybernetics*. 2013: Tianjin, China. p. 1916-1920.
6. Chen, Q., L. Yao, and J. Yang, *Short Text Classification Based on LDA Topic Model*, in *Proceeding of the 2016 International Conference on Audio, Language and Image Processing (ICALIP)*. 2016: Shanghai, Chaina. p. 749-753.
7. Muangkammuen, P., Narong Intiruk, and K.R. Saikaew, *Automated Thai-FAQ Chatbot using RNN-LSTM*, in *Proceeding of the 2018 22nd International Computer Science and Engineering Conference (ICSEC)*. 2018: Chiang Mai, Thailand. p. 1-4.
8. Yan, Y. and K. Zheng, *Text Classification Model Based on Multi-level Topic Feature Extraction*, in *Proceeding of the 2020 IEEE 6th International Conference on Computer and Communications*. 2020: Chengdu, China. p. 1661-1665.
9. Li, C.L.G.Z.Z., *News Text Classification Based on Improved Bi-LSTM-CNN*, in *Proceeding of the 2018 9th International Conference on Information Technology in Medicine and Education*. 2018: Hangzhou, China. p. 890-893.

VITA

NAME	Wasu Chunhasomboon
DATE OF BIRTH	16 March 1996
PLACE OF BIRTH	Thailand
INSTITUTIONS ATTENDED	B.B.A., Banking and Finance, Chulalongkorn University, 2018.
HOME ADDRESS	88/155 The Empire Place, Naradhiwas Rajanagarindra, Yannawa, Sathon, Bangkok 10120