A hybrid approach to hospital quality monitoring based on google maps reviews: Integrating *p*-control charts and bidirectional encoder representations from transformers (BERT)

Rossa Julia Nurfaizah[a], Muhammad Ahsan[a*] and Muhammad Hisyam Lee[b]

[a]*Department of Statistics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia*
[b]*Department of Mathematical Sciences, Universiti Teknologi Malaysia, Johor Bahru, Malaysia*

C H R O N I C L E	A B S T R A C T
	This study investigates the utilization of Google Maps reviews to assess hospital service quality. Patient-generated reviews were analyzed using a sentiment analysis framework incorporating the Bidirectional Encoder Representations from Transformers (BERT) classification model. The p control chart was employed to monitor the distribution of negative sentiment. The results of the sentiment analysis revealed a predominance of positive reviews over negative ones. The BERT classifier achieved excellent performance, with AUC values of 99.95% and 93.72% for training and testing data, respectively. However, the p control chart indicated that the hospital's performance still requires improvement, as several observations fell outside the statistically controlled range. Common patient complaints centered on lengthy wait times and queues, highlighting areas for targeted quality enhancement initiatives. This research demonstrates the potential of leveraging patient feedback to inform hospital quality improvement efforts.

1. Introduction

Health is one of the most important aspects of life because without good health, every human being will have difficulty in carrying out daily activities. The level of health in Indonesia is arguably still far behind other countries. Based on data from The Legatum Prosperity Index 2023, Indonesia ranks 87th out of 167 countries with a health index value of 71.13 out of 100 points. The calculation of health index points is based on six health indicators, namely behavioral risk factors, preventive interventions, care systems, mental health, physical health and mortality rates. To improve the health index, a facility that can support public health is needed, namely a hospital.

Hospitals are healthcare institutions that provide comprehensive, individualized medical services, including inpatient, outpatient, and emergency care. To ensure optimal patient outcomes, hospitals must continuously strive to enhance the quality of their services. Based on ownership, hospitals can be categorized into two primary types: Government Hospitals and Private Hospitals. Government Hospitals are managed by governmental entities, such as the Ministry of Health, local governments, military organizations, or state-owned enterprises. Conversely, Private Hospitals are owned and operated by private entities, often organized as social welfare organizations. Regarding the type of services offered, hospitals can be further classified into General Hospitals and

Specialized Hospitals. General Hospitals treat patients with a wide range of medical conditions, while Specialized Hospitals focus on providing specialized care for patients with specific diseases or conditions, such as cancer, maternity, or other specialized medical needs.

A large, tertiary-care general hospital, operated by the East Java Provincial government, is located in Surabaya, Indonesia. Equipped with a comprehensive array of sophisticated medical equipment, the hospital provides a diverse range of healthcare services. Annually, the facility serves hundreds of thousands of patients, necessitating a continuous focus on maintaining high-quality care. The hospital encourages patients to share their experiences and opinions via Google Maps to facilitate patient feedback and quality improvement. Patients are provided a platform to freely express their satisfaction or dissatisfaction with the services received, without fear of reprisal. By analyzing these patient-generated reviews, the hospital can identify areas of strength and weakness, enabling targeted improvements in service delivery (Pyon et al., 2011). Patient feedback is crucial for ensuring high-quality care, as it can reveal areas where patient satisfaction is lacking, leading to decreased loyalty and potential loss of patients (Michel, 2001). Therefore, patient reviews on Google Maps can be analyzed to gauge sentiment, providing hospitals with valuable insights for enhancing service quality.

Sentiment analysis uses natural language processing (NLP) techniques to identify whether a statement expresses a positive, negative, or neutral opinion (Khan & Baharudin, 2011). Sentiment analysis is a method used to automatically determine the public's feelings or opinions (Nandwani & Verma, 2021; Wankhade et al., 2022) . Previous studies have explored the application of machine learning and natural language processing techniques using different algorithms, such as the Naïve Bayes (Li et al., 2020; Pristiyono et al., 2021; Villavicencio et al., 2021), LSTM (Elfaik & Nfaoui, 2020; Huang et al., 2021; Jin et al., 2020), and SVM (AlBadani et al., 2022; Borg & Boldt, 2020; Obiedat et al., 2022). This study employed a Bidirectional Encoder Representations from Transformers (BERT) model for sentiment analysis. BERT's distinctive feature lies in its capacity to train language models by considering the complete context of a sentence, rather than sequentially processing words from left to right. This bidirectional training approach is superior to traditional methods that rely solely on unidirectional word order (Pota et al., 2020). BERT's bidirectional architecture enables the language model to capture contextual relationships between a word and its surrounding words, considering both preceding and subsequent tokens within a given sequence (Singh et al., 2021). This methodology is empirically demonstrated to outperform alternative approaches in terms of accuracy. Sentiment analysis results are categorized into three classes: positive, neutral, and negative. Negative reviews, indicative of patient dissatisfaction, highlight potential deficiencies in service delivery (Xu & Li, 2016).

One method that can be used to monitor a service process is SPC or Statistical Process Control (Montgomery, 2013). Implementing Statistical Process Control (SPC) for service quality monitoring is crucial for healthcare providers to ensure consistent performance. SPC enables early detection of deviations from desired service standards, allowing for proactive intervention and prevention of adverse patient outcomes. By continuously monitoring and analyzing service data, hospitals can identify emerging issues and implement targeted corrective measures to maintain or improve the overall quality of care (Rasouli & Zarei, 2016). SPC techniques, such as control charts and Pareto diagrams, are essential tools for quality management. Control charts are employed to assess the statistical control of a process, while Pareto diagrams prioritize problem areas for targeted improvement. In this study, the focus is on analyzing negative reviews and defects in hospital services using sentiment analysis. Attribute control charts, specifically P control charts, are appropriate for monitoring the proportion of defects in this context, as they can accommodate varying sample sizes and track defect trends over time. Pareto diagrams can further assist in identifying the most prevalent service issues or defect types experienced by patients.

Several studies have explored sentiment monitoring of mobile application reviews. Apsari, Ahsan, and Lee employed p and Laney p' attribute control charts to assess the quality of rating data and user reviews for the PeduliLindungi application (Apsari et al., 2023). The Convolutional Neural Network (CNN) method is used for sentiment analysis of reviews, which results in an AUC value of 79.38% (Firmansyah & Ahsan, 2023). The customer complaints monitoring with customer review data analytics proves that sentiment analysis can be combined with statistical process control analyses (Pribadi & Ahsan, 2023).

Sentiment analysis of patient reviews on Google Maps can be conducted using the Bidirectional Encoder Representations from Transformers (BERT) classification model. This state-of-the-art technique, which has gained significant traction in recent years, offers several advantages over traditional methods. BERT's ability to capture contextual nuances and long-range dependencies in the text makes it particularly well-suited for analyzing patient reviews, which often contain complex expressions of sentiment. Despite the growing interest in sentiment analysis, the application of BERT to patient reviews remains relatively unexplored. This presents an opportunity for novel research that could provide valuable insights into the patient experience and inform hospital improvement efforts. Given the moderate sample size and variability of the dataset, a p-control chart was selected for data monitoring. This statistical tool allows for the detection of trends and shifts in patient sentiment over time. By analyzing the sentiment expressed in patient reviews, this approach can shed light on the challenges faced by patients, identify emerging issues, and provide valuable insights for hospital administrators.Through the application of BERT and p-control charts, hospitals can gain a deeper

understanding of patient satisfaction and dissatisfaction, identify areas for improvement, and ultimately enhance the quality of care provided. This research can contribute to the development of more patient-centered healthcare systems and improve the overall patient experience.

2. Literature Review

2.1 Text Mining

Text mining, a subfield of data science, involves the extraction of valuable information from textual data through techniques such as data mining, machine learning, natural language processing, information retrieval, and knowledge management (Feldman & Sanger, 2006). Before the text data is analyzed, there is a text preprocessing stage that aims to clean the data from unnecessary information to facilitate the next analysis stage. The stages in text preprocessing are as follows.

a.   Case Folding, which is the process of converting all capital letters into lowercase letters.
b.   Filtering, which is the process of removing unnecessary data such as emoticons, punctuation marks, numbers, urls, repeated characters and spaces.
c.   Tokenizing, which is the process of breaking the original sentence into words or breaking the string sequence into pieces of words.
d.   Normalization, which is the process of normalizing words that were previously not standardized into standard words.

2.2 Classification

Classification, a fundamental task in data science, involves the development of predictive models or functions that differentiate between distinct categories or classes within a dataset. By analyzing the characteristics of known data points, these models aim to accurately predict the class membership of unseen instances (Han et al., 2012). In the classification process, there are two stages, which are as follows.

1.   Learning Step, which is a stage built into the algorithm for classification by analyzing data created from a database of tuples and their associated class labels.
2.   Classification Step, which is a step where the model obtained in the previous step is used to predict the class label on the test data.

2.3 Holdout Validation

Prior to model development, the dataset was partitioned into two distinct subsets: a training set and a testing set. The training set is utilized to train the model, while the testing set is employed to evaluate the model's performance on unseen data. In this study, the holdout validation method was adopted for data partitioning. This approach is well-suited for large datasets and offers the advantage of rapid processing time (Allibhai, 2018). When randomly partitioning data into training and testing sets, there is a risk of overrepresentation or underrepresentation of specific classes. This can lead to a biased dataset that does not accurately reflect the true distribution of the target variable. To mitigate this issue, a holdout stratification technique is employed. This method ensures that each class is proportionally represented in both the training and testing sets, thereby enhancing the generalizability and reliability of the model.

Fig. 1 depicts the stratified holdout validation method, a technique that ensures proportional representation of each class in both the training and testing datasets. In this example, the data is partitioned into an 80:20 ratio, with 80% allocated for training and 20% for testing. This approach helps mitigate potential biases that may arise from an uneven distribution of classes in the subsets. The distribution of data for training and testing in each class is done randomly, so it cannot be ascertained which data will be part of training or testing.

Fig. 1. Illustration of Stratified Holdout Validation

2.4 Confusion Matrix

A confusion matrix is a performance evaluation tool for classification models. It provides a tabular representation that compares the model's predicted class labels with the true class labels of the data points. The rows of the matrix correspond to the actual classes, while the columns represent the predicted classes. By analyzing the distribution of values within the confusion matrix, one can assess the model's accuracy, precision, recall, and other relevant metrics (Han et al., 2012). Confusion matrix can be seen in Table 1.

Table 1
Confusion Matrix

Actual Class	Prediction Class		
	Class 1	Class 2	Class 3
Class 1	X_{11}	X_{12}	X_{13}
Class 2	X_{21}	X_{22}	X_{23}
Class 3	X_{31}	X_{32}	X_{33}

The calculation of each element in the confusion matrix is as follows.

$$TP = X_{11} + X_{22} + X_{33} \tag{1}$$
$$FP = (X_{21} + X_{31}) + (X_{12} + X_{32}) + (X_{13} + X_{23}) \tag{2}$$
$$FN = (X_{12} + X_{13}) + (X_{21} + X_{23}) + (X_{31} + X_{32}) \tag{3}$$
$$TN = N_{total} - (TP + FP + FN) \tag{4}$$

where:

1. True Positive (TP), indicates that the actual positive class is correctly categorized by the classifier.
2. False Positive (FP), indicates that the negative actual class is categorized incorrectly by the classifier result.
3. False Negative (FN), indicates that the positive actual class is categorized incorrectly by the classifier result.
4. True Negative (TN), indicates that the negative actual class is correctly categorized by the classifier result.

Based on the TP, FP, FN and TN values, the accuracy, precision, specificity and sensitivity (recall) values can be obtained with balanced data in each class (Han et al., 2012). The accuracy value describes how accurately the system can classify data correctly. In other words, the accuracy value is the ratio between the correctly classified data and the entire data. The accuracy value can be obtained using Eq. (5).

$$accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{5}$$

Precision is the value of the accuracy of the system to show the correct positive data and negative data. Precision is the ratio of true positive predictions compared to the overall positive predicted results. The precision value can be obtained using Eq. (6).

$$precision = \frac{TP}{TP + FP} \tag{6}$$

Specificity is the correctness of predicting negative data compared to the total amount of negative data. The specificity value can be obtained using Eq. (7).

$$specificity = \frac{TN}{TN + FP} \tag{7}$$

Sensitivity or recall is a value that indicates the success rate of retrieving information about true positive and negative data. Sensitivity is the ratio of true positive predictions compared to the overall true positive data. The sensitivity value can be obtained using Eq. (8).

$$sensitivity \ (recall) = \frac{TP}{TP + FN} \tag{8}$$

If the data used is imbalanced, then the evaluation of the classification model can use Area Under Curve (AUC). The AUC value is a performance indicator for the Receiver Operating Characteristic (ROC) curve that can summarize the performance of the classifier into one value (Bekkar et al., 2013). The AUC value can be obtained by Eq. (9).

$$AUC = \frac{1}{2} \left(sensitivity + specificity \right) \hspace{3cm} (9)$$

The AUC value is always in the interval 0 to 1, if the value is closer to 1, the AUC value is better. The scale and interpretation used to explain the AUC value are described in Table 2.

Table 2
Interpretation of AUC Value

AUC Value	Interpretation
0.9 - 1	Excellent classification
0.8 – 0.89	Good classification
0.7 – 0.79	Fair classification
0.6 – 0.69	Poor classification
$\leq 0.5 – 0.59$	Failure

2.5 Transformers

Transformers, a novel deep learning architecture introduced in 2017, have demonstrated exceptional versatility across a range of natural language processing (NLP) tasks. Initially conceived for machine translation, these models have rapidly evolved to address diverse challenges, including sequence classification, question answering, and language modeling. Their ability to capture complex dependencies and contextual relationships within text has solidified their position as a cornerstone of contemporary NLP research (Kokab et al., 2022). Transformers incorporate an attention mechanism, enabling the model to selectively focus on various segments of the input sequence at each layer, thereby identifying long-range dependencies among words. Unlike traditional sequential models, transformers process all words in a sentence concurrently, facilitating parallel processing. The transformer architecture consists of two primary components: an encoder and a decoder (Vaswani et al., 2017).
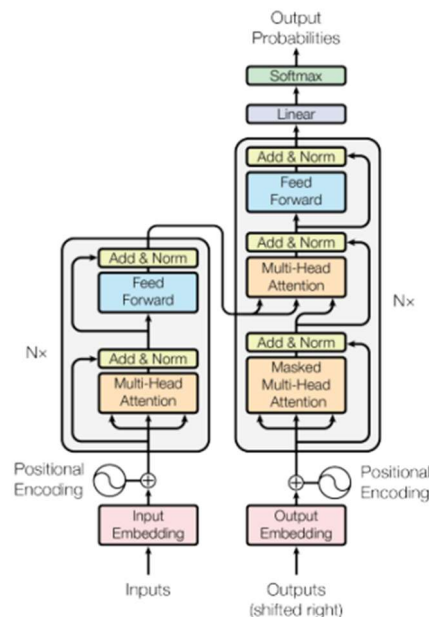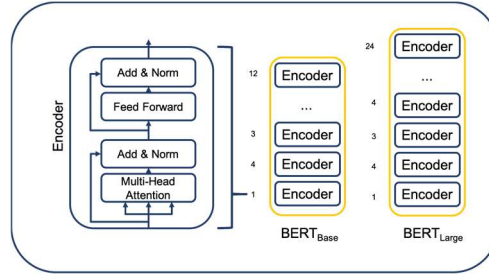

Fig. 2. Transformers Architecture

Fig. 2 illustrates the Transformer architecture, comprising an encoder and a decoder. The encoder processes the entire input text sequence through a stack of six identical layers. Each layer consists of two sub-layers: a multi-head attention mechanism and a feed-forward neural network. Conversely, the decoder generates a predicted output sequence, also utilizing a stack of six identical layers. While the decoder's architecture mirrors that of the encoder, it incorporates an additional sub-layer: masked multi-head attention. This layer strategically masks certain vectors generated by the encoder, preventing the decoder from accessing information about subsequent words in the input sequence. This mechanism discourages the decoder from simply copying the input and encourages it to learn to generate outputs based on the encoded representation of the entire input text.

2.6   Bidirectional Encoder Representations from Transformers

Bidirectional Encoder Representations from Transformers (BERT) is a state-of-the-art language model introduced by Google in 2018. Leveraging the transformer architecture, BERT was trained on a massive corpus of text data to learn contextual representations of words. BERT has demonstrated exceptional performance across a wide range of natural language processing tasks, including question answering, text classification, and sentiment analysis (Alaparthi & Mishra, 2021; Bello et al., 2023). BERT, a bidirectional encoder representation from transformers, is a pre-trained language model that leverages unlabeled text data to learn contextual representations. By considering the surrounding context from both the left and right sides at every layer, BERT is able to capture intricate relationships and nuances within the text (Devlin et al., 2019).


Fig. 3. BERT Architecture

The BERT model architecture, as depicted in Fig. 3, is a multi-layered bidirectional transformer, utilizing only the encoder component. Two primary variants of BERT exist: BERT$_{BASE}$ and BERT$_{LARGE}$, distinguished by their size and computational complexity. The following is a description of each model.

1.   BERT$_{BASE}$: this model is built from 12 encoder layers, 12 self-attention heads and 768 hidden sizes.
2.   BERT$_{LARGE}$: this model is built from 24 encoder layers, 16 self-attention heads and 1024 hidden size.

This study employs a BERT$_{BASE}$ model architecture, incorporating WordPiece embeddings with a vocabulary of 30,000 tokens. WordPiece, a subword tokenization method, is trained on a large dataset to automatically identify and learn meaningful word units, mitigating the Out-of-Vocabulary (OOV) problem. Commonly occurring words are preserved as whole words, while less frequent terms are segmented into subwords or even individual characters.
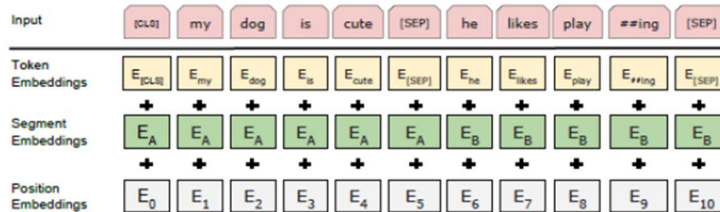
Fig. 4. BERT Input Representation

BERT's input representation, as illustrated in Figure 4, consists of several key components. Initially, the input text is tokenized into individual words. These tokens are then embedded, with BERT introducing two special tokens: a classification token ([CLS]) at the beginning of each sequence and a separation token ([SEP]) at the end of each sentence. To ensure consistent input length, padding is applied using a [PAD] token when sentences are shorter than the specified maximum length. Additionally, BERT incorporates segment embeddings to differentiate between tokens originating from sentence A and those from sentence B. Positional embeddings are also added to encode the relative position of each token within the sequence. The final input to the BERT encoder is a vector formed by summing these token, segment, and positional embeddings. BERT requires all input sentence sequences to be of equal length. The maximum sequence length is typically set to 512 due to the encoder's inherent limitation in producing outputs of that dimension.
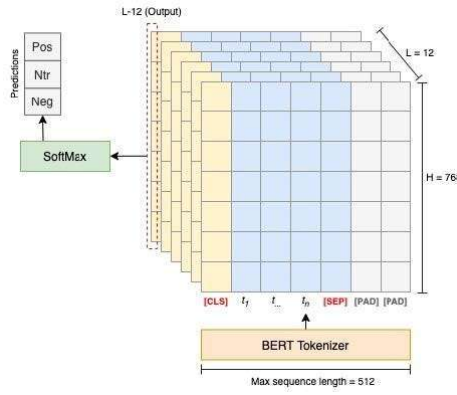
Fig. 5 BERT$_{BASE}$ Structure for Sentiment Analysis

The architectural components of BERT$_{BASE}$ for sentiment analysis are illustrated in Fig. 5. A sequence of words or tokens serves as the input to the model, which is then processed by an encoder stack. Each input is initially transformed into a fixed-length vector of 512 dimensions through an embedding process. To capture the positional information of words within the sequence, positional encoding is applied. This encoding is calculated using Eq. (10), which assigns a unique positional embedding to each word.

$$PE_{(pos,i)} = \begin{cases} \sin\left(\dfrac{pos}{10000^{\frac{i}{d_{model}}}}\right), if \ i \ is \ odd \\ \cos\left(\dfrac{pos}{10000^{\frac{i-1}{d_{model}}}}\right), if \ i \ is \ even \end{cases} \tag{10}$$

where:

pos : position of the word in the sequence or sentence where $pos = 0,1,2,…,n$
i　　: i -th embeddings position where $i = 0,1,2,…,I$
d_{model} : the size of the embeddings vector of each word

After that, the input will enter the encoder process that applies multi-head attention and provides output through the feed-forward network which is then continued by the next encoder. This process continues 12 times. The explanation of the encoder mechanism is as follows (Alammar, 2018).

1.　　Scaled Dot-Product Attention

Scaled Dot-Product Attention is a mechanism employed within the encoder of transformer models to discern the significance of individual tokens within an input sequence. By calculating the weighted sum of the input tokens based on their relevance, this attention mechanism enables the model to prioritize tokens that are more pertinent to the underlying task. This weighting process, known as attention, is crucial for capturing the contextual relationships between elements in the sequence.

Scaled Dot-Product Attention
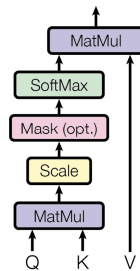
Fig. 6. Scaled Dot-Product Attention

As depicted in Fig. 6, the self-attention layer transforms each input vector into three distinct vectors: query, key, and value. These vectors are derived by multiplying the input embeddings by trainable weight matrices, each of which has a dimensionality of 512. The attention mechanism can be formally defined as a function that maps a query vector and a set of key-value pairs to an output vector. In matrix notation, the query, key, and value vectors can be represented as matrices Q, K, and V, respectively. The calculation of the attention output is governed by Eq. (11).

$$A = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{11}$$

where:
Q : query matrix
K : key matrix
V : value matrix
d_k : key vector dimension

In the attention mechanism, the dot product between the query and key vectors is multiplied and then "scaled" by dividing by the square root of the key vector dimension (d_k) to avoid too large a score and maintain stability in training. This step is done to measure the similarity between the query and key (attention value). After that, the softmax function is applied to the attention value to get the attention weight.

$$softmax(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}} \tag{12}$$

where x_i is the i-th element of the input vector, and n is the number of elements of the input. The softmax function is employed to normalize the attention scores, yielding a probability distribution that reflects the relative importance of each word within the context of the attention mechanism. These normalized attention weights are subsequently multiplied by the value matrix (V), resulting in a weighted representation of the values. This representation is then integrated into the subsequent stages of the encoder architecture. In essence, the attention transformer mechanism leverages the similarity between the query and key vectors to compute attention weights. These weights are then used to selectively combine information from the value vectors, thereby enabling the model to focus on the most relevant contextual elements.

2. Multi-head Attention

Multi-head attention is a mechanism that employs multiple parallel scaled dot-product attention layers. As illustrated in Fig. 7, these layers independently compute attention weights for different representations of the input data. Subsequently, the weighted representations are concatenated and transformed using a linear projection. This approach enables the model to capture complex relationships and dependencies within the input data by simultaneously attending to information from various subspaces at different positions.
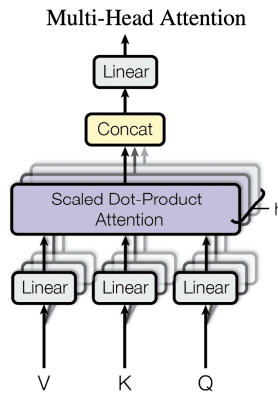

Fig. 7. Multi-head Attention

In use, the attention function is calculated on the set of query vectors simultaneously and packed into a Q matrix. The key and value vectors are also packed into K and V matrices. Denoted the input for multi-head attention is f. Then, f is transformed into d_k dimensional key, query and value.

$$K_j = f.W_j^k \tag{13}$$

$$Q_j = f.W_j^q \tag{14}$$
$$V_j = f.W_j^v \tag{15}$$

where the values of W_j^k, W_j^q, W_j^v are trainable projection matrices. So that the application of scaled dot-product attention from K_j, Q_j and V_j can be seen in Eq. (16).

$$A_j = softmax\left(\frac{Q_j K_j^T}{\sqrt{d_k}}\right)V_j \tag{16}$$

In order to obtain information from different subspace representations at different positions, a parallel attention calculation of H (multi-head attention) is used, which can be obtained by Eq. (17).

$$MH = Concat\left(\{head_j\}_{j=1}^H\right)W^O \tag{17}$$

where $head_j = A_j$, $W^O \in \mathbb{R}^{Hd_k \times d_{model}}$ and $d_k = d_{model}/H$.

The multi-head attention mechanism allows the model to jointly attend to information from different representations at different positions and focus on the relevant parts of the words in the sentence or sequence so as to improve the accuracy of the output.

3.   Feed-forward Network

The multi-head attention layer will produce an output in the form of the sum of the weight of the value vector, which will then enter the feed-forward network stage for each position. The feed-forward network stage consists of two linear transformations with the ReLU (Rectified Linear Unit) activation function. The notation of the feed-forward network can be seen in Equation 18 where x is input, W_1 and W_2 are weights, b_1 and b_2 are biases.

$$FFN(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{18}$$

Each sub-layer of the encoder is followed by a residual connection formed by adding the initial query input (Q) to each residual connection output. Normalization is put after the residual connection to stabilize and speed up the training process (Vaswani et al., 2017). After passing through all encoders, each token per position outputs a vector with a hidden size of 768. For the sentiment analysis process, the output that is considered is the output of the first position, token [CLS], because it is considered to average the word tokens to get the vector of sentences. The vector is used as input for the classifier. Later the classifier layer will produce logits or rough probability prediction results from the sentence to be classified. The formula to get the logits value can be seen in Eq. (19).

$$Logits = W_{output}.h_{CLS} + b_{output} \tag{19}$$

where $W_{output} \in \mathbb{R}^{n_{kelas} \times d_{model}}$ is a weight matrix that converts the [CLS] representation into logits for each class and b_{output} is the bias. Next, the softmax function will convert logits into probabilities by taking the exponent of each logits value so that the sum of each logits probability is exactly 1 where the probability value will be between 0 and 1. The class that has the highest probability will be the class of classification prediction results. If there are logits $z = (z_1, z_2, \dots, z_K)$ then the softmax function used to convert logits into probability can be seen in Eq. (20).

$$\sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} ; for\ i = 1, 2, \dots, K \tag{20}$$

where   :
$\sigma(z_i)$: probability value of the i-th class
z_i : logits value of the i-th class
e^{z_i} : exponential of the i-th class logits

During the BERT training process, the weight value of the model will always be updated by performing loss calculation, gradient calculation and optimizer update. At the loss calculation stage, the loss value between the model prediction and the actual label is calculated. The formula used to calculate the loss value is as follows.

$$Loss = -\sum_{i=0}^{2} y_i . \log_{(\sigma(z_i))} \tag{21}$$

where y_i is actual label value for class i (0 or 1). After that, the loss gradient value is calculated against the predicted probability with the following formula.

$$Loss\ Gradient = \frac{\partial Loss}{\partial \sigma(z_i)} = \sigma(z_i) - y_i \tag{22}$$

In the BERT model, the gradient calculation is performed for each weight in the model using the chain rule. The gradient of the weights in each layer can be calculated using the Eq. (23).

$$\frac{\partial L}{\partial W} = \frac{\partial L}{\partial Output} . \frac{\partial Output}{\partial W} \tag{23}$$

where $\frac{\partial L}{\partial W}$ is the loss gradient of weight W, $\frac{\partial L}{\partial Output}$ is the loss gradient of output layer and $\frac{\partial Output}{\partial W}$ is the output layer gradient of weight W.

After calculating the gradient, the AdamW optimizer is used to update the weights based on the gradient. In AdamW, the weight update for each parameter is done by considering the first and second moments of the gradient as well as an explicit setting for weight decay. The steps and formulas used in the AdamW algorithm are as follows.

a. Initialize the first moment (m) and second moment (v) where $m_0 = 0$ and $v_0 = 0$ and initialize the learning rate (α), $\beta_1 = 0{,}9$, $\beta_2 = 0{,}999$, $\epsilon = 10^{-8}$ and $\lambda =$ weight decay.
b. Update the first moment and second moment

- Gradient of loss with respect to weight $g_t = \frac{\partial L}{\partial W_t}$
- First moment update (exponential average of the gradient)
$$m_t = \beta_1 m_{t-1} + (1 - \beta_1)g_t \tag{24}$$
- Second moment update (exponential mean of the squared gradient)
$$v_t = \beta_2 v_{t-1} + (1 - \beta_2)g_t^2 \tag{25}$$

c. Calculate the bias corrected estimates
- First moment bias correction
$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \tag{26}$$
- Second moment bias correction
$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \tag{27}$$

d. Update parameters with weight decay
$$W_{t+1} = W_t - \alpha\left(\frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} + \lambda W_t\right) \tag{28}$$

The weight update process in model training is carried out continuously until it reaches a predetermined epoch. BERT combines two training frameworks, namely pre-training and fine-tuning, which can be seen in Figure 8. Pre-training is the stage when the model learns the language and context by performing Mask Language Modeling (MLM) and Next Sentence Prediction (NSP) simultaneously. MLM allows the representation to combine the left context and the right context, thus training the bidirectional transformer in depth and NSP which combines the representation of the pre-train text pair. Meanwhile, in the fine-tuning stage, the model is trained with labeled data.
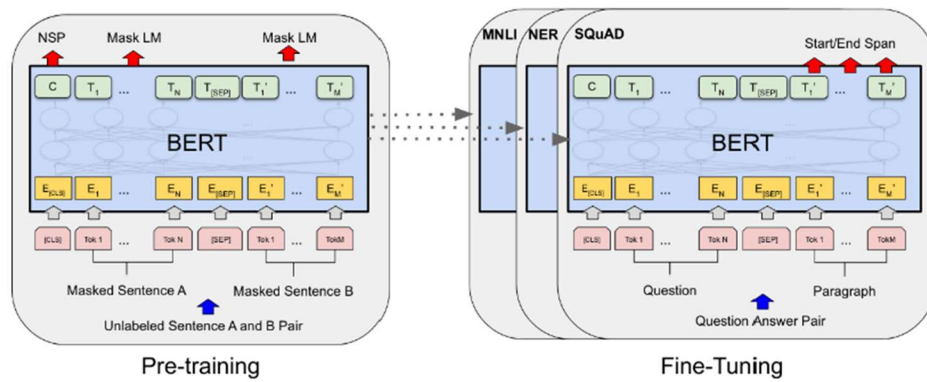
Fig. 8. Pre-training and Fine-tuning Illustration

2.6.1 Masked Language Modelling (MLM)

Masked Language Modeling (MLM) is a technique employed in natural language processing to enhance a model's understanding of context and word relationships. In this approach, a portion of the words within an input sentence are randomly replaced with masked tokens. The model is then tasked with predicting the original identity of these masked words based on the surrounding context. To accomplish this, a classification layer is typically added to the top of the encoder's output. This layer transforms the contextual representation into a probability distribution over the vocabulary. By multiplying the output vector by the embedding matrix and applying softmax, the model can calculate the likelihood of each word appearing in the masked position. Fig. 9 visually illustrates the MLM process.
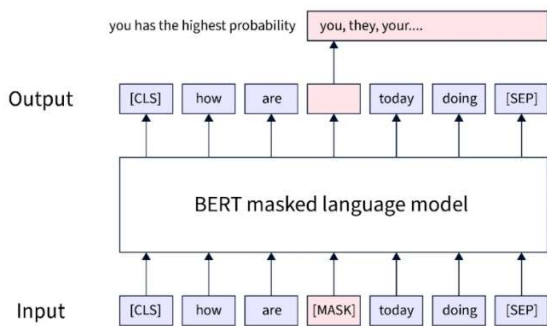
Fig. 9. Masked Language Modelling (MLM) Process
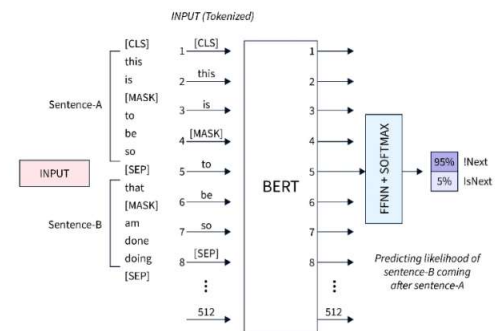

Fig. 10. Next Sentence Prediction (NSP) Process

2.6.2 Next Sentence Prediction (NSP)

Next Sentence Prediction (NSP) is a key component of BERT's training process, designed to enhance the model's comprehension of inter-sentential relationships. The model is presented with pairs of sentences and tasked with determining whether the second sentence is a logical continuation of the first. To ensure a diverse training set, 50% of the pairs are genuine, while the remaining 50% consist of a random sentence from the corpus. Fig. 10 illustrates the NSP process. The output of pre-training is a BERT model (learned weight) and 30,000 embedding vectors. These vectors are then used to represent the word as input to the trained model. The model then transforms the word representation based on the corresponding surrounding word (sentence context).

2.6.3 Fine-tuning BERT

The pre-trained Bidirectional Encoder Representations from Transformers (BERT) model serves as a versatile foundation for tackling various natural language processing (NLP) tasks. By incorporating additional layers tailored to specific objectives, BERT can be adapted to address diverse NLP challenges. To train a model for a particular task, the classifier component is fine-tuned, involving minor adjustments to the BERT model during the training process. This fine-tuning process is relatively straightforward due to the inherent flexibility of the self-attention mechanism within BERT's transformer architecture. This mechanism enables BERT to effectively model a wide range of tasks, encompassing both single-sentence and paired-sentence scenarios, through the strategic swapping of inputs and outputs as needed. Hyperparameter optimization is a critical step in fine-tuning NLP models.

While pre-training typically employs consistent hyperparameters, adjustments to batch size, learning rate, and the number of epochs is often necessary during fine-tuning to tailor the model to specific NLP tasks. Empirical evidence suggests that batch sizes of 16 and 32, in conjunction with Adam optimizer learning rates ranging from 5e-5 to 2e-5, and training epochs between 2 and 4, frequently yield satisfactory results across various NLP applications. However, it's important to recognize that the optimal hyperparameter values can vary depending on the specific task and dataset involved (Devlin et al., 2019). Fig. 11 shows an illustration of the BERT fine-tuning process for classification.
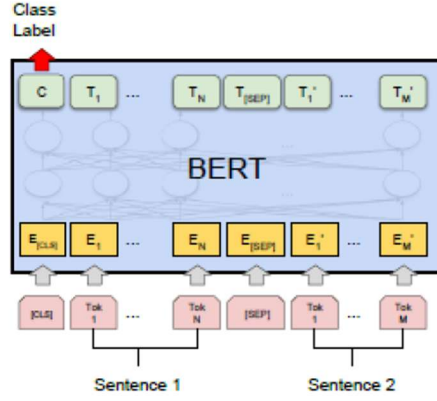
Figu. 11. Illustration of BERT Fine-tuning Process for Classification

2.6.4 IndoBERT

IndoBERT, a pre-trained language model specifically designed for Indonesian, is built upon the foundational architecture of BERT. Trained on the extensive Indo4B dataset, which comprises over 23 gigabytes of Indonesian text data, IndoBERT has been exposed to a vast corpus of both formal and informal language. This diverse training data, sourced from various domains including social media, blogs, news articles, and websites, equips IndoBERT with a robust understanding of the nuances and complexities of the Indonesian language (Wilie et al., 2020). There are four models in IndoBERT, namely IndoBERT$_{BASE}$, IndoBERT$_{LARGE}$, IndoBERT-lite$_{BASE}$, IndoBERT-lite$_{LARGE}$. In terms of accuracy, IndoBERT$_{LARGE}$ has higher accuracy, but the memory required is very large and the time required to run the system is very long. So in this research, the IndoBERT$_{BASE}$ model is used with the number of encoders 12 layers, 12 self-attention heads and 768 hidden sizes.

2.7 P-Control Chart

The p-chart is a statistical process control tool designed to monitor the proportion of defective units within a population. It operates under the assumption of a binomial distribution. Unlike other control charts that require consistent sample sizes, p-charts can accommodate varying numbers of observations. To construct a p-chart, the proportion of defective items in each sample is calculated and plotted against time. This visual representation aids in identifying trends or patterns indicative of process variability or shifts. The sample data taken is binomially distributed so it can be assumed that the proportion of defects in all products is p. Thus, the probability of the number of defects in a sample of size n is x units is as follows.

$$P(x; n; p) = C_x^n p^x = \frac{n!}{x!(n-x)!} p^x (1-p)^{n-x}, x = 0,1,2,\ldots,n \tag{29}$$

where if there is $X \sim Binomial(n, p)$ then the expectation value of X and the variance value of X are as follows.

$$E(X) = np \tag{30}$$
$$Var(X) = np(1-p) \tag{31}$$

If there are as many as Di defective units in sample i, then the proportion of defects in the i-th sample can be formulated as follows.

$$\widehat{p_i} = \frac{D_i}{n_i} \tag{32}$$

where   :

$\widehat{p_i}$: Proportion of negative reviews in the i-th subgroup ($i=1,2,\ldots,m$)
D_i : The number of negative reviews in each sample in the i-th subgroup ($i=1,2,\ldots,m$)
n_i : Sample size in the i-th subgroup ($i=1,2,\ldots,m$)

The expected value and variance of the proportion of defects in the i-th subgroup $\hat{p}_i$ are as follows.

$$E(\hat{p}_i) = E\left(\frac{D_i}{n_i}\right) = \frac{E(D_i)}{n_i} = \frac{n_i p}{n_i} = p \tag{33}$$

$$Var(\hat{p}_i) = Var\left(\frac{D_i}{n_i}\right) = \frac{1}{n_i^2}Var(D_i) = \frac{1}{n_i^2}(n_i p(1-p)) = \frac{p(1-p)}{n_i} \tag{34}$$

where the value of p is the actual defect proportion value in the population. If the value of p is unknown, then the value of p can be estimated using the average value of the proportion of defects $(\bar{p})$ which is formulated as follows.

$$\bar{p} = \frac{\sum_{i=1}^{m} D_i}{\sum_{i=1}^{m} n_i} \tag{35}$$

The control limits and center lines used in the p-control chart can be calculated using the following equation.

$$Upper\ Control\ Limit\ (UCL) = \bar{p} + 3\sqrt{\frac{\bar{p}(1-\bar{p})}{n_i}} \tag{36}$$

$$Central\ Line\ (CL) = \bar{p} \tag{37}$$

$$Lower\ Control\ Limit\ (LCL) = \bar{p} - 3\sqrt{\frac{\bar{p}(1-\bar{p})}{n_i}} \tag{38}$$

3. Research Methodology

3.1 Data Source

The research utilized secondary data consisting of patient reviews of Hospital in Surabaya City on Google Maps. These reviews were collected over a period spanning from January 1, 2016, to December 26, 2023. The data was segmented into two distinct phases for process control analysis: Phase I, encompassing reviews from January 1, 2016, to December 31, 2022, and Phase II, comprising reviews from January 1, 2023, to December 26, 2023.

3.2 Research Variable

The research variables used in the Bidirectional Encoder Representations from Transformer (BERT) classification process consist of predictor variables in the form of patient reviews of Hospital on Google Maps and response variables in the form of sentiment class classification of the patient reviews. Labeling of reviews is done using a rating value, where for rating values 1-2 are categorized as negative classes, rating values 3 as neutral classes, and rating values 4-5 as positive classes. In the neutral class reviews will be labeled manually to determine whether the review is truly neutral or there is a tendency of sentiment towards positive or negative. Reviews that tend to be positive are labeled as positive and those that tend to be negative are labeled as negative. While the research variables used for the p-control chart are the number of negative classes from classification using BERT on patient reviews at Hospital. Negative class reviews can be grouped based on the type of defect or obstacle experienced by the patient. Determination of the defect type category is done by identifying keywords from each negative class review. The types of defects in Hospital services based on patient reviews are shown in Table 3.

Table 3
Type of Service Defect

Category	Defect Type
1	Long service time and long queues
2	Unfriendly staff (security guards and other staff)
3	Obstacles related to National Healthcare Insurance
4	The parking lot is always full
5	Inadequate facilities
6	Health workers are not professional enough
7	Obstacles in the referral process
8	Obstacles during registration
9	Obstacles related to services in the emergency room
10	Other factors

3.3 Analysis Step

The analysis steps conducted in this research are explained as follows.

1. Scraping hospital patient review data on Google Maps using Outscraper.
2. Labeling sentiment classes based on the ratings given by patients on Google Maps.
    a. Negative class for rating 1-2 as well as rating 3 which leans towards negative sentiment.
    b. Neutral class for rating 3 whose reviews are completely neutral, in the form of constructive questions or suggestions.
    c. Positive class for rating 4-5 as well as rating 3 that leans towards positive sentiment.
3. Perform preprocessing on patient review text data.
    a. Perform case folding, which converts the entire text to lowercase.
    b. Perform filtering, which is the removal of unneeded data such as punctuation marks, emoticons, numbers, urls, repeated characters and spaces.
    c. Perform tokenizing, which is the process of separating a series of strings (sentences) into string pieces (words).
    d. Perform normalization, which is the process of normalizing words that were previously not standardized into standard words.
4. Perform descriptive statistical analysis on patient review data in the form of pie charts and bar charts.
5. Create pareto diagrams for the analyzed defect types.
6. Classifying the pre-processed review data with the BERT method.
7. Perform monitoring using reviews data.
8. Provide improvement recommendations to the management of Hospital.
9. Draw conclusions and suggestions.

4. Analysis and Discussion

4.1 Data Pre-processing

Before analysis, a pre-processing stage is needed first so that the data becomes easier to process. The process carried out is as follows.

a. Checking the scrapping data information which consists of 6 variables, namely author_title reviewid, review_text, review_rating, review_datetime_utc, and label (actual).
b. Renaming variables to make it easier to process and analyze.
c. Checking the missing values in the data, where from 2872 patient review data, there are 954 data that have no comments or only ratings. Thus, 954 data were deleted and not used in the study.
d. Installing and importing several libraries to perform text mining.
e. Removing emojis that are listed in the reviews.
f. Removing characters other than letters such as numbers, punctuation marks and some other symbols.
g. Change the letters in the review to lowercase.
h. Performing word normalization so that words that have the same meaning can be written in the same word form.

Table 4 is an example of the data cleaning results used in this analysis.

Table 4
Example of Data Preprocessing Steps

Process	Result
Original Data	*Selama suami dirawat di pplk lt 3 kami sangat puas dgn pelayanan dokter dan perawat sangat baik dan ramah.*
Removing emoji	*Selama suami dirawat di pplk lt 3 kami sangat puas dgn pelayanan dokter dan perawat sangat baik dan ramah.*
Removing characters other than letters	*Selama suami dirawat di pplk lt kami sangat puas dgn pelayanan dokter dan perawat sangat baik dan ramah*
Convert the letters in the review to lowercase letters	*selama suami dirawat di pplk lt kami sangat puas dgn pelayanan dokter dan perawat sangat baik dan ramah*
Perform word normalization	*selama suami dirawat di pplk lantai kami sangat puas dengan pelayanan dokter dan perawat sangat baik dan ramah*

4.2   Review Data Characteristics

After pre-processing, the next step is to analyze using descriptive statistics to determine the characteristics of the data used. In this research, pie charts and bar charts were used as descriptive statistical analysis tools. The pie chart shows the proportion comparison between negative, neutral and positive classes based on the rating. Fig. 12 shows the sentiment class comparison of patient reviews.
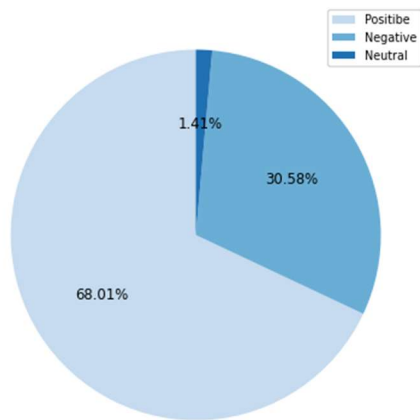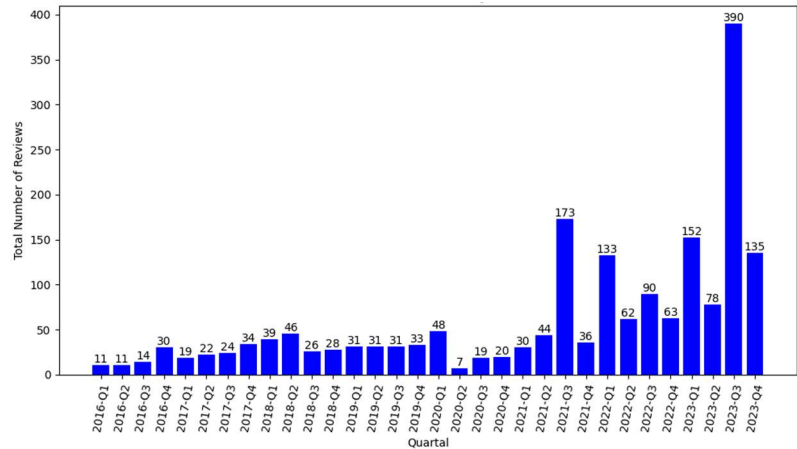

Fig. 12. Actual Sentiment Class Comparison

Fig. 13. Total Number of Reviews by Quarter Year

Based on Fig. 12, it can be seen that 68.01% of reviews are categorized as positive sentiments, 30.58% are categorized as negative sentiments, and 1.41% are categorized as neutral sentiments. This indicates that when viewed based on the rating, 68.01% of the total patients who commented were satisfied with the services, then 30.58% of patients who commented were still not satisfied with the services provided. While 1.41% of patients who commented gave a neutral rating to the service. So it can be concluded that when viewed from the rating, the service at the Hospital is quite good, marked by the number of positive sentiments that are more than other sentiment classes. In addition, it can also be seen that the number of reviews in each sentiment class is different so that it can be said that the patient review data is not balanced or imbalanced. To classify imbalanced data, a stratified holdout validation technique is needed to divide the data of each class proportionally. The division of data using this method guarantees that the training and testing data will have representatives from each class with the same percentage. Next, a bar chart will be created showing the actual number of reviews each quarter of the year which can be seen in Fig. 13. Based on Fig. 13, the most reviews occurred during the 3rd quarter of 2023. After tracing, the highest number of reviews occurred in August 2023, which amounted to 238 reviews. This may be due to the appointment of the hospital as the East Java Plenary and Main Hospital by the Ministry of Health in July 2023. So that more and more people seek treatment and are referred to this hospital for further treatment due to higher quality and qualified infrastructure. With the large number of people seeking treatment and the high use of social media today, it will encourage them to share their experiences in the form of reviews related to hospital services. On the other hand, the lowest number of reviews occurred during the 2nd quarter of 2020, namely in April 2020 which amounted to 0 reviews. This might have happened because in the 2nd quarter of 2020, the first time a Covid-19 case was detected in Surabaya. The existence of this case makes people more vigilant and tend not to leave the house, especially to the hospital for fear of being infected. So that not many people seek treatment and they tend not to provide reviews related to services at Hospital. Furthermore, Figure 14 shows the number of negative reviews every quarter of the year.
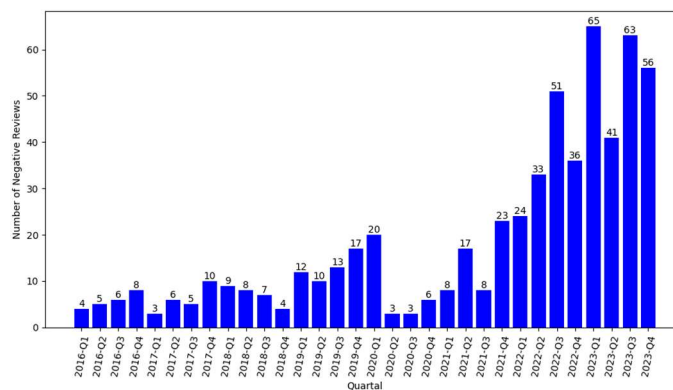

Fig. 14. Number of Negative Reviews by Quarter Year

Based on Fig. 14, it can be seen that the highest number of negative reviews occurred during the first quarter of 2023. This may happen because there is a system change in the service, where the Hospital implements an independent information technology (IT) system by January 2023. With this system change, it requires the hospital to re-register thousands of patient data to connect with National Healthcare Insurance. So that services for patients sometime after the system was implemented experienced delays and took longer. Therefore, this system change can trigger people to give negative reviews about existing services. Meanwhile, the lowest number of negative reviews occurred during the first quarter of 2017, the second quarter of 2020 and the third quarter of 2020. Negative reviews given by the community can be used as a sign that there are services that are lacking and cannot meet community expectations. Therefore, people express their complaints and criticisms through the Google Maps comments column. Table 5 will show the percentage of negative reviews in each month from January 2016 to December 2023 where this value is obtained from the number of negative reviews (actual data) compared to the number of reviews in that month multiplied by 100%.

Table 5

Percentage of Negative Reviews Each Month

| | Percentage of Negative Reviews Each Month | | | | | | | |
	2016	2017	2018	2019	2020	2021	2022	2023
January	0.00%	25.00%	50.00%	25.00%	47.82%	0.00%	42.85%	73.80%
February	100.00%	16.67%	7.14%	40.00%	38.88%	25.00%	55.56%	31.57%
March	16.67%	0.00%	9.09%	55.56%	28.57%	50.00%	7.92%	30.18%
April	100.00%	42.85%	20.00%	33.33%	-	20.00%	60.00%	52.17%
May	40.00%	0.00%	13.33%	41.17%	50.00%	85.71%	52.94%	60.00%
June	40.00%	42.85%	18.18%	12.50%	40.00%	33.33%	50.00%	46.67%
July	66.67%	14.28%	30.00%	41.67%	33.33%	3.12%	54.16%	37.14%
August	16.67%	40.00%	0.00%	18.18%	0.00%	20.00%	63.33%	7.56%
September	60.00%	16.67%	40.00%	75.00%	12.50%	25.00%	52.78%	23.17%
October	12.50%	18.18%	0.00%	40.00%	16.67%	80.00%	34.78%	24.56%
November	40.00%	25.00%	22.22%	71.42%	28.57%	53.84%	65.00%	55.81%
December	25.00%	45.45%	20.00%	54.54%	42.85%	61.53%	75.00%	51.42%

Based on Table 5, the highest percentage of negative reviews occurred in February and April 2016 at 100%. This may be due to the low public awareness of the comment feature on Google Maps in 2016, as well as concerns about the privacy and security of personal data online. As a result, many people are reluctant to leave comments on Google Maps. Even if someone does leave a review, they usually tend to leave a negative review because when someone has a bad experience, they usually have a very high desire to express their dissatisfaction. Meanwhile, if they have a good experience, they tend not to leave a review because they are satisfied with the service.

4.3   Characteristics of Defect Categories

Pareto diagrams can be used to characterize the category of defects in patient reviews related to hospital services. This diagram aims to show which problems have the highest frequency so that they can be resolved as quickly as possible. Before making a pareto diagram, categorization of each negative review or defect is done. The step taken to obtain the defect category is to create a word cloud from all negative review data.

Fig. 15. Word Cloud Negative Reviews

Based on the words that appear in Figu. 15, defect categories can be identified and arranged according to Table 3 which will then be used as the basis in determining the type of defect category in the data analysis per month. To create a pareto diagram, it is necessary to first define keywords for each defect category as follows.

Category 1 : 'pelayanan', 'pelayanannya', 'antri', 'antrian', 'lama', 'panjang', 'menunggu', 'nunggu'.
Category 2 : 'petugas', 'karyawan', 'satpam', 'staff', 'staf'', 'bentak', 'marah', 'sopan', 'ramah', 'judes', 'ketus'.
Category 3 : 'bpjs', 'kelas'.
Category 4 : 'parkir', 'parkiran', 'lahan', 'mobil', 'motor', 'penuh'.
Category 5 : 'fasilitas', 'ac', 'lift', 'ruang', 'ruangan', 'toilet', 'kamar', 'kotor', 'jorok'.
Category 6 : 'dokter', 'dokternya', 'perawat', 'perawatnya', 'suster', 'profesional'.
Category 7 : 'rujuk', 'rujukan'.
Category 8 : 'loket', 'registrasi', 'daftar', 'pendaftaran'.
Category 9 : 'igd', 'ugd'.
Category 10 : Words other than those listed in categories 1-9 in negative reviews

The Pareto diagram of the number of patient complaints related to hospital services is shown in Fig. 16 below.
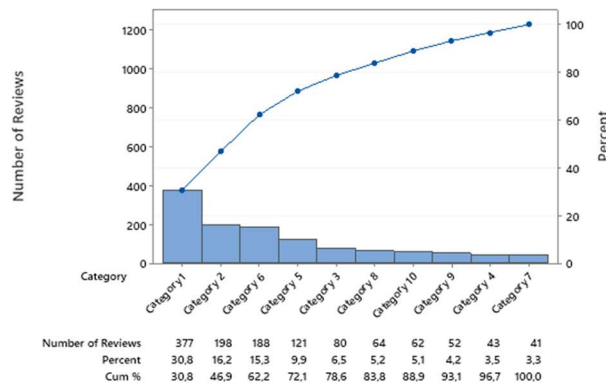

Number of Reviews	377	198	188	121	80	64	62	52	43	41
Percent	30,8	16,2	15,3	9,9	6,5	5,2	5,1	4,2	3,5	3,3
Cum %	30,8	46,9	62,2	72,1	78,6	83,8	88,9	93,1	96,7	100,0

Fig. 16. Pareto Chart of Patients Obstacles

Based on Fig. 16, it can be seen that the highest number of obstacles or defects complained about by patients is category 1, namely long service and long queues with a value of 30.8%. This needs further attention because this queue makes the service inefficient, where many patients have come from the morning but have only finished being served in the afternoon. The second most common obstacle is category 2, namely officers who are less friendly with a value of 16.2%. People who want treatment and come to the hospital are often greeted by security guards and other staff who are fierce and bitchy, making patients feel uncomfortable in treatment. The third most common obstacle is category 6, namely health workers who are less professional with a value of 15.3%. Patients complained that medical staff, both doctors and nurses, were less responsive and did not come immediately when really needed. Some patients also complained about the lack of performance of the nurses because in the process of installing the infusion it had to be done many times before the infusion could be installed. In addition, there are still several other obstacles according to their respective disability categories where the description of this category refers to Table 3.

4.4 Sentiment Analysis of Patient Reviews using Bidirectional Encoder Representations from Transformers (BERT)

The data used for the classification process using BERT is patient review data which has been cleaned in the previous text mining process. Later, this classification will produce categories of several unlabeled items into a certain discrete class data set. Before analysis, the review data is first divided into training data and testing data using the stratified holdout validation method with a ratio of 80%: 20%. This division will make each class, whether positive, neutral or negative, proportionally divided into training data and testing data. Table 6 shows the results of the division of training data and testing data.

Table 6
Division of Training Data and Testing Data

Type	Class	Amount	Total
Training Data	Positive	1039	
	Neutral	22	1528
	Negative	467	
Testing Data	Positive	260	
	Neutral	5	382
	Negative	117	

Before classification using the BERT algorithm, the review data must first be adjusted to the input that can be accepted by BERT. Therefore, BertTokenizer is needed, a tokenizer that aims to tokenize sentences and produce input that is suitable for the BERT algorithm. Fig. 17 shows an example of BERT input.
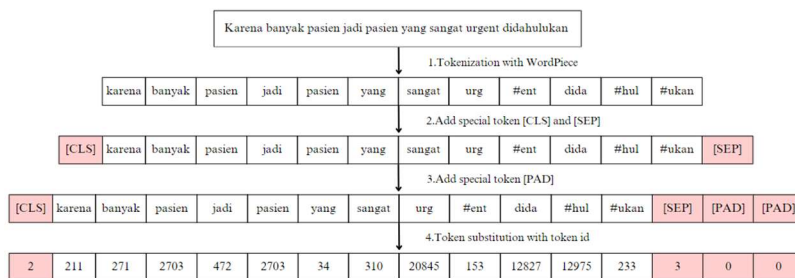
Fig. 17. Example of BERT Input

Fig. 17 shows an illustration of BERT input using BertTokenizer, where after obtaining a clean dataset the results of preprocessing are broken into words. For words that are not in the vocabulary (out of vocabulary), separation is done into subwords with the ## symbol. Furthermore, a special token [CLS] is added at the beginning of the sentence and a token [SEP] at the end of each sentence. BERT accepts input with a maximum length of 512 inputs. If the sentence is shorter than 512, it will be padded by adding [PAD] token and if the sentence is longer, it will be deleted to 512 inputs. After that, each token is coded according to the vocabulary index, where the token [CLS] has id 2, the token 'karena' has id 211 and so on. The following is the result of the BertTokenizer that has been done.

$$\begin{bmatrix} 2 & 211 & 271 & 2703 & \cdots & 0 & 0 \\ 2 & 776 & 2620 & 1073 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 2 & 1753 & 1612 & 724 & \cdots & 0 & 0 \end{bmatrix}$$

Each review will be represented by one vector in the matrix above, where each vector will be 1×512 in size. So the size of the matrix above is 1910×512 where 1910 is the number of reviews and 512 is the dimension of each review vector. This matrix will be used as input for classification using the BERT method. Next, the adjusted input will be passed into the BERT network, which is a stack of 12 layer transformers encoder. Each encoder layer has two sub layers, the first is a multi-head self-attention mechanism, and the second is a fully connected feed forward network. After passing through all encoders, the output vector of each token is obtained. However, only the output vector of token [CLS] will be used as the input vector for the classifier. In this research, a fine-tuning technique is used with the IndoBERT-base-p1 model, one of the models that uses the BERT-base architecture. This technique uses a pre-trained model and only learns a little more to reach the optimal point on a new task. This model has been trained using 4 billion words with approximately 250 million formal and colloquial sentences in Indonesian (Wilie et al., 2020). This research uses the Transformers library provided by HuggingFace. The Transformers library has a BertForSequenceClassification class designed for classification tasks. The BertForSequenceClassification class works by entering the output of the pooler to calculate logits. It then applies a softmax function to the resulting logits values to get a class prediction value for each review. To get the best results, several combinations of hyperparameters are used in the modeling.

Table 7
Comparison of Hyperparameter Combinations in the BERT Model

No	Hyperparameter Combinations	AUC		Accuracy	
		Training Data	Testing Data	Training Data	Testing Data
1	*Batch Size* 16 *Learning Rate* 10^{-5} *Epoch* 5	99.32%	92.21%	98.17%	91.62%
2	*Batch Size* 16 *Learning Rate* 2×10^{-5} *Epoch* 10	100%	89.92%	100%	91.88%
3	**Batch Size 16** **Learning Rate 2×10^{-5}** **Epoch 5**	**99.95%**	**93.72%**	**99.61%**	**92.15%**
4	*Batch Size* 16 *Learning Rate* 2×10^{-5} *Epoch* 10	100%	89.90%	100%	91.36%

Table 7 shows several combinations of hyperparameters built to obtain the best accuracy and AUC values for the BERT model to be used. Of the four combinations, it is found that the 3rd combination is the best model where for training data the accuracy value is 99.61% and the AUC value is 99.95%, which means that the AUC value on training data classification is excellent classification. As for the testing data, the accuracy value is 92.15% and the AUC value is 93.72%, which means that the AUC value on the testing data classification is excellent classification. Model 3 is considered the best combination because the resulting model can classify the testing data better than other combinations. It can be seen that models 2 and 4 have higher training AUC values than model 1, but these two models are not used as the best combination because the models produced during training are very good, resulting in poor performance on the testing data. Fig. 18 shows the confusion matrix of the best model chosen, Model 3.
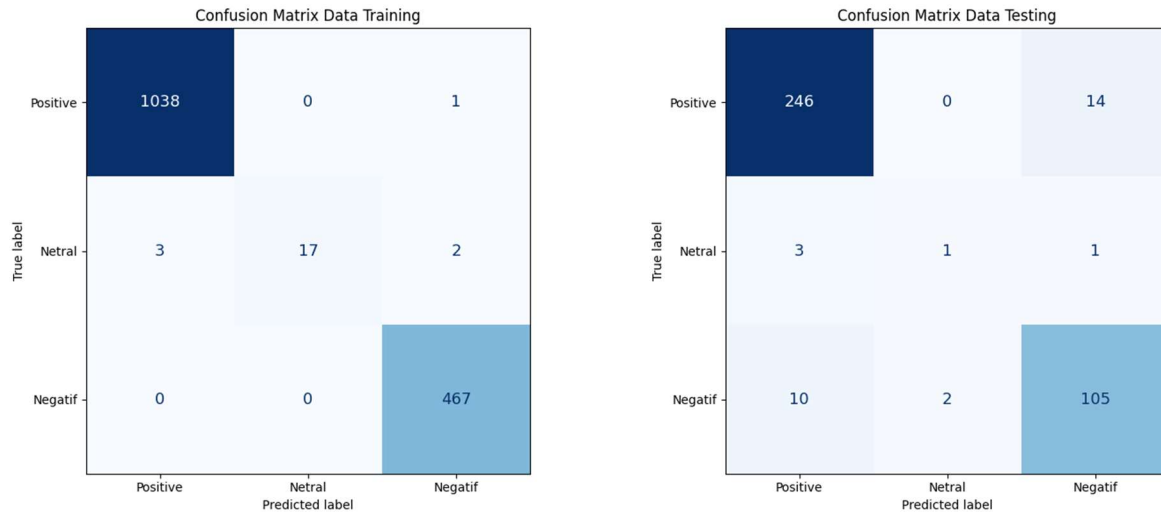
Fig. 18. Confusion Matrix of the Best BERT Model

Based on Fig. 18, it can be seen that from 1528 training data, the model is able to classify as much as 1522 data correctly. While from 382 testing data, the model is able to classify as much as 352 data correctly. Of the 1910 data, 1300 data were classified as positive sentiment, 20 data were neutral sentiment and 590 data were negative sentiment. Furthermore, a pie chart is made comparing the sentiment class of patient reviews based on the best BERT model classification results.
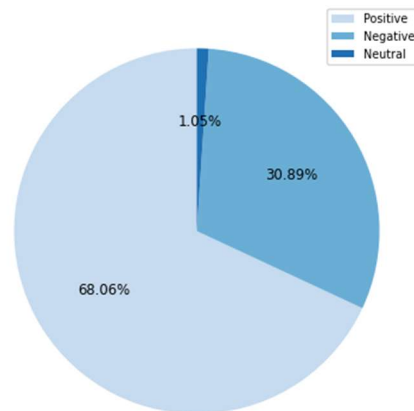
Fig. 19. Comparison of Sentiment Classes of BERT Classification Results

Based on Fig. 19, it can be seen that 68.06% of reviews fall into the positive class, 30.89% of reviews fall into the negative class and 1.05% of reviews fall into the neutral class. So it can be concluded that based on the sentiment analysis of the review data, 68.06% of the total patients who commented were satisfied with the services at the hospital, then 30.89% of patients who commented were still not satisfied with the services provided. While 1.05% of patients who commented gave a neutral rating to the service. Next, a word cloud visualization will be made using the prediction results with BERT. Figure 20 is a word cloud regarding negative sentiments.

Fig. 20. Word Cloud of Negative Sentiment from BERT Results

Fig. 21. Word Cloud of Positive Sentiment from BERT Results

Based on Fig. 20, it can be seen that the frequency of words that often appear are the words 'tidak', 'dokter', 'perawat', 'pelayanan', 'nunggu', 'lama', 'petugas', and others. The words that often appear show that patients who seek treatment at the hospital mostly complain about things related to long services, long queues and unfriendly officers. Next, Fig. 21 shows the word cloud for positive sentiment. Based on Fig. 21, it can be seen that the frequency of words that often appear in actual data and predicted data for positive sentiment is not much different. Both show frequent words such as 'pelayanan', 'baik', 'fasilitas', 'lengkap', 'ramah', 'nyaman', and others. The words that often appear show that despite the negative reviews given by patients, there are also many positive reviews that give praise and also gratitude regarding good service and also complete facilities and friendliness of officers. Next, Fig. 22 shows the word cloud for neutral sentiment.

**Fig. 22.** Word Cloud of Neutral Sentiment from BERT Results

Based on Fig. 22, it can be seen that most patients give balanced neutral reviews where patients write positive comments and negative comments as well. In addition, patients also write about the situation at the hospital such as crowded or quiet, ask about several things such as complaint numbers, surgery schedules, and provide constructive suggestions so that hospital can be better in the future.

4.5 P-Control Chart Based on Review

Monitoring of the classification result data using the BERT method is carried out using a *p*-control chart. Prediction results with negative sentiment classes can be said to be defects that indicate service defects in hospital. The following are the results of the *p*-control chart analysis based on review data in phases I and II.
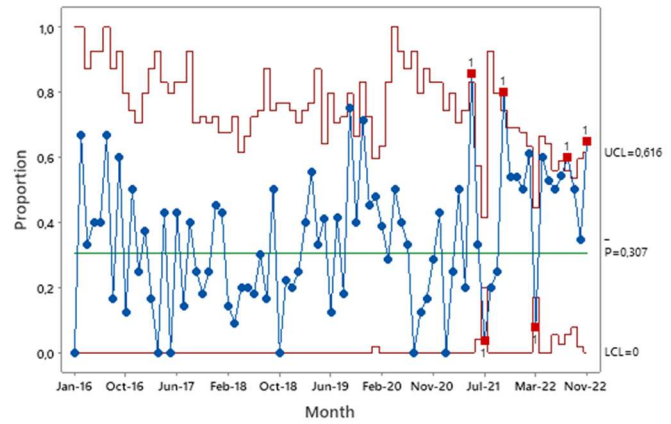
4.5.1. P-Control Chart in Phase I Based on Review

The phase I p-control chart based on review data was created using patient review data related to services at the hospital taken from Google Maps on January 1, 2016 to December 31, 2022 which had been classified using the BERT method. This data is used to obtain the average defect proportion value that will be used to monitor patient review data in phase II. Fig. 23 shows the phase I *p*-control chart with 3 sigma limit that can be used to determine whether the service in the hospital has been statistically controlled or not.
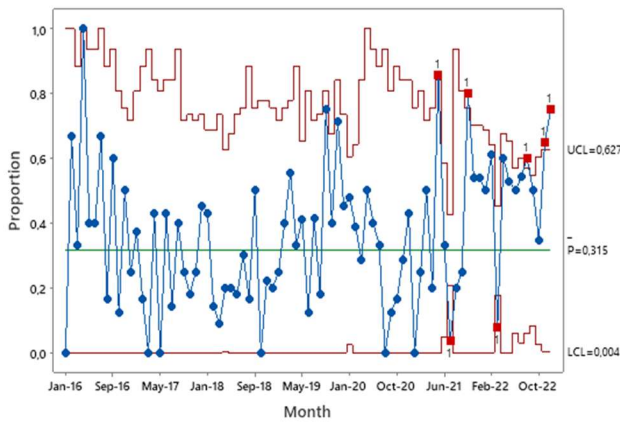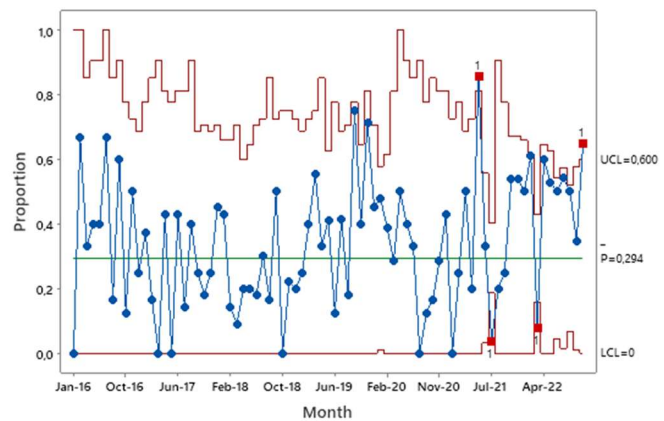
Fig. 23. Phase I P-Control Chart Based on Reviews

Fig. 24. Phase I Iteration 1 P-Control Chart

Based on Fig. 23, it can be seen that the phase I *p*-control chart has an average value of the proportion of defects of 0.315 with different upper control limits and lower control limits for each subgroup. In this control chart, there are 7 observations that are outside the control limits with details of 5 observations outside the upper control limit and 2 observations outside the lower control limit. However, in this study, observations that are outside the lower control limit are considered as in control observations. In addition, there is 1 observation that has a defect proportion value of 1 where the observation is considered an observation that is outside the upper control limit so that it needs to be deleted. Thus, deletion will be carried out on observations with a defect proportion value of 1 and deletion one by one on observations outside the farthest control limit from the UCL so that all observations can be statistically controlled. Fig. 24 shows the phase I iteration 1 *p*-control chart after the first deletion. Based on Fig. 24, it can be seen that the phase I iteration 1 *p*-control chart has an average value of the proportion of defects of 0.307. In this control chart, there are 4 observations that are outside the upper control limit or UCL. Thus, deletion will be carried out again on observations outside the farthest upper control limit so that all observations can be statistically controlled. Fig. 25 shows the phase I iteration 2 *p*-control chart after the second deletion.
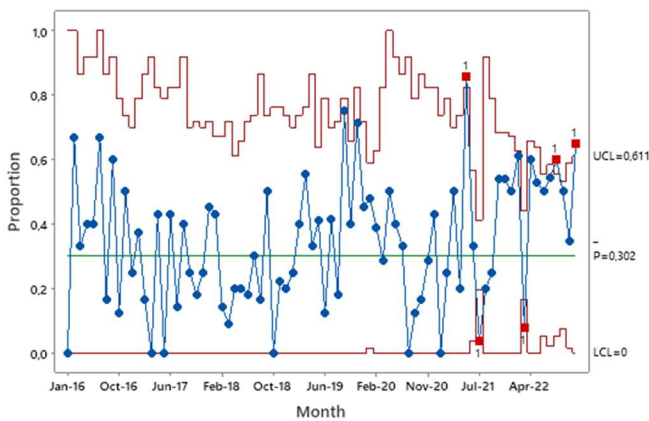
Fig. 25. Phase I Iteration 2 P-Control Chart

Fig. 26. Phase I Iteration 3 P-Control Chart

Based on Fig. 25, it can be seen that the phase I *p*-control chart iteration 2 has an average value of the proportion of defects of 0.302. In this control chart, there are 3 observations that are outside the upper control limit or UCL. Thus, deletion will be carried out again on observations outside the farthest upper control limit so that all observations can be statistically controlled. Fig. 26 shows the pha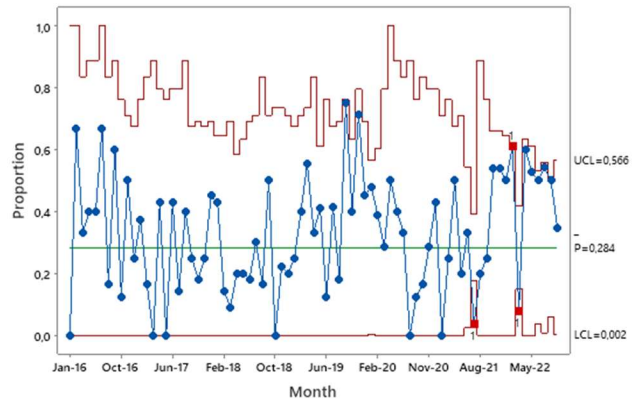se I iteration 3 *p*-control chart after the third deletion. Based on Fig. 26, it can be seen that the phase I *p*-control chart iteration 3 has an average value of the proportion of defects of 0.294. In this control chart, there are 2 observations that are

outside the upper control limit or UCL. Thus, deletion will be carried out again on the observation outside the control limit so that all observations can be statistically controlled. Fig. 27 shows the phase I iteration 4 *p*-control chart after the fourth deletion.
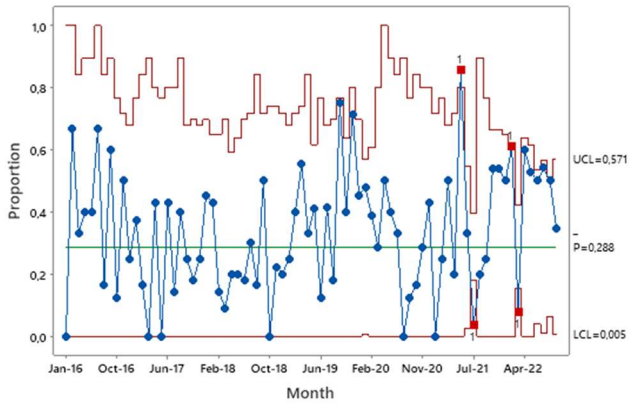
Fig. 27. Phase I Iteration 4 P-Control Chart

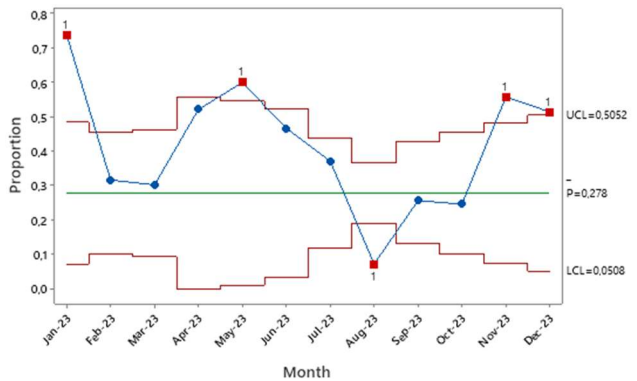
Fig. 28. Phase I Iteration 5 P-Control Chart

Based on Fig. 27, it can be seen that the phase I *p*-control chart iteration 4 has an average value of the proportion of defects of 0.288. In this control chart, there are 2 observations that are outside the upper control limit or UCL. Thus, deletion will be carried out again on the observation outside the control limit so that all observations can be statistically controlled. Fig. 28 shows the phase I iteration 5 *p*-control chart after the fifth deletion. Based on Figu. 28, it can be seen that the phase I *p*-control chart iteration 5 has an average value of the proportion of defects of 0.284. In this control chart, there is 1 observation that is outside the upper control limit or UCL. Thus, deletion will be carried out again on the observation outside the control limit so that all observations can be statistically controlled. Fig. 29 shows the phase I iteration 6 *p*-control chart after the sixth deletion.
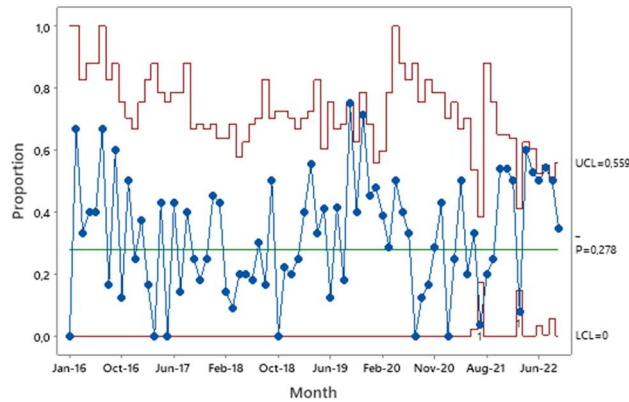
Fig. 29. Phase I Iteration 6 P-Control Chart

Fig. 30. Phase II P-Control Chart

Based on Fig. 29, it can be seen that the phase I *p*-control chart iteration 6 has an average value of the proportion of defects of 0.278 and there are no observations that are outside the upper control limit or UCL. So it can be concluded that the *p*-control chart is statistically controlled. The average proportion of defects on the phase I iteration 6 *p*-control chart will be used to monitor hospital service review data in phase II.

4.5.2.   P-Control Chart in Phase II Based on Review

The phase II I *p*-control chart was created based on patient review data related to hospital services on Google Maps from January 1, 2023 to December 26, 2023. The phase II *p*-control chart was made using the average proportion of defects obtained in the phase I *p*-control chart analysis which has been statistically controlled, which is 0.278. The calculation of the proportion of defects, LCL, and UCL values for each monthly observation is done in the same way as the calculation of the proportion of defects, LCL, and UCL values on the phase I *p*-control chart. Figure 30 shows the phase II *p*-control chart. Based on Fig. 30, it can be seen that there are still several out of control observations that make the phase II p control diagram not yet statistically controlled. Table 8

is a summary of the main causes of out of control, namely the 3 categories of causes of out of control with the highest percentage value for each month.

Table 8
Summary of Out of Control (OOC) Observations Based on Phase II P-Control Chart

No	Month	Causes of Out of Control (OOC) Observations	Percentage Causes of Out of Control (OOC) Observations
1	January 2023	Category 1	35.29%
		Category 2	17.64%
		Category 6	14.70%
2	February 2023	-	-
3	March 2023	-	-
4	April 2023	-	-
5	May 2023	Category 1	26.67%
		Category 6	23.33%
		Category 2	16.67%
6	June 2023	-	-
7	July 2023	-	-
8	August 2023	Category 1	29.54%
		Category 2	22.72%
		Category 6	22.72%
9	September 2023	-	-
10	October 2023	-	-
11	November 2023	Category 1	30.61%
		Category 6	20.40%
		Category 5	12.24%
12	December 2023	Category 1	24.39%
		Category 6	19.51%
		Category 2	14.63%

Based on Table 8, there are the most categories of causes of OOC observations and their percentages where the percentage calculation is obtained from the number of each category of causes of OOC observations divided by the total categories of causes of OOC observations. The majority of the causes of OOC observations are category 1, category 2 and category 6. In January 2023, the percentage of causes of OOC observations in category 1 reached the highest value with a percentage of 35.29%. In August 2023, the percentage of causes of category 2 OOC observations reached the highest value with a percentage of 22.72%. And in May 2023, the percentage of causes of category 6 OOC observations reached the highest value with a percentage of 23.33%.

4.6   Service Improvement Recommendations

Based on the results of the analysis in the previous section, the factors causing patient dissatisfaction with hospital services can be found. Some recommendations can be given as follows.

Recommendation 1: **Optimize Patient Flow and Resource Allocation**

- Increase staffing: Augment the number of service personnel (e.g., receptionists, medical assistants) to reduce wait times.
- Enhance online registration: Refine the online registration system to better manage patient inflow and minimize queues.

Recommendation 2: **Foster a Patient-Centered Culture**

- Implement comprehensive training: Provide staff with training in hospitality and patient-centered care to enhance their interpersonal skills.
- Evaluate and adjust training: Regularly assess the effectiveness of training programs and make necessary modifications.

Recommendation 3: **Ensure Adequate Staffing Levels**

- Conduct staffing needs assessments: Regularly evaluate staffing levels against established standards to identify shortages.
- Propose staffing increases: Submit proposals to the relevant government to increase the number of healthcare workers.

Recommendation 4: **Prioritize Facility Maintenance and Hygiene**

- Conduct regular inspections: Implement routine inspections of facilities and equipment to identify and address maintenance issues.
- Enhance cleaning protocols: Strengthen cleaning procedures to maintain a clean and sanitary environment.

Recommendation 5: **Streamline Insurance Verification and Service Delivery**

- Verify insurance information promptly: Ensure accurate and timely verification of patient insurance coverage.
- Align services with insurance benefits: Coordinate service delivery with patient insurance plans to avoid discrepancies.

Recommendation 6: **Improve Registration Processes**

- Integrate online and offline registration: Develop a seamless integration between online and offline registration systems to avoid inconsistencies.
- Address queue management issues: Implement measures to prevent queue manipulation and ensure fair access to services.

Recommendation 7: **Enhance Operational Efficiency**

- Improve customer service responsiveness: Enhance the responsiveness of customer service channels (e.g., phone, email).
- Streamline operational processes: Regularly review and optimize operational procedures to minimize inefficiencies.

Recommendation 8: **Improve Emergency Room Response Times**

Increase emergency room staffing: Allocate additional healthcare personnel to the emergency room to reduce wait times.

Recommendation 9: **Expand Parking Facilities**

Increase parking capacity: Expand the available parking space to accommodate the growing number of patients.

Recommendation 10: **Optimize Referral Processes**

Streamline referral procedures: Simplify and expedite the referral process to minimize delays for patients requiring specialized care.

5. Conclusions

This study investigates the quality of hospital services based on patient reviews on Google Maps, employing a hybrid methodology that combines p-control charts and Bidirectional Encoder Representations from Transformers (BERT). Sentiment analysis using BERT revealed a positive sentiment bias among reviews, with 68.06% classified as positive, 30.89% as negative, and 1.05% as neutral. This suggests a preponderance of positive feedback. The classification accuracy, as measured by the AUC value, was exemplary for both training (99.95%) and testing (93.72%) datasets. P-control chart analysis of patient reviews indicated a statistically controlled process in phase II, suggesting consistent quality. The primary challenges encountered by patients were extended wait times and service durations. To mitigate these issues, increasing the number of service providers and human resources, along with enhancing the online registration system, are recommended. Future research could benefit from a larger dataset, particularly for subgroups, to yield more representative and less sensitive results. Employing alternative attribute control charts, such as those differentiating between critical, major, and minor defects, could provide more nuanced insights. Additionally, exploring other classification methods beyond BERT might lead to improved accuracy in sentiment analysis.

References

Alammar, J. (2018). *The Illustrated Transformer*. Github. https://jalammar.github.io/illustrated-transformer/

Alaparthi, S., & Mishra, M. (2021). BERT: A sentiment analysis odyssey. *Journal of Marketing Analytics*, *9*(2), 118–126.

AlBadani, B., Shi, R., & Dong, J. (2022). A novel machine learning approach for sentiment analysis on Twitter incorporating the universal language model fine-tuning and SVM. *Applied System Innovation*, *5*(1), 13.

Allibhai, J. (2018). *Hold-out vs. Cross-validation in Machine Learning*. Medium. https://medium.com/@jaz1/holdout-vs-cross-validation-in-machine-learning-7637112d3f8f

Bekkar, M., Djemaa, H. K., & Alitouche, T. A. (2013). Evaluation Measures for Models Assessment over Imbalanced Data Sets. *Journal of Information Engineering and Applications*, *3*(10), 27–38.

Bello, A., Ng, S.-C., & Leung, M.-F. (2023). A BERT framework to sentiment analysis of tweets. *Sensors*, *23*(1), 506.

Borg, A., & Boldt, M. (2020). Using VADER sentiment and SVM for predicting customer response sentiment. *Expert Systems with Applications*, *162*, 113746. https://doi.org/https://doi.org/10.1016/j.eswa.2020.113746

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, *1*(Mlm), 4171–4186.

Elfaik, H., & Nfaoui, E. H. (2020). Deep bidirectional LSTM network learning-based sentiment analysis for Arabic text. *Journal of Intelligent Systems*, *30*(1), 395–412.

Feldman, R., & Sanger, J. (2006). The Text Mining Handbook:Advanced Approaches in Analyzing Unstructured Data. In *The Text Mining Handbook*. Cambridge University Press. https://doi.org/10.1017/cbo9780511546914

Firmansyah, D., & Ahsan, M. (2023). Monitoring Kualitas Pada Aplikasi MyPertamina Berdasarkan Rating Pengguna di Google Play Menggunakan Diagram Kendali p. *Jurnal Sains Dan Seni ITS*, *12*(2), D158–D163.

Han, J., Kamber, M., & Pei, J. (2012). Data Mining: Concepts and Techniques. In *Data Mining: Concepts and Techniques* (3rd ed.). Elsevier Inc. https://doi.org/10.1016/C2009-0-61819-5

Huang, X., Zhang, W., Tang, X., Zhang, M., Surbiryala, J., Iosifidis, V., Liu, Z., & Zhang, J. (2021). Lstm based sentiment analysis for cryptocurrency prediction. *Database Systems for Advanced Applications: 26th International Conference, DASFAA 2021, Taipei, Taiwan, April 11–14, 2021, Proceedings, Part III 26*, 617–621.

Jin, Z., Yang, Y., & Liu, Y. (2020). Stock closing price prediction based on sentiment analysis and LSTM. *Neural Computing and Applications*, *32*, 9713–9729.

Khan, A., & Baharudin, B. (2011). Sentiment classification using sentence-level semantic orientation of opinion terms from blogs. *2011 National Postgraduate Conference - Energy and Sustainability: Exploring the Innovative Minds, NPC 2011*, 1–7. https://doi.org/10.1109/NatPC.2011.6136319

Kokab, S. T., Asghar, S., & Naz, S. (2022). Transformer-based deep learning models for the sentiment analysis of social media data. *Array*, *14*, 100157.

Li, Z., Li, R., & Jin, G. (2020). Sentiment analysis of danmaku videos based on naïve bayes and sentiment dictionary. *Ieee Access*, *8*, 75073–75084.

Michel, S. (2001). Analyzing service failures and recoveries: A process approach. *International Journal of Service Industry Management*, *12*(1), 20–33. https://doi.org/10.1108/09564230110382754

Montgomery, D. (2013). *Introduction to Statistical Quality Control Seventh Edition* (Issue 112). John Wiley & Sons Inc.

Nandwani, P., & Verma, R. (2021). A review on sentiment analysis and emotion detection from text. *Social Network Analysis and Mining*, *11*(1), 81. https://doi.org/10.1007/s13278-021-00776-6

Obiedat, R., Qaddoura, R., Al-Zoubi, A. M., Al-Qaisi, L., Harfoushi, O., Alrefai, M., & Faris, H. (2022). Sentiment Analysis of Customers' Reviews Using a Hybrid Evolutionary SVM-Based Approach in an Imbalanced Data Distribution. *IEEE Access*, *10*, 22260–22273. https://doi.org/10.1109/ACCESS.2022.3149482

Pota, M., Ventura, M., Catelli, R., & Esposito, M. (2020). An effective BERT-based pipeline for Twitter sentiment analysis: A case study in Italian. *Sensors*, *21*(1), 133.

Pribadi, N. A., & Ahsan, M. (2023). Monitoring Quality of KAI Access Application Based on Customer Reviews on Google Play Store Using Laney p' Control Chart Based on Convolutional Neural Network. *Proceedings of the 5th International Conference on Statistics, Mathematics, Teaching, and Research 2023 (ICSMTR 2023)*, 175–184. https://doi.org/10.2991/978-94-6463-332-0_20

Pristiyono, Ritonga, M., Ihsan, M. A. Al, Anjar, A., & Rambe, F. H. (2021). Sentiment analysis of COVID-19 vaccine in Indonesia using Naïve Bayes Algorithm. *IOP Conference Series: Materials Science and Engineering*, *1088*(1), 012045.

Pyon, C. U., Woo, J. Y., & Park, S. C. (2011). Service improvement by business process management using customer complaints in financial service industry. *Expert Systems with Applications*, *38*(4), 3267–3279. https://doi.org/10.1016/j.eswa.2010.08.112

Rasouli, O., & Zarei, M. H. (2016). Monitoring and reducing patient dissatisfaction: a case study of an Iranian public hospital. *Total Quality Management and Business Excellence*, *27*(5–6), 531–559. https://doi.org/10.1080/14783363.2015.1016869

Singh, M., Jakhar, A. K., & Pandey, S. (2021). Sentiment analysis on the impact of coronavirus in social life using the BERT model. *Social Network Analysis and Mining*, *11*(1), 33.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *2017-December*(Nips), 5999–6009.

Villavicencio, C., Macrohon, J. J., Inbaraj, X. A., Jeng, J.-H., & Hsieh, J.-G. (2021). Twitter sentiment analysis towards covid-19 vaccines in the Philippines using naïve bayes. *Information*, *12*(5), 204.

Wankhade, M., Rao, A. C. S., & Kulkarni, C. (2022). A survey on sentiment analysis methods, applications, and challenges. *Artificial Intelligence Review*, *55*(7), 5731–5780. https://doi.org/10.1007/s10462-022-10144-1

Wilie, B., Vincentio, K., Winata, G. I., Cahyawijaya, S., Li, X., Lim, Z. Y., Soleman, S., Mahendra, R., Fung, P., Bahar, S., & Purwarianti, A. (2020). *IndoNLU: Benchmark and Resources for Evaluating Indonesian Natural Language Understanding*.

Xu, X., & Li, Y. (2016). The antecedents of customer satisfaction and dissatisfaction toward various types of hotels: A text mining approach. *International Journal of Hospitality Management*, *55*, 57–69. https://doi.org/10.1016/j.ijhm.2016.03.003