*ICIC Express Letters*
Volume **13**, Number **8**, August **2019**

ICIC International ©2019 ISSN 1881-803X
pp. **653–662**

# REVEAL THE CUSTOMER BEHAVIOR FROM BUSINESS SMS TEXT USING NAMED ENTITY RECOGNITION

Siti Mariyah[1], Indrajani Sutedja[2], Ranu Yulianto[3] and Setia Pramana[1]

[1]Center for Computational Statistics Studies
STIS Polytechnic of Statistics
Jl. Otto Iskandardinata No. 64C, Bidaracina, Jatinegara, Jakarta 13330, Indonesia
{ sitimariyah; setia.pramana }@stis.ac.id

[2]Information Systems Department
School of Information Systems
Bina Nusantara University
Jl. K. H. Syahdan, No. 9, Kemanggisan, Palmerah, Jakarta 11480, Indonesia
indrajani@binus.ac.id

[3]Statistics Indonesia
Kabupaten Wakatobi, Sulawesi Tenggara 93795, Indonesia
ranu.yulianto@bps.go.id

Received December 2018; accepted February 2019

Abstract. *Texts owned by an organization can provide early warning of risk and compliance issues by bringing new insight into how the customer acts, what the interest of each customer is, how the customer consumes the business product, and how the customers' profiles are. The challenge is how to take out the potential benefit of texts. Text mining is a solution. In this paper, we report our research in developing text mining application namely Named Entity Recognition (NER) applied to the collection of short message service owned by a marketing agency. We built NER modules which can perform NER modeling and prediction to extract important entities stated in each SMS text from three SMS text collections (Bank X, Bank Y, and Accommodation Reservation Company Z). We constructed three NER models by using three SMS corpora. The NER models can predict and extract entities from 195,036 SMS texts of Bank X with accuracy 99.66%, 26,072 SMS texts of Bank Y with accuracy 98.88% and 525,121 SMS texts of Accommodation Reservation Company Z with accuracy 96.20%. The extracted entities are analyzed to reveal the customer behavior and to get the new insight of business.*
**Keywords:** Text mining, Business SMS text, Named entity recognition

1. **Introduction.** Texts are generated every second which are either in the digital or non-digital form. The text comes from many sources such as internal documents, reports, forms and notes in the customer relationship management, Short Message Service (SMS) text, customer's emails, customer's social media. Those texts will bring new insight into how the customer acts, what the interest of each customer is, how the customer consumes the business product, and how the customers' profiles are. Texts offer a detailed understanding of business that will lead in better decisions making or the opportunity to create new business products which match the customers' preferences. The challenge is how to take out the potential benefit of texts and text mining is solution for the challenge. Text mining, also referred to as text analytics, is the process of exploring and analyzing huge collection of text to derive high-quality information. Text mining applications in business domain have been dominated in the area of sentiment analysis, spam filtering, information extraction, and document clustering.

DOI: 10.24507/icicel.13.08.653

We studied some previous researches which relate to our study. Mostafa [2] conducted both a qualitative and quantitative methodology in analyzing 3516 tweets to understand the sentiment of customers towards well-known brands such as Nokia, T-Mobile, IBM, KLM, and DHL. Netzer et al. utilized the user-generated content to describe market structures and competitive landscape insights [3]. They used a large collection of texts from sedan automobile forums on Edmunds.com and diabetes drug forums (from five forum websites). By using text mining apparatus and social network analysis, they succeeded in exploring and explaining sedan cars market structures, and adverse drug reactions include consumer's perceptions of products. [4] studied how individual and national differences influence the relationship between social media use and customer brand relationship. They tested whether social media interaction is associated with brand perception and relationship quality on brand anthropomorphism. Although they did not employ text mining approach to analyze the social media text, they found that social media was positively related to brand relationship quality. Amado et al. [5] conducted a text mining and topic-based literature analysis to understand research trends on big data in marketing. They scrutinized 1560 articles and used five dimensions that consist of big data, marketing, geography, sector, and products. They concluded that there is a gap for research in finding the pros and cons for organizations to invest in big data and research in big data applications to marketing is still in an early stage. Based on [1], applications of text mining or text analytics are divided into seven groups that are information retrieval, information extraction, concept extraction, natural language processing, clustering, classification, and web mining. Information extraction comprises entity extraction, also popular as Named Entity Recognition (NER), co-reference, and relationship extraction. Ek et al. [6] developed NER from SMS written in Swedish to extract locations, names, dates, times, and telephone numbers using regular expression complemented with classifiers using logistic regression. [7] conducted NER over Electronic Health Records (EHR). EHR is not only used for tracing single patient history, but also for population studies with clinical or administrative purposes. The researcher combined a direct match technique with fuzzy matching and stemmed matching.

There are still limited studies that apply NER on SMS text in business domains. The literature review shows that sentiment analysis is the most popular case in business field while NER is commonly applied to transactional report documents and in health domain. In our country, the established business corporation began to realize the benefits of text mining. However, the utilization of text mining to improve the business is not an easy matter because text mining draws upon the contributions of knowledge input from many external disciplines. Text analytics involves statistics, machine learning, management science, artificial intelligence, computer science, and other disciplines. Since the inherent ambiguity that results from differences in syntax and semantics, such as the use of slang, sarcasm, regional dialects, and technical words or terms specific to industries, text mining can be challenging as well as applying NER to SMS text. This paper reports our research project in utilization large amount of SMS text owned by marketing agency who provides mobile services. The mobile services offer solution integrated with mobile device in the form of SMS sending to grow awareness and engagement between brand and customers. The research objective is to reveal the customer behavior by developing information extraction modules that are applied on SMS text. Specifically, we developed Named Entity Recognition (NER) models which can extract customer and business-related information (called entities) automatically from collections of SMS text. The extracted entities then are analyzed to get insight that can explain the customers' behavior globally from their SMS notifications. The experiment was conducted on three collections of SMS text which are subjective to three business corporation clients called Bank X, Bank Y, and Accommodation Reservation Company Z. Regarding confidentiality, the names of both marketing agency and the business corporations were not published. Our research focused on how to develop robust and accurate NER models, where extracted entities can be further analyzed to get customers' profile and business insight. In this paper, Section 2 explains the research methodology, Section 3 describes the experimental studies and results and Section 4 concludes the study and tells the future works.

## 2. **Research Methodology.**

2.1. **Algorithm in SpaCy.** We explored three NLP packages comprising Natural Language Toolkit (NLTK), TextBlob, and SpaCy. NLTK provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum [8]. TextBlob is NLTK implementation. It provides a simple API for diving into common NLP tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation [9]. SpaCy is an Industrial-Strength Natural Language Processing Tool that is implemented in Python [10]. Furthermore, SpaCy is different from other NLPs as it is created for industrial and not for academic purpose. SpaCy is the main competitor for NLTK. SpaCy has been chosen as the tool because SpaCy is the fastest library [11]. SpaCy applies neural network model architecture. Two key components are parser (parsing models) and NER modules. Parsing models apply dependency parsing scheme using Bidirectional Long-Short Term Memory (BiLSTM) Feature Representations [18]. The critical step in parser design is electing the proper feature function and traditionally, state-of-the-art parsers rely on linear models over hand-crafted feature functions. SpaCy applies BiLSTM which is specific kind of Recurrent Neural Network (RNN) to create feature functions automatically. It is called bidirectional since it is composed of two LSTMs, $\text{LSTM}_F$ and $\text{LSTM}_R$, one reading the sequence in its regular order and the other reading it in reverse. Given $n$-words input sentence $s$ with words $w_1, w_2, \ldots, w_n$ together with the corresponding tags $t_1, t_2, \ldots, t_n$, each word $w_i$ and POS $t_i$ are associated with embedding vectors $e(w_i)$ and $e(t_i)$ and create a sequence of input vectors $(x_{1:n})$ in which each $x_i$ is a concatenation of corresponding word and POS vectors:

$$x_i = e(w_i) \circ e(p_i) \tag{1}$$

The embedding is trained together with the model. This encodes each word in isolation, disregarding its context. The context is introduced by representing each input element as its BiLSTM vector $v_i$:

$$v_i = BILSTM(x_{1:n}, i) \tag{2}$$

The feature function $\Phi$ is then a concatenation of a small number of vectors. The extract feature function is parser dependent. The resulting feature vectors are then scored using a non-linear function, namely Multi-Layer Perceptron with one hidden layer (MLP):

$$MLP_\theta(x) = W^2 \cdot \tanh\left(W^1 \cdot x + b^1\right) + \ b^2 \tag{3}$$

where $\theta = \{W^1, W^2, b^1, b^2\}$ are the model parameters. The basic fundamental of parser is based on transition-based framework, the arc-eager transition system and the imitation learning objective. In the arc-eager system [17], a parser configuration is triple $c = (\Sigma, B, A)$ such that $\Sigma$ (the *stack*) and $B$ (the *buffer*) are disjoint sub lists of the nodes $V_x$ of some sentence $x$ and $A$ is a set of dependency arcs over $V_x$ (and some label set $L$). The initial configuration for a sentence $x = w_1, w_2, \ldots, w_n$ to be $c_s(x) = ([0], [1, \ldots, n], \{\})$; and the terminal configuration of the form $c = (\Sigma, [], A)$ for any stack $\Sigma$ and arc set $A$.

Figure 1 shows there are four types of *transitions* in the arc-eager transition system [17].

1) The *LEFT-ARC*$_l$ transition (for a dependency label $l$) adds the arc $(b, l, s)$ to $A$ where $s$ is the node on the top of stack and $b$ is the first node in the buffer, and pops the

| Transition | | Precondition |
|---|---|---|
| LEFT-ARC$_l$ | $(\sigma\|i, j\|\beta, A) \Rightarrow (\sigma, j\|\beta, A \cup \{(j, l, i)\})$ | $\neg[i = 0] \wedge \neg\exists k \exists l'[(k, l', i) \in A]$ |
| RIGHT-ARC$_l$ | $(\sigma\|i, j\|\beta, A) \Rightarrow (\sigma\|i\|j, \beta, A \cup \{(i, l, j)\})$ | |
| REDUCE | $(\sigma\|i, \beta, A) \Rightarrow (\sigma, \beta, A)$ | $\exists k \exists l[(k, l, i) \in A]$ |
| SHIFT | $(\sigma, i\|\beta, A) \Rightarrow (\sigma\|i, \beta, A)$ | |

FIGURE 1. Transition for the arc-eager transition system [17]

stack. It has as a precondition that the token $s$ is not the artificial root node 0 and does not already have a head.

2) A *RIGHT-ARC*$_l$ transition (for any dependency label $l$) adds the arc $(s, l, b)$ to $A$, where $s$ is the node on the top of the stack and $b$ is the first node in the buffer, and pushes the node $b$ onto the stack.
3) The *REDUCE* transition pops the stack and is subject to the preconditions that the top token has a head.
4) The *SHIFT* transition removes the first node in the buffer and pushes it onto the stack.

NER modules in SpaCy apply transition based NER with Stacked Long-Short Term Memory Networks (LSTMs) model. In the transition-based framework, the input sentence will be stored in a buffer which will then be evaluated. The word being evaluated will go into stack and predicted to be included in what entity of the word is. This stack contains not only one word but also can contain several words that may be included in one entity. Figure 2 gives example of transition sequence for a sentence with Stack-LSTM model. First, it starts with an empty stack, all words of the sentence are in the buffer, and there are no predictable entities. Then, it defines actions that change the state. Last, it predicts the sequence of actions. The LSTM model itself is a model that utilizes the stack to predict the order of words in the stack. The LSTM model is trained using back propagation techniques in general. Long short term memory refers to the ability of the model to"read" (pushing) and "forget" (popping) [13]. The LSTM model can capture three phases from the stack: 1) word of the buffer that will be inserted into the stack, 2) history of actions taken, and 3) all words in the stack. Here the three stages above are three stages that are often defined as (input, output, forget) [14].

| Transition | Output | Stack | Buffer | Segment |
|---|---|---|---|---|
| | [] | [] | [Mark, Watney, visited, Mars] | |
| SHIFT | [] | [Mark] | [Watney, visited, Mars] | |
| SHIFT | [] | [Mark, Watney] | [visited, Mars] | |
| REDUCE(PER) | [(Mark Watney)-PER] | [] | [visited, Mars] | (Mark Watney)-PER |
| OUT | [(Mark Watney)-PER, visited] | [] | [Mars] | |
| SHIFT | [(Mark Watney)-PER, visited] | [Mars] | [] | |
| REDUCE(LOC) | [(Mark Watney)-PER, visited, (Mars)-LOC] | [] | [] | (Mars)-LOC |

FIGURE 2. Transition sequence for *Mark Watney visited Mars* with Stack-LSTM model [12]

2.2. **The research steps.** Figure 3 explains our research steps, starting from gathering the business SMS text to revealing business insights or recommendation. First step is data gathering. Collections of business SMS text used in this research are provided by the marketing agency who offers mobile services to business corporations as part of their means of communication to their customers. We conducted experiments with two types of companies, Bank and Accommodation Reservation Company. Regarding confidentiality, the name of corporations was called Bank X, Bank Y, and Accommodation Reservation Company Z. Also, SMS text examples cannot be shown in this paper. The data gathering

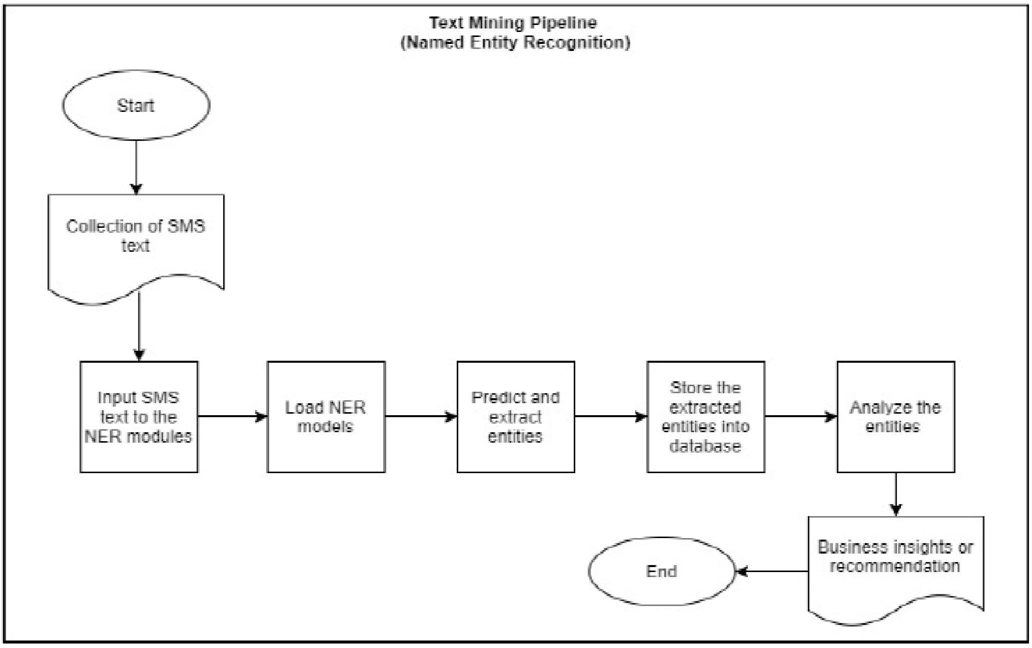


FIGURE 3. Text mining pipeline (named entity recognition)

process gathered 200,036 SMS texts of Bank X, 31,072 SMS texts of Bank Y and 530,121 SMS texts of Accommodation Reservation Company Z. Next step is defining entities. Before defining the entities, the initial process is observing the SMS text format, message content, composition, and order of the three sources. It is to understand message text pattern and to decide appropriate entities. We defined mobile phone number, transaction type, account number, nominal, date, time, and transaction note entities for Bank X. For Bank Y, the entities are mobile phone number, transaction type, account number, nominal, date, time, origin account number, and destination account number entities. Lastly, for Accommodation Reservation Company Z, the entities include mobile phone number, message type, card expired, booking code, name of hotel, hotel domicile, hotel address, name of guest, length of stay, stay duration, type of booked room, number of booked room, nominal, route flight, number of passenger, message source, and date.

We conducted supervised learning to construct NER model. Due to this, we had to build the training set (called as corpus). We performed manually taggingbu human annotator to get the best and accurate corpus. We used General Architecture for Text Engineering (GATE) to help the tagging process. GATE is an infrastructure for developing and deploying software components that process human language. It is nearly 15 years old and is in active use for all types of computational task involving human language. GATE excels at text analysis of all shapes and sizes and GATE has an IDE. GATE Developer is an integrated development environment for language processing components bundled with a very widely use information extraction system and a comprehensive set of other plugins [16]. One of features that we used from GATE Developer is annotation tools to annotate our SMS text. 5000 SMS texts from each corporation were annotated. Annotation or tagging is the process of giving a name to a word/a phrase/a token in the text. For example:

SMS text: During the ISMI-ICTAS 2018 conference, Siti Mariyah stayed at hotel A with a deluxe room for 2 nights on 4 to 6 September 2018. The annotation result for the text: During the ISMI-ICTAS 2018 conference, *Siti Mariyah <<name of guest>>* stayed at *hotel A <<name of hotel>>* with a *deluxe <<type of booked room>>* room for *2 nights <<length of stay>>* on *4 to 6 September 2018 <<stay duration>>.*

Then, the annotated SMS texts were validated by an expert to minimalize or remove the annotation errors. The validated annotated texts were transformed into SpaCy-needed input format. We changed the structure data from xml into array format. We called this format as corpus so that we have three different corpora. These corpora are used to develop NER model by applying supervised learning method. By using SpaCy, we trained each corpus to create NER model. The iterations are set to 10 and samples are set to 1000. It means that for each iteration, SpaCy takes 1000 random samples of annotated SMS text to observe the entities positions and compositions. We used custom tagging API from SpaCy with `.add_label` function to add custom label to the stream. It needs about five to seven hours to build one NER model using a computer with 4 GB RAM. The sign that the modeling process is finished is constructed two folders that are NER and vocabulary folders and two files that are meta and tokenizer files. After the NER models obtained, the next step is model validation. The validation aims to measure the accuracy and performance of NER models in recognizing and predicting entities written in SMS text. The validation was conducted by the prediction test of new SMS text and calculation of accuracy. The new SMS text means that the SMS text is never used for training the model and text does not have tag or label. We built NER modules that automatically call the NER models to predict and extract all defined entities from new SMS text in streaming fashion. The extracted entities are directly stored indefined tables in SMS database. There are four tables in which each table stores the extraction results of each NER model and one table is dedicated to store all SMS text that does not have any defined entities at all. It was done since the observation result told that system possibly has many promo SMS. In this research, exploratory data analysis was used to explore and understand the extracted entities stored in the database. We searched for what type of business transaction was mostly done by customers, what financial products are the most in demand, and how many times customers use the same business products in a given time range.

3. **Results and Discussion.** In this research, the text mining pipeline has been successfully built. The text mining pipeline focuses on NER case on large collection of SMS business. Therefore, the user or staff of marketing agency inputs the SMS text only in streaming fashion and then can directly get the defined entities in a structured manner (in the form of database). According to the testing results, the NER models averagely can recognize and extract entities from four SMS texts in one second. We reported the performance of NER models on testing sets. Our testing sets contain 26,072 SMS texts of Bank X, 95,036 SMS texts of Bank Y and 525,121 SMS texts of Accommodation Reservation Company Z.

TABLE 1. The performance of NER model on collection of SMS text of Bank X

| No | Entities | True Prediction | False Prediction | Null | Filled | Accuracy | Error |
|---|---|---|---|---|---|---|---|
| 1 | Mobile phone number | 26072 | 0 | 0 | 26072 | 100% | 0% |
| 2 | Account number | 25981 | 0 | 91 | 25981 | 100% | 0% |
| 3 | Nominal | 26072 | 0 | 0 | 26072 | 100% | 0% |
| 4 | Date | 26072 | 0 | 0 | 26072 | 100% | 0% |
| 5 | Time | 25981 | 2 | 89 | 25983 | 99.99% | 0.01% |
| 6 | Transaction type | 25981 | 5 | 86 | 25986 | 99.98% | 0.02% |
| 7 | Transaction notes | 127 | 3 | 25942 | 130 | 97.69% | 2.31% |
| Average: | | | | | | 99.66% | 0.34% |

a) Bank X

The NER models can extract seven entities from each SMS text comprising mobile phone number, account number, nominal, date, time, transaction type, and transaction notes. We got more than 99% of accuracy for all entities except transaction notes. Transaction notes are rare entities, and there are only 130 occurrences (the percentage is about 0.49%) in testing set. The number of characters of transaction notes entity is longer than other entities. The words or phrases used by this entity are more diverse and the semantic is too. It causes the accuracy is lower than the others.

b) Bank Y

Table 2 shows that we could extract important information from SMS text of Bank Y such as mobile phone number, account number, nominal, date, time, transaction type, origin account number and destination account number. The performance of NER models decreases at 91.19% on recognizing destination account number. This is because the destination account number entity is written in abbreviated format with no fix length and structure.

Table 2. The performance of NER model on collection of SMS text of Bank Y

| No | Entities | True Prediction | False Prediction | Null | Filled | Accuracy | Error |
|---|---|---|---|---|---|---|---|
| 1 | Mobile phone number | 95036 | 0 | 0 | 95036 | 100% | 0% |
| 2 | Account number | 95036 | 0 | 0 | 95036 | 100% | 0% |
| 3 | Nominal | 94925 | 111 | 0 | 95036 | 99.88% | 0.12% |
| 4 | Date | 95036 | 0 | 0 | 95036 | 100% | 0% |
| 5 | Time | 95036 | 0 | 0 | 95036 | 100% | 0% |
| 6 | Transaction type | 86555 | 0 | 8481 | 86555 | 100% | 0% |
| 7 | Origin account number | 95036 | 0 | 0 | 95036 | 100% | 0% |
| 8 | Destination account number | 86666 | 8370 | 0 | 95036 | 91.19% | 8.81% |
| Average: | | | | | | 98.88% | 1.12% |

c) Accommodation Reservation Company Z

SMS text of Accommodation Reservation Company Z provides more rich information about business transaction compared to both Bank X and Bank Y. The NER model is able to recognize and extract 16 entities which consist of mobile phone number, message type, card expired, booking code, name of hotel, hotel address, name of guest, length of stay, stay duration, type of booked room, number of booked room, nominal, route flight, number of passenger, date, and departure time. Five entities comprising mobile phone number, message type, booking code, number of booked room, and number of passenger can be perfectly extracted while other entities cannot.

The two worst accuracies are in type of booked room and card expired entities by 74.22% and 75.15% respectively. We found that types of hotel room are written in many terms, which can be named based on the number of beds, facilities, or combination of these two components. On the other hand, we can say that vocabulary for types of hotel room is larger than other entities. Another cause is that different hotels have different formats of SMS text for customer's notification after a customer books the room. Types of booked rooms are written in any position, which can be in the start, in the middle or in the end of sentence. The card expired entity has bad accuracy of NER due to bad definition of this entity itself and the lack of training set. It means that the card expired entities will

Table 3. The performance of NER model on collection of SMS text of Accommodation Reservation Company Z

| No | Entities | True Prediction | False Prediction | Null | Filled | Accuracy | Error |
|---|---|---|---|---|---|---|---|
| 1 | Mobile phone number | 525121 | 0 | 0 | 525121 | 100% | 0% |
| 2 | Message type | 482453 | 0 | 42668 | 482453 | 100% | 0% |
| 3 | Card expired | 12111 | 4005 | 509005 | 16116 | 75.15% | 24.85% |
| 4 | Booking code | 324636 | 14 | 200471 | 324650 | 100% | 0% |
| 5 | Name of hotel | 115955 | 5233 | 403933 | 121188 | 95.68% | 4.32% |
| 6 | Hotel address | 36248 | 92 | 488781 | 36340 | 99.75% | 0.25% |
| 7 | Name of guest | 171463 | 1932 | 351726 | 173395 | 98.89% | 1.11% |
| 8 | Length of stay | 101520 | 10 | 423591 | 101530 | 99.99% | 0.01% |
| 9 | Stay duration | 11588 | 174 | 513359 | 11762 | 98.52% | 1.48% |
| 10 | Type of booked room | 114428 | 39742 | 370951 | 154170 | 74.22% | 25.78% |
| 11 | Number of booked room | 106342 | 2 | 418777 | 106344 | 100% | 0% |
| 12 | Nominal | 51004 | 133 | 473984 | 51137 | 99.74% | 0.26% |
| 13 | Route flight | 75002 | 1173 | 448946 | 76175 | 98.46% | 1.54% |
| 14 | Number of passenger | 339061 | 0 | 185520 | 339601 | 100% | 0% |
| 15 | Date | 430836 | 3734 | 90551 | 434570 | 99.14% | 0.86% |
| 16 | Departure time | 346020 | 984 | 178117 | 347004 | 99.72% | 0.28% |
| Average: | | | | | | 96.20% | 3.8% |

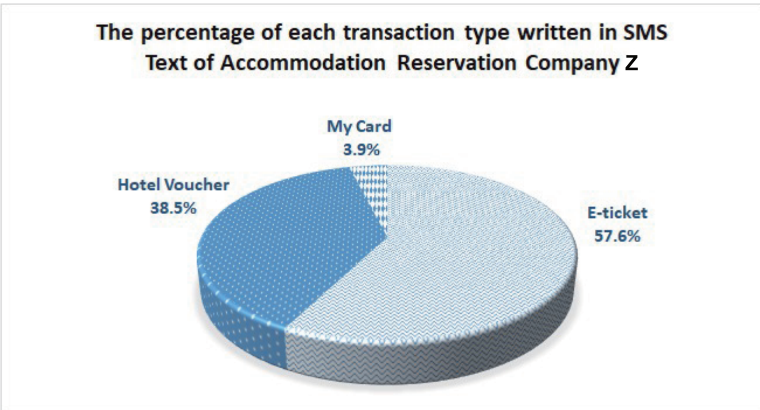


Figure 4. Types of business transaction written in SMS text of Accommodation Reservation Company Z

appear only for customers who have card and use the card for reservation. Data analysis was performed using extracted entities to get customer behavior aggregately. Figures 4-6 direct to understand the business transaction kind which is written in SMS text.

Figure 4 shows that flight booking (e-ticket) more frequently happened than hotel reservation and card services. Figure 5 shows only two kinds of business transactions informed via SMS text and the frequency of both transactions is almost equal. Another analysis is linking the user with the same mobile phone number to get all transactions he or she did. Figure 6 tells that there are various business transactions informed by SMS text and debit transaction dominates among others. Debit transaction, transfer ATM, and withdrawal ATM are three frequent transactions done by clients of Bank X. We conclude

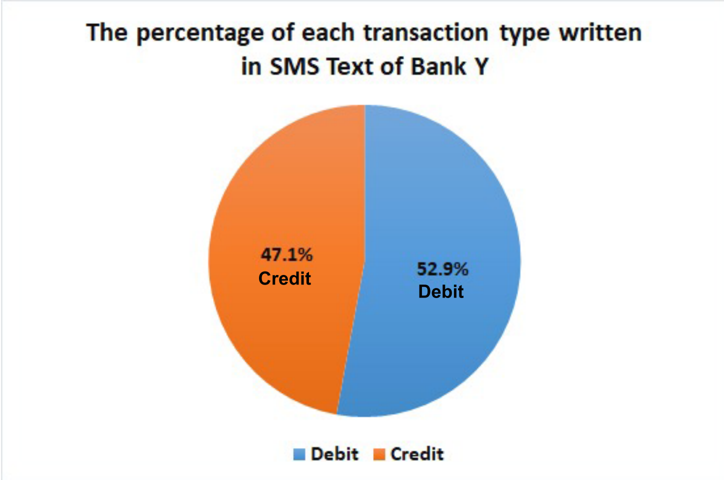


FIGURE 5. Types of business transaction written in SMS text of Bank Y

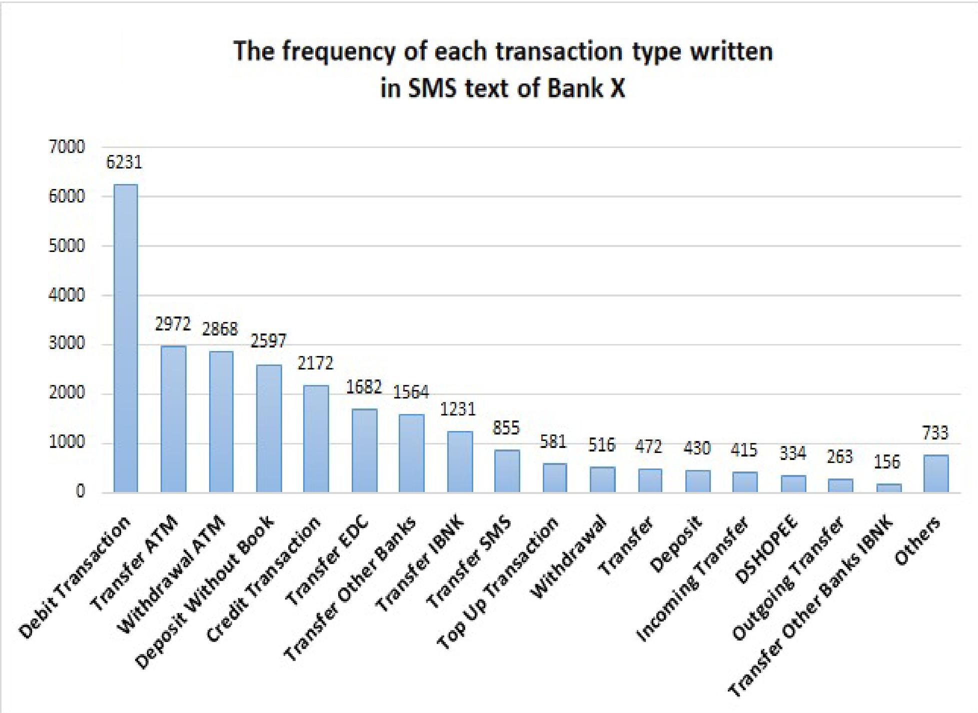


FIGURE 6. Types of business transaction written in SMS text of Bank X

that ATM use dominates banking transactions since the simplicity and the quickness services delivered by ATM machines. Then apply classification analysis to those users whether the users are on the high-level economy or not based on the sum of transaction's nominal. All analyses depend on the business's goals defined by the chief executive officer of the corporation/company.

4. **Conclusion and Future Works.** This research has built a text mining pipeline in the case of named entity recognition on a collection of SMS text owned by marketing agency. The process of recognition and extraction of valuable transactional data has been automated and are called as entities written in SMS text. The NER models that were constructed can do those tasks very well, with most accuracy more than 90%. We developed a database which stores the extracted entities to be used in the analysis stage. We also gave one example of how to utilize this large collection of entities to understand

the user behavior in consuming the business products. This research is performed using standard personal computer which has limited computational resources. Therefore, it needs about 5-7 hours to construct the NER models and about one second to recognize and extract entities written in SMS text. In the real case, the marketing agency usually sends about 5 million SMS text per month for one company or corporate client. If using the standard computer to process all SMS texts, it needs approximately 347.22 hours or 14.5 days. So, for the next research, it can utilize big data tools or softwares such as Hadoop and Spark to process the large collection SMS text.

## REFERENCES

[1] B. Mohit, Named entity recognition, *Natural Language Processing of Semitic Languages*, pp.221-245, 2014.

[2] M. M. Mostafa, More than words: Social networks' text mining for consumer brand sentiments, *Expert Systems with Applications*, vol.40, no.10, pp.4241-4251, 2013.

[3] O. Netzer, R. Feldman, J. Goldenberg and M. Fresko, Mine your own business: Market-structure surveillance through text mining, *Marketing Science*, vol.31, no.3, pp.521-543, 2012.

[4] S. Hudson, L. Huang, M. S. Roth and T. J. Madden, The influence of social media interactions on consumer-brand relationships: A three-country study of brand perceptions and marketing behaviors, *International Journal of Research in Marketing*, vol.33, no.1, pp.27-41, 2016.

[5] A. Amado, P. Cortez, P. Rita and S. Moro, Research trends on big data in marketing: A text mining and modeling based literature analysis, *European Research on Management and Business Economics*, vol.24, no.1, pp.1-7, 2018.

[6] T. Ek, C. Kirkegaard, H. Jonsson and P. Nugues, Named entity recognition for short text messages, *Procedia Social and Behavioral Sciences*, 2011.

[7] A. P. Quimbaya, A. S. Munera, R. A. G. Rivera, J. C. D. Rodriguez, O. M. M. Velandia, J. A. G. Pena and C. Labbe, Named entity recognition over electronic health records through a combined dictionary-based approach, *Procedia Computer Science*, 2016.

[8] *Natural Language Toolkit – NLTK 3.3 Documentation*, http://www.nltk.org/, Accessed on 13 August 2018.

[9] *TextBlob: Simplified Text Processing – TextBlob 0.15.1 Documentation*, http://textblob.readthedocs.io/en/dev/, Accessed on 13 August 2018.

[10] spaCy, *Industrial-Strength Natural Language Processing in Python*, http://www.spacy.io, Accessed on 13 August 2018.

[11] J. D. Choi, J. Tetreault and A. Stent, It depends: Dependency parser comparison, *Proc. of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, vol.1, pp.387-396, 2015.

[12] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami and C. Dyer, Neural architectures for named entity recognition, *Proc. of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp.260-270, 2016.

[13] C. Dyer, M. Ballesteros, W. Ling, A. Matthews and N. A. Smith, Transition-based dependency parsing with stack long short-term memory, *arXiv preprint arXiv:1505.08075*, 2015.

[14] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink and J. Schmidhuber, LSTM: A search space odyssey, *IEEE Transactions on Neural Networks and Learning Systems*, vol.28, no.10, pp.2222-2232, 2017.

[15] M. Honnibal and M. Johnson, An improved non-monotonic transition system for dependency parsing, *Proc. of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015.

[16] *GATE.ac.uk – sale/tao/splitch1.html*, https://gate.ac.uk/sale/tao/splitch1.html, Accessed on 13 August 2018.

[17] G. Yoav and J. Nivre, A dynamic oracle for arc-eager dependency parsing, *International Conference on Computational Linguistics*, vol.527, pp.236-254, 2015.

[18] E. Kiperwasser and Y. Goldberg, Simple and accurate dependency parsing using bidirectional LSTM feature representations, *Trans. Assoc. Comput. Linguist.*, vol.4, pp.313-327, 2016.