IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Mathematical Modelling of Engineering Problems
Vol. 11, No. 9, September, 2024, pp. 2566-2572
Journal homepage: http://iieta.org/journals/mmep

Enhancing the Diversity of Smart Reply Suggestions: A Novel Approach Combining Text Classification and Post-Processing Techniques for Real Conversations in Bahasa Indonesia

Mohamad Adhikasurya Haidar[1], Syafri Bahar[2], Nanang Susyanto[1*]

[1] Department of Mathematics, Universitas Gadjah Mada, Yogyakarta 55281, Indonesia
[2] Gojek Indonesia, Jakarta 12160, Indonesia

Corresponding Author Email: nanang_susyanto@ugm.ac.id

https://doi.org/10.18280/mmep.110927

Received: 5 December 2023
Revised: 24 April 2024
Accepted: 30 July 2024
Available online: 29 September 2024

Keywords:
smart reply, natural language processing, text clustering, postprocessing methods, reply suggestions

ABSTRACT

Smart Reply is a Natural Language Processing application that offers suggestions for replies, enabling users to respond to messages quickly without having to type them out. However, the application sometimes generates reply suggestions that have similar meaning. In this research, we employ text clustering methods to address this problem. Furthermore, the application often provides specific information, such as phone numbers and addresses, which can be generated from the training data. To address this problem, we implement post-processing methods, including removing phone numbers or addresses from reply candidates and utilizing the most frequently used replies to introduce greater variety. Our results demonstrate the effectiveness of our approach in diversifying reply suggestions.

1. INTRODUCTION

Smart Reply has become a widely used feature in many messaging applications, offering users suggested responses to incoming messages. It also has garnered significant attention in recent research conducted by Shay et al. [1]. This functionality facilitates quick and easy communication, saving time and effort for users. For example, Smart Reply in email applications provides convenient, contextually appropriate responses with just a single click [2]. The core of the system lies in using text classification to generate suitable responses [3]. Despite its popularity, there is room for improvement in the quality of these suggestions. Often, the responses provided by Smart Reply systems are too similar in meaning, which can make them feel repetitive and limit their usefulness. To tackle this issue, researchers have proposed a variety of text classification techniques aimed at diversifying the range of Smart Reply results [4].

One approach involves the use of character-level convolutional networks, as explored by Zhang et al. [5]. This technique analyzes the text at the character level, which can help in capturing subtle nuances and variations in language that might be missed at the word or phrase level. Another method, multitask learning, as investigated by Peng et al. [6], involves training a model on multiple related tasks simultaneously, which can improve its ability to generalize and produce diverse responses. Additionally, hierarchical attention networks, studied by Yang et al. [7], use multiple layers of attention mechanisms to focus on different parts of the input text, allowing the model to generate more contextually relevant responses.

Recent advancements in natural language processing (NLP) have also seen the use of pre-trained language models, such as BERT (Bidirectional Encoder Representations from Transformers) by Devlin et al. [8] and the Universal Sentence Encoder by Cer et al. [9], for text classification tasks. These models have been trained on large corpora of text and have demonstrated superior performance across a variety of NLP tasks, including text classification. Their ability to understand and generate text makes them powerful tools for improving Smart Reply systems by providing more nuanced and varied response suggestions. Moreover, researchers have explored training generative language models on extensive datasets to improve the relevance of these suggestions [10]. From a sentiment perspective, researchers are also working to generate responses that are contextually relevant and emotionally appropriate [11].

In addition to using pre-trained models, researchers have explored the concept of transfer learning for Smart Reply. Transfer learning involves first training a model on a large, general dataset and then fine-tuning it on a smaller, specific target dataset, as discussed by Howard and Ruder [12]. This approach is particularly useful when dealing with limited training data, as it allows the model to leverage knowledge acquired from the larger dataset and apply it to the specific task at hand. This method has shown promising results in improving the quality and diversity of text classification tasks, including Smart Reply.

Other advanced techniques proposed for enhancing Smart Reply systems include deep neural networks, as detailed by LeCun et al. [13], and the use of word embeddings, such as those developed by Mikolov [14] and Pennington et al. [15]. Word embeddings map words to high-dimensional vectors that capture semantic meanings, which can be used to find similarities and differences between words. Contextualized word vectors, introduced by McCann et al. [16], further refine this approach by considering the context in which words appear, providing a deeper understanding of language and improving the relevance of generated responses.

Despite these advancements, a significant challenge remains: the tendency of Smart Reply systems to generate responses with similar meanings, even when employing diverse text classification techniques. Character-level convolutional networks, multitask learning, and hierarchical attention networks, while valuable, do not entirely solve the problem of generating genuinely diverse and contextually appropriate responses. This can lead to the system offering suggestions that are repetitive or not particularly useful in different conversational contexts. Additionally, Smart Reply systems might suggest specific information such as phone numbers or addresses, which may be inappropriate or sensitive in certain situations.

To address these issues, researchers have proposed various post-processing techniques. For instance, methods like removing personally identifiable information (such as phone numbers and addresses) from the suggested responses can help prevent inappropriate suggestions. Another approach involves incorporating time-based greetings or context-specific language to make the suggestions more relevant and varied, as suggested by Bhatia et al. [17]. However, these post-processing methods have not been extensively explored in the academic literature, and their effectiveness in enhancing the diversity and appropriateness of Smart Reply suggestions has not been thoroughly evaluated.

In conclusion, while significant progress has been made in improving the diversity and relevance of Smart Reply systems, challenges remain. Ongoing research is needed to refine these techniques and develop new methods for generating responses that are not only diverse but also contextually appropriate and sensitive to the nuances of human communication.

While a number of studies and references have delved into the diversity of smart replies in the English language, there appears to be a notable gap in the literature when it comes to research specifically targeting the diversity of smart replies in Bahasa Indonesia. To the best of our knowledge, no existing publication has thoroughly explored this particular aspect. This gap is especially significant given that Bahasa Indonesia possesses unique linguistic characteristics that differentiate it markedly from English. For instance, Bahasa Indonesia has a different grammatical structure, vocabulary usage, and cultural context, all of which can impact the way smart replies are generated and perceived. Recognizing these differences is crucial for developing more accurate and culturally sensitive smart reply systems. In this paper, we set out to propose a novel methodology aimed at enhancing the diversity of Smart Reply outcomes by employing sophisticated text classification techniques. Our approach seeks to address the specific challenges and nuances associated with generating diverse replies in Bahasa Indonesia.

To achieve this, we leverage the term frequency and inverse document frequency method, or TF-IDF, which is a well-established technique in the field of natural language processing [18]. TF-IDF is used to convert reply suggestions into numerical representations, which are essential for computational analysis. This method helps us to quantify the importance of words within a given dataset of responses, based on their frequency and distribution across different documents. By doing so, we can better understand which words are most significant in shaping the meaning of the replies. Following the TF-IDF transformation, we implement K-means clustering, a popular unsupervised machine learning algorithm, to categorize the replies into groups based on their semantic similarity [19]. The goal of clustering is to organize the replies in a way that groups similar meanings together, thereby making it easier to analyze and generate diverse replies. To evaluate the quality of these clusters, we utilize the Silhouette score, a metric that measures how similar an object is to its own cluster compared to other clusters. A higher Silhouette score indicates better-defined and more distinct clusters.

The vectorization of words using TF-IDF is performed in the usual manner, converting text into a matrix of numerical values that represent the significance of each word. This allows for more precise manipulation and analysis of the text data. Meanwhile, the K-means clustering algorithm groups the replies into clusters, which helps in organizing the responses by their underlying meanings. This clustering process is crucial for identifying the diversity of replies, as it helps to ensure that similar responses are not redundantly suggested. After the initial clustering, we apply several post-processing techniques to further enhance the diversity of the suggestions. These techniques include the removal of sensitive or personally identifiable information, such as phone numbers or addresses, which could inadvertently be included in the responses. Additionally, we incorporate the most frequently used replies into the system to introduce a greater variety of responses. This step is essential for ensuring that the smart reply system can handle a wide range of conversational scenarios and provide responses that are both relevant and varied.

Our results demonstrate that the proposed methodology is effective in diversifying Smart Reply results, particularly for real-world conversations conducted in Bahasa Indonesia between drivers and customers. The approach not only enhances the diversity of the replies but also improves the overall relevance and appropriateness of the suggestions. This research contributes to the broader field of natural language processing by providing insights into how to adapt smart reply systems to different languages and cultural contexts, thereby making them more inclusive and effective.

In general, our proposed methodology builds on previous research in text classification and post-processing techniques for Smart Reply. However, our approach is novel in its combination of these techniques and in its evaluation of clustering quality using the Silhouette score. Furthermore, our post-processing techniques extend beyond mere removal of phone numbers and addresses; they include clustering to diversify Smart Reply results. The clustering ensures that responses with identical meanings are only considered once.

The remainder of this paper is organized as follows. In Section 2, we describe our methodology to diversify Smart Reply results. In Section 3, we present the data set and report and analyze our experimental results. Finally, in Section 4, we provide conclusions and directions for future work.

2. TWO-STEP METHOD

Our proposed method for enhancing Smart Reply systems involves a two-step process: the initial smart reply generation followed by a post-processing phase. In the first step, known as the regular smart reply phase, the system generates a set of candidate responses based on the input message. This step utilizes advanced text generation and classification techniques to produce a list of potential replies. However, it's important to note that these candidates can sometimes include responses with similar meanings or redundant information. This is a common issue in smart reply systems, where the generated replies may not always offer the variety needed to handle different conversational contexts effectively.

To address this issue and ensure that the suggested responses are diverse and contextually relevant, we introduce a second step called the post-processing phase. In this phase, the generated candidate replies undergo further refinement to diversify the options presented to the user. The goal of this post-processing step is to filter out redundant or overly similar responses and enhance the variety of the top-ranked suggestions. This is achieved by employing various techniques such as semantic analysis, clustering, and removal of repetitive elements. For instance, the system might analyze the semantic content of each reply and prioritize those that offer distinct perspectives or additional useful information. By doing so, the post-processing step ensures that the final suggestions presented to the user are not only diverse but also meaningful and relevant to the conversation.

Ultimately, this two-step method aims to improve the overall user experience by providing a wider range of high-quality reply options. The regular smart reply phase quickly generates a broad set of potential responses, while the post-processing phase fine-tunes these responses to offer the most diverse and appropriate suggestions. This approach helps to address common limitations in current smart reply systems, such as the tendency to produce repetitive or irrelevant responses, thereby making the technology more useful and engaging for users.

2.1 Smart reply step

The system utilizes a learning framework of sequence-to-sequence, utilizing Long Short-Term Memory networks (LSTMs) to forecast text sequences. Input sequences comprise incoming messages, while the output distribution encompasses potential replies. The primary goal of the Smart Reply system is to determine the most relevant response to messages given to the system. Essentially, when a message ***o*** is given and a collection of all potential replies *R* is set, our objective is to find:

$$r^* = \underset{r \in R}{argmax}\; P(\boldsymbol{r}|\boldsymbol{o}) \qquad (1)$$

To achieve this, we will build a model that evaluates responses and selects the one with the highest score.

This problem is easily adaptable to sequence-to- sequence learning because we score the ***r*** series based on another ***o*** series of tokens. This model uses LSTM. The inputs are tokens of the original message, marked as $o_1, \dots, o_n$, and the output shows the conditional probability distribution of the response token sequence given to the input, i.e.,

$$P(r_1, \dots, r_m|o_1, \dots, o_n) \qquad (2)$$

This distribution can be expressed as a product of *m* conditional probabilities:

$$P(r_1, \dots, r_m|o_1, \dots, o_n) = \prod_{i=1}^{m} P(r_i|o_1, \dots, o_n, r_1, \dots, r_{i-1}) \qquad (3)$$

Initially, the model processes the original message token sequence and incorporated a special message end token o_n. The hidden state of the LSTM encodes the vector representation of the whole message. Subsequently, utilizing this hidden state, the softmax output is calculated, representing $P(r_1|o_1, \dots, o_n)$, which represents the probability distribution of the first response token. When the response tokens are inputted sequentially, at each time step *t*, the softmax is interpreted as $P(r_t|o_1, \dots, o_n, r_1, \dots, r_{t-1})$. With this decomposition, those softmaxes are utilized to determine $P(r_1, \dots, r_m|o_1, \dots, o_n)$.

When provided with a large number of messages, the objective aims to maximize the likelihood of a recorded response provided their corresponding original messages:

$$\sum_{(\boldsymbol{o},\boldsymbol{r})} logP(r_1, \dots, r_m|o_1, \dots, o_n) \qquad (4)$$

The model is trained against this objective using stochastic gradient descent.

In the inference phase, we input an original message and use the Softmax output to obtain a probability distribution across vocabulary in each step. This may be done in various options: (1) to generate random samples from $P(r_1, \dots, r_m|o_1, \dots, o_n)$, one approach is to sample one token in each step of time and return it to the model or (2) to estimate the most probable response to the original message, a greedy strategy entails choosing the most likely tokens at each step of time and returning them into the model. Alternatively, a less greedy approach like beam search involves selecting the top *m* tokens, inputting them into the model, retaining the best response prefix *m*, and iterating the process.

After generating candidate responses, we proceed to the post-processing step, which involves removing uninformative words (or characters) and clustering to further diversify the candidate responses. We use the beam score value from the Smart Reply model to select a response with the highest beam score value from each cluster, thereby determining our final responses.

2.2 Post processing

2.2.1 Removing and representing replies

The conversations between drivers and customers often contain specific information, such as phone numbers and addresses, which are not informative, so we need to remove them. Additionally, frequent replies, such as various expressions of gratitude, are common, so we select one representative reply.

1) Removal of the phone or address number: In this post-processing step, we detect Indonesia's phone number code (+62 or 08) within the responses. Once detected, the corresponding reply is removed. The process of removing specific addresses follows a similar approach to removing phone number replies. Figure 1 shows an example of how phone and address numbers are removed.
2) Taking one representative word(s): In our current smart reply application, frequent responses, such as expressions of gratitude, are common. We perform a basic categorization by selecting one representative word(s) from the variations. Figure 2 shows that the word "Thanks" is taken to represent "Thanks a lot", "Thanks", and "Thank You".

2.2.2 Clustering

Before clustering the new candidate responses, it is necessary to convert the responses into numeric value representations. Otherwise, the clustering algorithm will not be able to process the input. In this scenario, the TF-IDF word representation is employed. TF-IDF, which stands for term frequency-inverse document frequency, is a metric that can assess the importance or relevance of string representations (words, phrases, lemmas, etc.) in a document within a collection of documents (also referred to as a corpus). TF-IDF comprises two components: term frequency and inverse document frequency.

TF operates by assessing the frequency of a certain terms concerning the document. There are several measures or methods to define frequency:

- Raw count (rc): The frequency of the word in a document,
- TF adjusts the length of the document: the raw number of events is divided by the number of words in the document,
- Log-scaled frequency: For example, log(1+rc),
- Boolean frequency: 1 is the term appearing in the document, 0 is the term not appearing in the document.

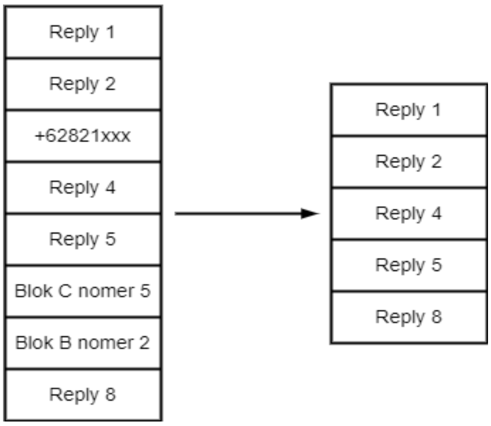


Figure 1. Removing phone or address number

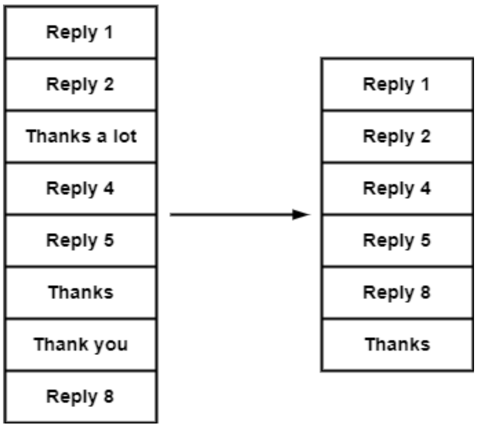


Figure 2. Most frequently used replies

Inverse Document Frequency (IDF) assesses how common or rare words are in the corpus. The IDF is calculated in the following way: t represents the term (word) evaluated for its commonness, and N represents the total number of documents (d) in D. The denominator is simply the number of documents where the word t appears in:

$$\log\left(\frac{N}{count(d \in D : t \in d)}\right) \tag{5}$$

Occasionally, a does not appear in the corpus, potentially leading to an error due to the division by zero. An approach for solving this problem is by increasing the current number one, effectively creating a denominator of (1+).

The IDF serves to adjust for common words such as "of," "as," "the," etc., which are prevalent in a corpus of English. Consequently, through the IDF, the influence of common terms can be reduced, allowing fewer common terms to have a greater impact on:

$$tfidf(t, D) = tf(t, d) \times \log\left(\frac{N}{count(d \in D : t \in d)}\right) \tag{6}$$

After the data have been converted into numerical value representations, it is time to cluster the new candidate replies. The k-Means is one of the most popular partitioning clustering algorithms.

A dataset Y comprising entities $y_i \in Y$ with dimension V, where, $i \in \{1,2,\dots,N\}$, this algorithm produces m non-empty disjoint sets $S_1, S_2, \dots, S_m$ with $S_1, \cup S_2 \cup \cdots \cup S_m = S$ centered around central points $C = \{c_1, c_2, \dots, c_m\}$ through iterative minimization of the total distance within clusters between entity and a central point:

$$W_k = W(S, C) = \sum_{k=1}^{m} \sum_{i \in S_k} d(y_i, c_k). \tag{7}$$

Every central point c_k uniquely characterizes a cluster S_k and is occasionally referred to as its candidate. The above k-Means criterion yields an index that signifies the quality of clustering, where lower values denote better clustering. The distance measure $d(y_i, c_k)$, in the above equation, typically uses the squared Euclidean distance:

$$d(y_i, c_k) = \sum_{v \in V} (y_{iv} - c_{kv})^2 \tag{8}$$

minimizing the square error criterion.

The minimization process comprises three straightforward steps, repeated to the convergence:

1) Choose the value of the m entity $y_i \in Y$ as the first central point $c_1, c_2, \dots, c_m$.
2) Associate every $y_i \in Y$ with the cluster S_k, called c_k, which is the nearest central point to y_i.
3) Stop and produce S and C if S remains unchanged. If not, go back to Step 2 and update each center point c_k to the cluster S_k.

Because of its greedy nature, K-Means cannot guarantee convergence to the global minima, and its last clustering outcome heavily relies on the first central points. To estimate the number of clusters, there are several cluster validity indexes (CVIs) available, including the silhouette index (scoring).

The silhouette score Rousseeuw [20] is a useful metric for determining the optimal number of clusters in K-Means clustering. It relies on silhouette values for each entity y_i, evaluating how well y_i aligns with the cluster it belongs to. The silhouette score contrasts the cohesion of the cluster (determined by the distance from all the entity in the same cluster) and the separation of the cluster:

$$s(y_i) = \frac{b(y_i) - a(y_i)}{\max\{a(y_i), b(y_i)\}}. \quad (9)$$

Here, $a(y_i)$ represents the average dissimilarity of $y_i \in S_k$ to all other $y_j \in S_k$, and $b(y_i)$ denotes the lowest dissimilarity across S_ls that has not assigned to y_i, and calculated as the mean dissimilarities to $y_j \in S_l, l \neq k$. Therefore, $-1 \leq s(y_i) \leq 1$. When $s(y_i)$ is approximately zero, y_i could potentially be positioned in a different cluster without worsening the separation of the cluster. Negative $s(y_i)$ indicates that the grouping of y_i's hinders clustering and separation, while close to one $s(y_i)$ indicates the contrary. The overall validity of the clustering can be quantified through Silhouette scoring, defined as $\frac{1}{N}\sum_{i \in Y} s(y_i)$.

The final step of the method is to select one reply from each cluster group. The reply chosen from each cluster is the one with the highest value of the beam score. For example, if the final candidate replies consist of 4 clusters, there will be 4 unique responses selected as the final candidate replies to be output to the application. Therefore, the choice of a reply from each cluster is not random but is based on its beam score value.

3. RESULTS

In this section, we will put the proposed method into practice to assess its effectiveness in enhancing the diversity of Smart Reply results, specifically for real-world conversations between drivers and customers using Bahasa Indonesia. The dataset utilized for this evaluation is derived from chat applications operated by a prominent technology company based in Indonesia. This dataset includes a wide range of conversational exchanges that reflect the actual interactions between users of the service.

The data processing is handled by an in-house application developed by the company, which employs sophisticated techniques including Term Frequency-Inverse Document Frequency (TF-IDF) and Long Short-Term Memory (LSTM) networks. The TF-IDF technique is utilized to convert text data into numerical representations that capture the significance of words in the context of their usage. This representation is crucial for subsequent analysis and processing. LSTM networks are applied to capture long-range dependencies and contextual information within the text, which helps in understanding and generating more relevant and coherent replies.

To further enhance the diversity of the generated responses, K-means clustering is employed. This technique groups similar replies together based on their semantic content, which allows for the identification and separation of responses with similar meanings. The clustering process helps in filtering out redundant replies and promoting a wider range of response options.

The effectiveness of the proposed method is evaluated by inputting various types of questions and analyzing the resulting replies. The evaluation focuses on several key aspects, particularly the performance in terms of diversification. We aim to compare the traditional "regular" smart reply approach with our proposed method to highlight improvements in response variety.

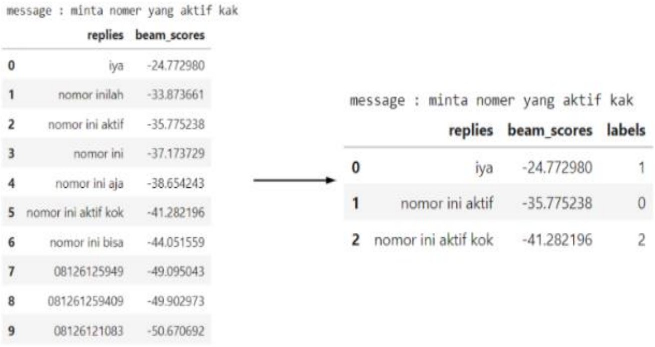
We present three specific cases to illustrate the effectiveness of our method:

1) **Specific addresses or numbers:** This case examines how well the method handles replies involving precise details such as addresses or phone numbers, which can often be repetitive or irrelevant.
2) **Expressions of gratitude:** This case evaluates how the method deals with responses expressing thanks or appreciation, which may vary in form but should remain contextually appropriate.
3) **General messages:** This case focuses on general or non-specific messages, assessing how well the method generates diverse replies in more open-ended conversational contexts.

By applying the proposed methods to these cases, we aim to demonstrate that the results are satisfactory in terms of response diversity. While our approach effectively addresses the primary issue of diversifying replies, we will provide specific examples of how different types of input messages yield varied responses. These examples will showcase the practical outcomes of the method and highlight its ability to generate a richer and more varied set of suggestions, thus improving the overall user experience in real-world interactions.

3.1 Case 1

The first case we address involves scenarios where the message requests specific phone numbers or addresses. This situation is particularly relevant because smart reply systems must handle requests for such detailed information carefully to avoid providing inappropriate or redundant responses. To illustrate the approach, we refer to Figure 3, which shows the results for handling requests for phone numbers.



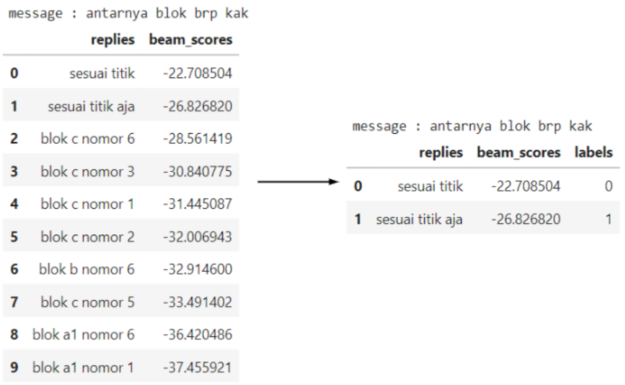
message : minta nomer yang aktif kak

	replies	beam_scores
0	iya	-24.772980
1	nomor inilah	-33.873661
2	nomor ini aktif	-35.775238
3	nomor ini	-37.173729
4	nomor ini aja	-38.654243
5	nomor ini aktif kok	-41.282196
6	nomor ini bisa	-44.051559
7	08126125949	-49.095043
8	081261259409	-49.902973
9	08126121083	-50.670692

message : minta nomer yang aktif kak

	replies	beam_scores	labels
0	iya	-24.772980	1
1	nomor ini aktif	-35.775238	0
2	nomor ini aktif kok	-41.282196	2

Figure 3. Phone numbers



message : antarnya blok brp kak

	replies	beam_scores
0	sesuai titik	-22.708504
1	sesuai titik aja	-26.826820
2	blok c nomor 6	-28.561419
3	blok c nomor 3	-30.840775
4	blok c nomor 1	-31.445087
5	blok c nomor 2	-32.006943
6	blok b nomor 6	-32.914600
7	blok c nomor 5	-33.491402
8	blok a1 nomor 6	-36.420486
9	blok a1 nomor 1	-37.455921

message : antarnya blok brp kak

	replies	beam_scores	labels
0	sesuai titik	-22.708504	0
1	sesuai titik aja	-26.826820	1

Figure 4. Address

The process begins with post-processing, which is a critical step in our algorithm. During this phase, we specifically remove all candidate responses that might include phone numbers or addresses. This step is essential to ensure that the responses provided are not only relevant but also free from potentially sensitive information. Figures 3 and 4 illustrate this process in detail. Figure 3 presents the initial set of candidate responses, which include some that inadvertently contain phone numbers or addresses.

Following the removal of these responses, Figure 4 depicts the filtered set of candidate replies. As expected, after excluding those containing addresses or phone numbers, only two candidate replies remain. This reduction is a direct result of the post-processing step, which ensures that the responses meet the criteria of being free from sensitive information.

For the clustering phase, the number of clusters is configured to be a minimum of two. This configuration is chosen to ensure that the responses are adequately grouped even if the set of candidate replies is small. In this case, although the remaining two replies seem to convey similar meanings, the K-means clustering algorithm separates them into two distinct clusters. This separation occurs because K-means clustering operates on the principle that each data point should belong to the cluster with the nearest mean. Since the algorithm is unable to merge the two replies into a single cluster due to their minimal difference, it creates two separate clusters. This behavior is indicative of the clustering process's sensitivity to even slight variations in the data, which can result in distinct clusters for responses that are otherwise quite similar in meaning.

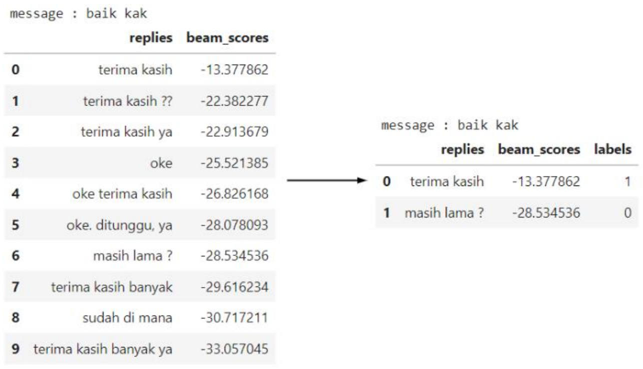
3.2 Case 2

This example focuses on a common type of response message, specifically expressions of gratitude. In many conversations, especially in a customer service context, responses often involve variations of the phrase "thank you." This particular case highlights how frequently such expressions appear and how they are handled by our method.

In Indonesian, the phrase "terima kasih" translates to "thank you." This expression is a fundamental part of polite conversation and can appear in various forms and contexts. To address this, we analyze the candidate responses generated for messages that contain this phrase. Initially, any candidate replies containing the phrase "terima kasih" is manually clustered. This manual clustering step is performed to group these responses without using automated clustering algorithms such as K-means. The goal is to organize these responses based on their specific variations and ensure that each cluster represents a unique form of expressing gratitude.

Once the manual clustering is complete, one representative message is extracted from each cluster to ensure diversity. This extraction provides a diverse set of gratitude expressions that can be used in responses. Following this, K-means clustering is applied to the remaining candidate replies. The application of K-means clustering helps in further grouping the remaining responses based on their semantic content, allowing for a more structured and varied set of replies.

Figure 5 illustrates the results of this process. It clearly shows that the "regular" smart reply system tends to generate a high volume of candidate responses that are primarily variations of "thank you," resulting in a somewhat repetitive set of replies. In contrast, our proposed method demonstrates a more diversified range of responses. By employing manual clustering to handle common expressions of gratitude and subsequently using K-means clustering for the remaining responses, our approach ensures that the final suggestions are more varied and contextually appropriate. This enhancement improves the overall user experience by providing responses that are not only relevant but also more engaging and less monotonous.



message : baik kak

	replies	beam_scores
0	terima kasih	-13.377862
1	terima kasih ??	-22.382277
2	terima kasih ya	-22.913679
3	oke	-25.521385
4	oke terima kasih	-26.826168
5	oke. ditunggu, ya	-28.078093
6	masih lama ?	-28.534536
7	terima kasih banyak	-29.616234
8	sudah di mana	-30.717211
9	terima kasih banyak ya	-33.057045

message : baik kak

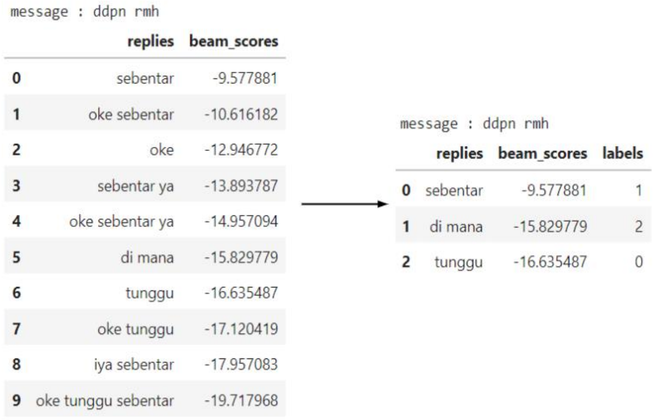
	replies	beam_scores	labels
0	terima kasih	-13.377862	1
1	masih lama ?	-28.534536	0

Figure 5. Most frequent replies

3.3 Case 3

In the final case, we examine the scenario where the input message is "ddpn rmh," which translates to "already in front of the house." This phrase is commonly used in conversations to indicate that someone has arrived at a specific location, typically a house. In response to this type of message, users might expect replies such as "wait a second," "I will be there in a minute," or "where exactly?" These responses are designed to provide additional context or prompt further clarification.

Our smart reply system generates responses to such input messages. However, it is observed that multiple responses produced by the system convey the same underlying meaning. For instance, the system might output several variations of "I will be there shortly" or "please provide more details," resulting in a set of responses that are semantically similar.



message : ddpn rmh

	replies	beam_scores
0	sebentar	-9.577881
1	oke sebentar	-10.616182
2	oke	-12.946772
3	sebentar ya	-13.893787
4	oke sebentar ya	-14.957094
5	di mana	-15.829779
6	tunggu	-16.635487
7	oke tunggu	-17.120419
8	iya sebentar	-17.957083
9	oke tunggu sebentar	-19.717968

message : ddpn rmh

	replies	beam_scores	labels
0	sebentar	-9.577881	1
1	di mana	-15.829779	2
2	tunggu	-16.635487	0

Figure 6. General message

Figure 6 illustrates this situation by comparing the output of the regular smart reply system with that of our proposed method. The figure clearly shows that the traditional smart reply system tends to produce a high number of similar responses, which can be repetitive and lack diversity. On the other hand, our proposed method is designed to address this issue by enhancing the variety of the responses. It achieves this by employing advanced techniques to ensure that the replies generated are more diverse and contextually appropriate.

The cases discussed above collectively demonstrate that our proposed method is effective in generating more varied responses. This diversification is particularly beneficial in practical applications, as it allows companies to offer their customers a broader range of response options. By providing multiple, distinct response choices, companies can better meet the diverse needs and preferences of their customers. This not only improves customer satisfaction by increasing the likelihood of receiving a response that closely matches their expectations but also enhances the overall quality of interaction within the smart reply system.

4. CONCLUSION

In this paper, we have introduced the two-step method to diversify responses in smart replies. Our method begins with the regular smart reply step, producing candidate responses that may have similar meanings. After obtaining the candidates, we apply post-processing techniques, including the removal of uninformative words or characters, representing common replies for the most frequent responses, and clustering to further diversify the candidate responses. We have observed that our proposed method performs exceptionally well in generating diversified replies based on real data from conversations between drivers and customers in Bahasa Indonesia. A potential limitation of the proposed method is its susceptibility to incorrect clustering when K-means proves ineffective. Considering alternative clustering methods may offer a solution to this challenge. Although the proposed method is tested using Bahasa Indonesia, it is also applicable for investigation in other languages using the same approach.

REFERENCES

[1] Shay, M., Davidson, R., Grinberg, N. (2024). EnronSR: A benchmark for evaluating AI-generated email replies. In Proceedings of the International AAAI Conference on Web and Social Media, 18: 2063-2075. https://doi.org/10.1609/icwsm.v18i1.31448

[2] Kannan, A., Kurach, K., Ravi, S., Kaufmann, T., Tomkins, A., Miklos, B., Corrado, G., Lukacs, L., Ganea, M., Young, P., Ramavajjala, V. (2016). Smart reply: Automated response suggestion for email. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 955-964. https://doi.org/10.1145/2939672.2939801

[3] Shah, H., Jaidka, K., Ungar, L., Fagan, J., Grosser, T. (2023). Building a multimodal classifier of email behavior: Towards a social network understanding of organizational communication. Information, 14(12): 661. https://doi.org/10.3390/info14120661

[4] Henderson, M., Al-Rfou, R., Strope, B., Sung, Y.H., Lukács, L., Guo, R., Kumar, S., Miklos, B., Kurzweil, R. (2017). Efficient natural language response suggestion for smart reply. arXiv Preprint, arXiv: 1705.00652. https://doi.org/10.48550/arXiv.1705.00652

[5] Zhang, X., Zhao, J., LeCun, Y. (2015). Character-level convolutional networks for text classification. Advances in Neural Information Processing Systems, 28: 649-657.

[6] Wang, W.C., Feng, S., Gao, W., Wang, D.L., Zhang, Y.F. (2018). Personalized microblog sentiment classification via adversarial cross-lingual multi-task learning. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, pp. 338-348.

[7] Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E. (2016). Hierarchical attention networks for document classification. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 1480-1489.

[8] Devlin, J., Chang, M.W., Lee, K., Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv Preprint, arXiv:1810.04805. https://doi.org/10.48550/arXiv.1810.04805

[9] Cer, D., Yang, Y., Kong, S.Y., Hua, N., Limtiaco, N., John, R.S., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., Sung, Y.H., Strope, B., Kurzweil, R. (2018). Universal sentence encoder. arXiv Preprint, arXiv: 1803.11175. https://doi.org/10.48550/arXiv.1803.11175

[10] Jayaraman, B., Ghosh, E., Inan, H., Chase, M., Roy, S., Dai, W. (2022). Combing for credentials: active pattern extraction from smart reply. arXiv Preprint arXiv: 2207.10802. https://doi.org/10.48550/arXiv.2207.10802

[11] Dias, I., Rei, R., Pereira, P., Coheur, L. (2022). Towards a sentiment-aware conversational agent. In Proceedings of the 22nd ACM International Conference on Intelligent Virtual Agents, pp. 1-3. https://doi.org/10.1145/3514197.3549692

[12] Howard, J., Ruder, S. (2018). Universal language model fine-tuning for text classification. arXiv Preprint, arXiv: 1801.06146. https://doi.org/10.48550/arXiv.1801.06146

[13] LeCun, Y., Bengio, Y., Hinton, G. (2015). Deep learning. Nature, 521(7553): 436-444. https://doi.org/10.1038/nature14539

[14] Mikolov, T. (2013). Efficient estimation of word representations in vector space. arXiv Preprint, arXiv: 1301.3781.

[15] Pennington, J., Socher, R., Manning, C.D. (2014). Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532-1543.

[16] McCann, B., Bradbury, J., Xiong, C., Socher, R. (2017). Learned in translation: Contextualized word vectors. Advances in Neural Information Processing Systems, 30: 6297-6308.

[17] Garten, J., Kennedy, B., Sagae, K., Dehghani, M. (2019). Measuring the importance of context when modeling language comprehension. Behavior Research Methods, 51: 480-492. https://doi.org/10.3758/s13428-019-01200-w

[18] Kim, S.W., Gil, J.M. (2019). Research paper classification systems based on TF-IDF and LDA schemes. Human-Centric Computing and Information Sciences, 9: 1-21. https://doi.org/10.1186/s13673-019-0192-7

[19] Li, Y., Wu, H. (2012). A clustering method based on K-means algorithm. Physics Procedia, 25: 1104-1109. https://doi.org/10.1016/j.phpro.2012.03.206

[20] Rousseeuw, P.J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics, 20: 53-65. https://doi.org/10.1016/0377-0427(87)90125-7