

How does natural language processing identify the issues?

Priati Assiroj¹, Sirojul Alam², Harco Leslie Hendric Spits Warnars³

¹Department of Technology Management, Immigration Polytechnic, Depok, Indonesia

²Department of Cyber Defense Engineering, Faculty of Sciences and Defense Technology, The Republic of Indonesia Defense University, Bogor, Indonesia

³Department of Computer Science, Binus Graduate Program, Doctor of Computer Science, Bina Nusantara University, Jakarta, Indonesia

Article Info

Article history:

Received Feb 13, 2024

Revised May 13, 2024

Accepted Jun 5, 2024

Keywords:

Application

Natural language processing

Review

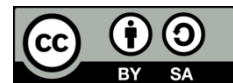
Service

Topic modeling

ABSTRACT

Product innovation and service improvement have become essential or crucial for organisations, including public service organisations. The Indonesia Immigration Directorate released the m-passport application to enhance its quality of service. The m-passport application is considered good as it has been downloaded over a million times. Like immigration officers, this application seems to be at the forefront, reflecting an increasingly better service. However, there was still a need for significant improvement in the application. Improvements can be made to the application by considering user feedback or reviews. Reviews provided by users, approximately 12K, will serve as input for improving or enhancing the application. This was made possible as users interact directly with the application. The most common issues are one-time password or OTP verification code with a probability value of 0.044, errors when logging in with a probability value of 0.283, and slow response applications with a probability value of 0.125.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Sirojul Alam

Department of Cyber Defense Engineering, Faculty of Sciences and Defense Technology

The Republic of Indonesia Defense University

Bogor, Indonesia

Email: sirojul.alam@tp.idu.ac.id

1. INTRODUCTION

M-passport is a mobile application from the Directorate General of Immigration, Indonesia, which allows users to apply for a new passport or renew an expired visa. By allowing users to submit their data and upload the requirements anytime, anywhere, m-passport streamlines the application process. All the immigration offices of Indonesia can be accessed from this application. These featured services are available everywhere, even if users are in their homes; users do not have to waste their time. With one account, users can submit multiple passport applications together; users can apply in all immigration offices in Indonesia, quickly pay, and self-paced scheduling [1].

Excess data has been collected as part of human engagement in cyberspace. This data is unstructured, such as video, images, and text. The massive volume of data, which we call big data, contains valuable things, and we must obtain pertinent information. The challenge is about data processing and interpretation [2]. One example of information that may be gleaned from the text data sets is the propensity of mobile application users. This data comes from the reviews users have left for an application they use. These reviews show how satisfied they were and which issues had the most impact on their satisfaction.

Becoming a small field of artificial intelligence, natural language processing (NLP) focuses on enhancing a computer's ability to understand, analyse, and interpret human language content meaningfully [3]. It evolved along with the increase of computation and the rise of deep learning [4] in which our computer can

understand and process human language automatically, such as in performing text summarisation, topic modelling or segmentation, translation, voice recognition, and named entity recognition [3], identifying sentiment and emotion toward products or services, speech-to-speech translation, and social media mining [5].

One part of NLP is topic modelling [6]. Topic modelling is a technique in text analysis aimed at identifying and extracting main topics that are latent within text data [7]. Topic modelling usually utilises machine learning techniques such as latent dirichlet allocation (LDA), which is well-known for identifying hidden themes within documents [7]. Non-negative matrix factorization (NMF) is recognised as a matrix factorisation method [8], as well as latent semantic analysis (LSA), a statistical technique for obtaining topics within a collection of texts [9], [10]. Vayansky and Kumar [11] reviewed various topic modelling approaches capable of handling topic correlation, change over time, and short texts, aiming to encourage diversity in topic modelling methods based on user needs. BERTopic was introduced to extend topic modelling as a clustering task by extracting coherent topic representations through a class-based variation of term frequency-inverse document frequency (TF-IDF) [12].

Topic modelling offers several advantages, including rapid analysis of unstructured textual data, improved understanding of data across various formats, and enhanced customer experiences through automated categorisation and routing of support requests [13]. However, it should be noted that topic modelling does not look like keyword extraction, which aims to choose individual terms rather than broader themes. In addition, recent advances in deep learning have led to the development of hybrid models that combine topic modelling with a pre-trained language model, such as bidirectional encoder representation from transformers (BERT) or utilising uniform manifold approximation and projection (UMAP) and hierarchical DBSCAN (HDBSCAN) for clustering based on semantic similarity [14].

User or customer satisfaction is a critical aspect of organisational operation [15], [16]. Customer satisfaction measures whether customers are happy or unhappy with a company's products, services, or capabilities. It indicates the company's achievement in fulfilling customers' demands, which we can obtain through reviews, ratings, or surveys [17]. Customer satisfaction is inversely proportional to customer complaints. Customer complaints are an important and valuable source of information that can be used to recognise problems and decide whether to improve products or services. Complaint management is essential in maintaining customer satisfaction and loyalty and addressing organisational weaknesses.

The topic has many facets and has been investigated in many fields, for example in information retrieval in user experience. It is a crucial process in software maintenance and gives valuable things from user feedback [18]. This is particularly important in mobile applications, such as in change requests and user sentiment [19]. For example, in a movie mobile application, the user interface and additional features play a crucial role in user experience.

2. METHOD

In our research, we leverage topic modelling within NLP to pinpoint issues within the m-passport Android mobile application, as expressed in user reviews on the Google PlayStore. By conducting this study, we aim to identify the primary source of the problem and subsequently propose enhancements to ensure superior application services. Implementing topic modelling typically involves preprocessing stages such as tokenisation, removal of stopwords, feature selection, and transformation. LDA is a crucial technique utilised in this context, offering a flexible framework for discerning underlying topics within a collection of documents and elucidating how these topics are reflected in word distribution [7]. LDA's versatility extends its applicability to various domains, including document modelling, text classification, collaborative filtering, and recommendation systems, providing insights into document latent structures. Additionally, the embedded topic model (ETM) introduced in [7] combining traditional topic modelling with word embeddings enhances model quality and performance. Non-negative matrix factorisation (NMF), as demonstrated in [8], proves effective in decomposing the term-document matrix into interpretable factor matrices, enforcing non-negativity constraints, and facilitating topic extraction. NMF is particularly valuable for tasks such as document clustering, offering automated topic identification, improved semantic representation, and enhanced clustering outcomes.

Moreover, latent semantic indexing (LSI), as discussed in [10], operates effectively on Hindi text data by generating topics represented by word lists, interpreting these topics based on identified words, and visualising the results for enhanced comprehension. We opt for the sentence embedding technique in our work due to its proven quality and performance, as mentioned in [7]. This research initiative is initiated through steps outlined in Figure 1, encompassing data preprocessing and topic modelling, and results in visualisation to effectively analyse user reviews and extract actionable insights for improving the m-passport application.

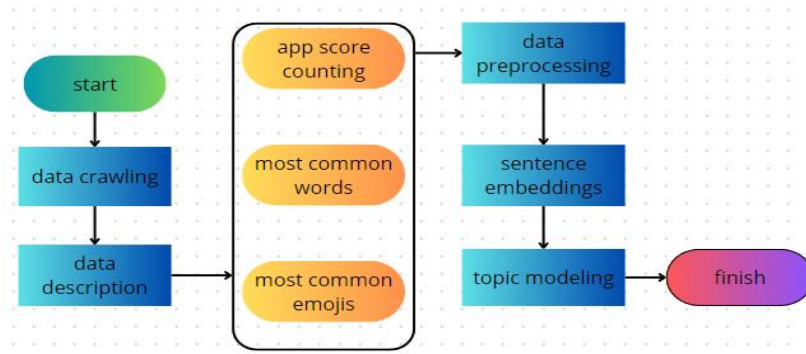


Figure 1. Research flows

This research commenced with data crawling, also known as web crawling, which involves systematically browsing the internet to collect data and information. This collected data can be applied for various purposes, such as web data mining, focused crawling, or social media analysis. A notable example of utilising this concept is found in the work of [20], who conducted social media analysis on Facebook. In our research, we leverage this data crawling methodology to gather user reviews of the m-passport mobile application for subsequent analysis, mainly focusing on topic modelling. We amassed 12,504 reviews of the m-passport mobile application from Google PlayStore from its initial release until the latest version. An analysis of the collected data revealed that a significant portion of the reviews is negative. This experiment was conducted using Google Colaboratory, and the function responsible for data crawling is depicted in Figure 2.

Figure 2 demonstrates the practical application of Google Colaboratory in the context of data scraping. In this endeavour, we relied on the widely used ‘google_play_scraper’ library, recognised as a standard tool for extracting data from Google PlayStore. Specifically, our focus was on harvesting user reviews associated with the application ‘id.go.immigrate.paspor_online’, which corresponds to the official identity of the m-passport mobile application. Our primary objective was to compile reviews composed in Bahasa Indonesia from users residing in Indonesia. This strategic approach ensured that the collected data aligned closely with the application's user base's target demographic and linguistic preferences. Following the data acquisition phase, we organised the reviews chronologically, sorting them from the most recently posted to the earliest entries. This sorting mechanism facilitates efficient analysis of user sentiments and trends over time, providing valuable insights into the evolving perceptions and experiences of the m-passport application within the Indonesian user community. An example of the data we have obtained can be observed in Figure 3.

Figure 3 provides a subset of our dataset, showcasing the five most recent and five oldest entries out of a total of 12,504 records. The dataset consists of user reviews about the m-passport mobile application, sourced from Google PlayStore. Notably, the most recent review in our dataset was posted on January 7, 2024, at 00.46 AM, while the oldest review dates back to December 30, 2021, at 03.49 AM. Upon analysing this sample of ten entries, it becomes evident that the majority, specifically 7 out of 10 reviews, express negative sentiments, highlighting various issues or concerns encountered by users. Conversely, the remaining three reviews convey positive feedback, indicating satisfactory user experiences with the m-passport application. Analysing this dataset enables us to understand user feedback dynamics comprehensively, allowing for informed decision-making and continuous improvement efforts.

```

[ ] from google_play_scraper import Sort, reviews_all

ulasan = reviews_all(
    'id.go.imigrasi.paspor_online',
    sleep_milliseconds = 0,
    lang = 'id',
    country = 'id',
    sort = Sort.NEWEST,
    count = 1000,
    filter_score_with = None,
)
    
```

Figure 2. Data crawling function

	userName	score	at	content
0	Toni17 Galung17	5	2024-01-07 00:46:26	Ok
1	April Diana	1	2024-01-06 15:38:13	Apps berat. Loadingnya lama
2	M3LL0W ./	1	2024-01-06 12:34:21	Payah
3	rafida insani	1	2024-01-06 10:04:48	Aplikasi apa ini., sdh masukan data, tp gak bi...
4	Willy Sazmita	1	2024-01-06 05:37:58	Tidak bisa memilih tanggal kedatangan. Sudah c...
...
12499	Sukirno Nur	5	2022-01-04 02:08:13	Mantabbbb
12500	Ifan Sastra Hirata	2	2021-12-31 12:07:17	Tidak ditemukan Kanim di lokasi anda
12501	Handriyanti	1	2021-12-31 08:32:04	Lemot bgt pas milih lokasi pengambilan paspor ...
12502	Ibnu Prayudha Pangestu	5	2021-12-31 02:59:30	Aplikasi ini sangat membantu bagi saya yang me...
12503	HAKIM	2	2021-12-30 03:49:33	Masih trial ya... Di banyuwangi sudah ada kant...

12504 rows x 5 columns

Figure 3. Data sample

The next step involves describing the data further. In this phase, we leverage various Python libraries to analyse the dataset comprehensively. Specifically, we aim to identify the most common words utilised by users in their reviews, ascertain the prevalent emojis employed to convey sentiments and calculate the overall application score rating [21]. To facilitate this analysis, we have developed a set of functions designed to tally the frequency of commonly used words in the reviews and to enumerate the occurrence of emojis embedded within the reviews. These functions are presented in Figure 4. By employing these functions, we can gain valuable insights into the linguistic patterns and emotional expressions prevalent among users and derive quantitative metrics to evaluate user satisfaction.

Figure 4 displays several libraries utilised in our study, including 'emoji', 're', and 'collection'. These libraries support our workflow by enabling the extraction of words from text and the extraction of words from reviews. After extracting emojis and phrases from the reviews, we tally the most frequently occurring words and emojis. Through this process, we have identified words such as 'aplikasi' (application), 'bisa' (can), 'tidak' (not), 'mau' (want), 'daftar' (registration), and 'ada' (there) as the most commonly used words. Additionally, various types of emojis have been identified, including 'thumbs down', 'roll and laugh', 'pouting face', 'folded hands', 'beaming face', 'pile of poo', and others. Furthermore, we calculate the number of application ratings ranging from one star to five stars. Subsequently, the next step in our analysis involves data preprocessing. We perform various text preprocessing procedures in this step using the NLTK Python library. These procedures include text cleaning, case-folding, tokenisation, filtering, and stemming. These are standard procedures in text preprocessing essential for text analysis. Text cleaning removes unused characters such as hashtags, links, numbers, and special characters from the text data. The function responsible for text cleaning is depicted in Figure 5.

```
!pip install emoji
import re
from collections import Counter
import emoji

# Function to extract emojis from text
def extract_emojis(s):
    return ''.join(c for c in s if c in emoji.EMOJI_DATA)

# Preprocess text to extract words and emojis
mpaspor['words'] = mpaspor['content'].apply(lambda x: re.findall(r'\b\w+\b', x.lower()))
mpaspor['emojis'] = mpaspor['content'].apply(extract_emojis)

# Aggregate all words and emojis
all_words = sum(mpaspor['words'], [])
all_emojis = ''.join(mpaspor['emojis'])

# Counting word and emoji frequencies
word_freq = Counter(all_words)
emoji_freq = Counter(all_emojis)
```

Figure 4. Counting words and emojis

```
[ ] def cleaningText(text):
    text = re.sub(r"b'", '', text) #deleting b'
    text = text.replace('\n', '') #deleting new line
    text = re.sub(r"#[A-Za-z0-9]+", '', text) # deleting '#' hashtag
    text = re.sub(r"http\S+", '', text) # deleting link
    text = re.sub(r'[0-9]+', '', text) # deleting number
    text = text.translate(str.maketrans('', '', string.punctuation))
    text = text.strip(' ')
    return text
```

Figure 5. Function for text cleaning

In Figure 5, we observe that the text undergoes cleaning to remove unnecessary characters such as hashtags and special characters. Text cleaning is a crucial step in preparing raw data for analysis. It aims to transform the text into a format suitable for subsequent analyses or applications, thereby enhancing the results' quality, accuracy, and reliability. Text case-folding, or text normalisation or standardisation, involves converting text into a consistent format. This process typically entails transforming text data into either lowercase or uppercase. Our research converts the text into lowercase for consistency, efficiency, and vocabulary size reduction. Lowercasing collapses different case variations of the same word into a single representation, streamlining subsequent text-processing tasks. We have already developed functions for text case-folding, tokenisation, and text filtering, as illustrated in Figure 6. These functions are essential components of the text preprocessing pipeline and contribute to ensuring the consistency and quality of the processed text data.

```
#Case Folding
def casefoldingText(text):
    text = text.lower()
    return text

#Tokenizing
def tokenizingText(text):
    text = word_tokenize(text)
    return text

#Filtering
def filteringText(text):
    listStopwords = set(stopwords.words('indonesian'))
    filtered = []
    for txt in text:
        if txt not in listStopwords:
            filtered.append(txt)
    text = filtered
    return text
```

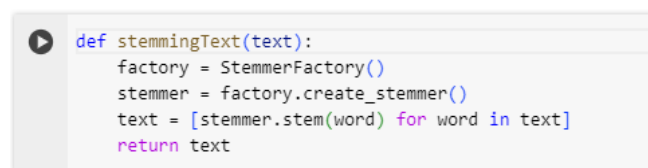
Figure 6. Function for case folding, tokenizing, and filtering

Figure 6 illustrates three text-processing functions utilised in our research. Text tokenisation is splitting every sentence into individual words, referred to as tokens. These tokens can encompass words, phrases, sentences, or even characters, depending on the application's specific requirements. Text filtering involves selecting or retaining data based on specific criteria. In our study, the requirement for filtering is the Indonesian language. The objective is to enhance the quality of the data and render it more suitable for analysis. Stemming reduces inflected forms of words to their root or base form, essentially transforming words into their basic form [22]. This process can be particularly challenging in languages with complex and ambiguous structures [23]. Various stemming techniques have been evaluated for their advantages and limitations, aiming to enhance text processing in machine learning and information retrieval. We have developed a function for performing stemming, as depicted in Figure 7.

Stemming plays a pivotal role in text processing and finds applications in various NLP tasks, including sentiment analysis and part-of-speech tagging. Reducing the dimensionality of text data can enhance the accuracy and precision of machine learning algorithms. It is an initial preprocessing step before undertaking other NLP tasks, contributing to improved performance. The subsequent step in our analysis involves sentence embeddings. Sentence embeddings represent a potent tool in NLP, utilised in sentiment analysis and question classification tasks [24], [25]. One approach involves semantic subspace analysis to

construct a sentence model that captures the semantic relationships between words [26]. Sentence embeddings convert linguistic information into a numerical representation, enabling computers to effectively comprehend and analyse natural language. This technique is invaluable for question categorisation, natural language inference, and sentiment analysis, representing sentences as vectors in high-dimensional spaces to facilitate analysis. These sentence embeddings are often derived from pre-trained language models, with methods such as dynamic interaction to enhance representation [24] and semantic subspace for improved performance and assist in managing incomplete phrases [26].

The application of sentence embeddings in topic modelling is significant, as they aid in identifying and clustering similar sentences or documents based on their semantic meanings. Sentence embeddings streamline the topic modelling process, facilitating the extraction of hidden topics from text documents. Topic modelling, a widely used technique in NLP, aims to uncover latent topics within text documents. It simplifies text data organisation by grouping common words and generating specified topics [27]. This method explores hidden semantics in text, enabling the identification of document topics and succinctly capturing document themes.



```
def stemmingText(text):
    factory = StemmerFactory()
    stemmer = factory.create_stemmer()
    text = [stemmer.stem(word) for word in text]
    return text
```

Figure 7. Function for stemming

3. RESULTS AND DISCUSSION

We have successfully compiled a dataset comprising reviews of the m-passport Android mobile application from December 30th, 2021, to January 7th, 2024, totalling 12,504 reviews. This dataset is structured into four columns: 'userName', 'score', 'at', and 'content' as outlined in Figure 3. The 'userName' column contains user identification data, while the 'score' column denotes user ratings assigned to the application. These ratings range from 1 to 5 stars, signifying varying degrees of satisfaction, with 1 star indicating an abysmal rating, 2 stars indicating a poor rating, 3 stars indicating a neutral rating, 4 stars representing a good rating, and 5 stars representing an exceptional rating. Our analysis focuses solely on the 'score' and 'content' columns. In examining the 'score' and 'content' columns, we aimed to describe the dataset comprehensively. We aimed to extract pertinent information relevant to our research, including user ratings, common words utilised in review, and frequently used emojis. We visualised the distribution of application ratings in Figure 8, facilitating a deeper understanding of user sentiments towards the m-passport over the specific period.

By analysing the data extracted from user reviews, we seek to gain insights into the user experience, identify prevalent issues, and explore opportunities for improvement. This comprehensive approach enables us to glean valuable insights that can inform decision-making processes to enhance overall quality and user satisfaction. The data presented in Figure 8 shows that the overall rating for the application is notably poor. User ratings, ranging from 1 to 5 stars, provide direct insight into their experiences and satisfaction. Specifically, the rating distribution indicates a significant prevalence of low ratings, with 1-star ratings comprising the majority at 9995, 2-star ratings at 772, 3-star ratings at 343, 4-star ratings at 245, and 5-star ratings at 1042. However, at this juncture, the exact reasons behind these low ratings remain unclear. Without a deeper understanding of the underlying factors influencing user ratings, it is challenging to ascertain the precise issues plaguing the application.

Moving forward, our focus shifts to identifying the most prevalent words frequently used in user reviews and the emojis commonly embedded within them. This approach is crucial for gaining insights into potential areas for improvement within the application and gauging user sentiment through emojis, as illustrated in Figures 8 and 9. By pinpointing the dominant words and emojis, we established reference points for implementing enhancements to the application and effectively measuring the sentiment.

Figure 9 illustrates the frequency of occurrence of the six most common words in reviews of the m-passport Android mobile application. The x-axis represents these words, while the y-axis denotes the frequency of each word within the corpus. The word 'aplikasi' (application) appears the most frequently, occurring 5,661 times in the corpus. Similarly, the word 'bisa' (can) is utilised 3,667 times, followed by 'tidak' (not) with 2,772 occurrences. Additionally, the word 'mau' (want) is used 1,995 times, while both

'daftar' (registration) and 'ada' (there) appear 1,718 and 1,445 times, respectively, within the corpus. This visualisation provides insights into the linguistic patterns prevalent in user reviews of the m-passport application. The prominence of certain words highlights recurring themes or topics discussed by users, shedding light on areas of interest or concern. For instance, the frequent occurrence of words such as 'aplikasi' and 'can' suggests discussions about the functionality and capabilities of the m-passport application. Conversely, words like 'tidak' may indicate dissatisfaction or issues encountered by users.

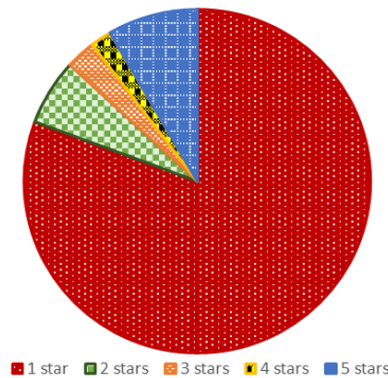


Figure 8. Application rating

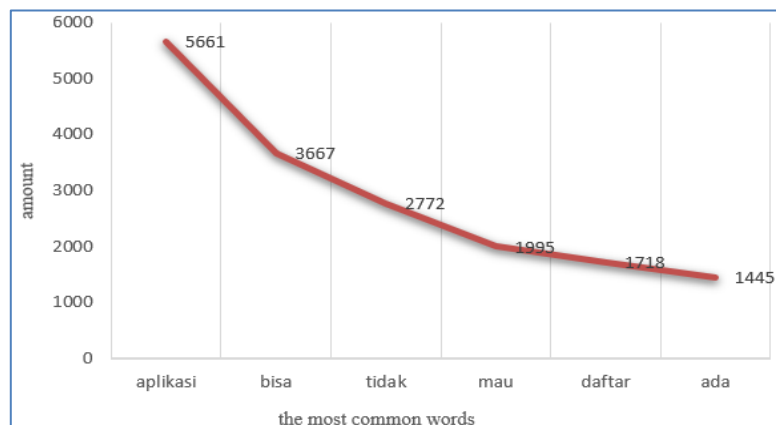


Figure 9. The most dominant words

By understanding the language users use and the topics they commonly discuss, developers and stakeholders can make informed decisions to enhance the user experience and address user concerns effectively. This data-driven approach to linguistic analysis facilitates targeted interventions to improve user satisfaction and optimise the overall performance of the m-passport application. We also examined the frequency of emojis users integrate in their reviews, as these emojis serve as indicators of users' emotional responses during their interactions with the m-passport application. Figure 10 depicts the most frequently utilised emojis. By scrutinising the prevalence of emojis, we attain a more profound comprehension of users' emotional states and sentiments regarding the application. Emojis function as visual representations of user experience, aiding developers in gauging user satisfaction, frustration, or other emotional responses this application elicits.

Figure 10 presents an overview of the most frequently utilised emojis within user reviews, offering valuable insights into the prevailing sentiment following interactions with the application. These emojis are ranked based on frequency, with the x-axis representing the emojis types and the y-axis indicating their respective counts. Notably, the 'thumb-down' emojis emerge as the predominant symbol, occurring 459 times, indicative of widespread user dissatisfaction with the application's performance. Conversely, the count of 'thumb-up' emojis is notably lower, totalling only 91 instances, suggesting a markedly lower level of user satisfaction, with only approximately 20% of users expressing contentment. A diverse array of other emojis,

conveying emotions ranging from frustration to sadness, underscores the multifaceted nature of user experience and sentiments. Following data collection, we proceeded with text processing procedures, which included text cleaning, case-folding, tokenisation, filtering, and stemming, as delineated earlier. These preprocessing steps were instrumental in preparing the text data for subsequent analysis, namely sentence embeddings and topic modelling. By employing these techniques, we could effectively preprocess the text data, making it suitable for further study and interpretation.

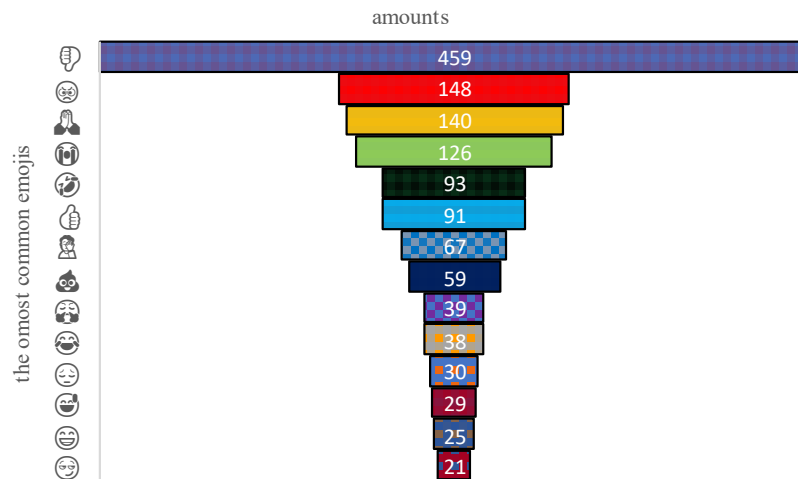


Figure 10. The most common emojis

We conducted topic modelling to ascertain the key aspects that need to be addressed to enhance the services provided through the m-passport application. Utilising the BERTopic model, we visualised its distribution to gain insights into its structure. Figure 11 depicts the distribution of the 14 topics generated by the model. We aimed to uncover underlying themes and issues within the user review through topic modelling, which can inform strategies for improving the m-passport application. Advanced techniques such as BERTopic allow us to extract meaningful patterns and topics from the text data.

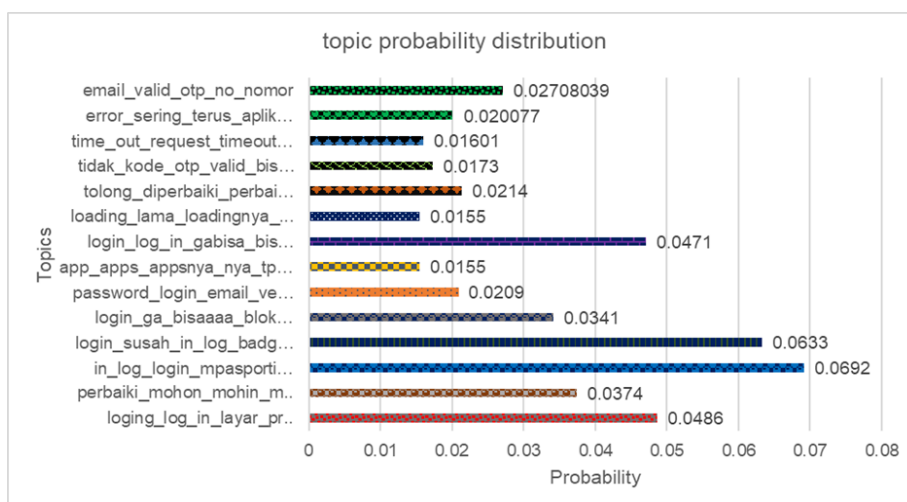


Figure 11. Topic distribution

Figure 11 provides insights into the most prevalent topics derived from review data, offering valuable guidance for improving the m-passport application’s performance and user experience. Each of the 14 identified topics is assigned a score reflecting its prominence within the dataset, with a higher score

indicating greater prevalence. The topic ‘email_valid_otp...’ sheds light on users’ challenges with OTP code verification sent to their email addresses or phone numbers, suggesting a potential area for enhancement in the verification process. Similarly, the topic labelled ‘error_sering_terus...’ underscores users’ recurrent encounters with application errors, highlighting the need for bug fixes and software optimisation.

Of particular concern is the prevalent issue of ‘login errors’, which emerges as users’ most frequently reported problem. This finding underscores the importance of addressing login functionality issues to streamline user access and enhance overall satisfaction. The analysis reveals that the application’s speed and responsiveness are critical areas of concern, as indicated by the topic of slow performance. Stakeholders are urged to prioritise efforts to optimise application performance, ensuring seamless user interaction and minimising frustration.

4. CONCLUSION

NLP is crucial in extracting valuable insights from user interactions within the m-passport application. By leveraging NLP techniques, we can effectively analyse user feedback to identify areas of improvement and enhance product quality and service delivery strategies. User reviews serve as a rich source of information, enabling us to understand their complaints and respond to their needs, ultimately aiming to elevate the application’s quality and user experience. NLP allows us to gauge user sentiment by analysing the emojis in their reviews. We have identified several key issues that require immediate attention, including OTP, login errors, and performance issues. These findings underscore the importance of innovation and proactive measures to address these issues effectively. Stakeholders can now focus on improving the OTP code delivery process, resolving login errors, and optimising the application’s performance. User satisfaction is a vital indicator of organisational success, emphasising the importance of continuously striving to meet and exceed user expectations through iterative improvement.

REFERENCES

- [1] A. M. A. Grisales, S. Robledo, and M. Zuluaga, “Topic modeling: perspectives from a literature review,” *IEEE Access*, vol. 11, pp. 4066–4078, 2023, doi: 10.1109/ACCESS.2022.3232939.
- [2] V. S. Tomashevskaya and D. A. Yakovlev, “Research of unstructured data interpretation problems,” *Russian Technological Journal*, vol. 9, no. 1, pp. 7–17, Mar. 2021, doi: 10.32362/2500-316X-2021-9-1-7-17.
- [3] M. Agarwal, “An overview of natural language processing,” *International Journal for Research in Applied Science and Engineering Technology*, vol. 7, no. 5, pp. 2811–2813, May 2019, doi: 10.22214/ijraset.2019.5462.
- [4] M. N. O. Sadiku, Y. Zhou, and S. M. Musa, “Natural language processing,” *International Journal of Advances in Scientific Research and Engineering*, vol. 4, no. 5, pp. 68–70, 2018, doi: 10.31695/IJASRE.2018.32708.
- [5] M. D. Lytras, A. Visvizi, and J. Jussila, “Social media mining for smart cities and smart villages research,” *Soft Computing*, vol. 24, no. 15, pp. 10983–10987, Aug. 2020, doi: 10.1007/s00500-020-05084-3.
- [6] B. Bhukya, M. Sheshikala, and B. Bhukya, “NLP based topic modeling for healthcare: analyzing patient reviews to improve quality of care and access to services,” in *2023 International Conference on Emerging Techniques in Computational Intelligence (ICETCI)*, Sep. 2023, pp. 7–12, doi: 10.1109/ICETCI58599.2023.10330957.
- [7] A. B. Dieng, F. J. R. Ruiz, and D. M. Blei, “Topic modeling in embedding spaces,” *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 439–453, Dec. 2020, doi: 10.1162/tacl_a_00325.
- [8] R. Parimala and K. Gomathi, “TOPNMF: topic based document clustering using non-negative matrix factorization,” *Indian Journal of Science and Technology*, vol. 14, no. 31, pp. 2590–2595, 2021, doi: 10.17485/ijst/v14i31.1293.
- [9] A. J. Rawat, S. Ghildiyal, and A. K. Dixit, “Topic modelling of legal documents using NLP and bidirectional encoder representations from transformers,” *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 28, no. 3, pp. 1749–1755, Dec. 2022, doi: 10.11591/ijeecs.v28.i3.pp1749-1755.
- [10] S. K. Ray, A. Ahmad, and C. A. Kumar, “Review and implementation of topic modeling in Hindi,” *Applied Artificial Intelligence*, vol. 33, no. 11, pp. 979–1007, Sep. 2019, doi: 10.1080/08839514.2019.1661576.
- [11] I. Vayansky and S. A. P. Kumar, “A review of topic modeling methods,” *Information Systems*, vol. 94, p. 101582, Dec. 2020, doi: 10.1016/j.is.2020.101582.
- [12] M. Grootendorst, “BERTopic: Neural topic modeling with a class-based TF-IDF procedure,” *arXiv preprint*, 2022, [Online]. Available: <http://arxiv.org/abs/2203.05794>.
- [13] K. Pykes, “What is topic modeling? an introduction with examples,” *Datacamp*, 2023. <https://www.datacamp.com/tutorial/what-is-topic-modeling> (accessed Feb. 27, 2024).
- [14] Cogitotech, “Topic modeling: algorithms, techniques, and application,” *Data Science Central*, 2021. <https://www.datasciencecentral.com/topic-modeling-algorithms-techniques-and-application> (accessed Feb. 02, 2024).
- [15] M. R. da S. Souza Filho and M. M. F. Pereira, “Compliance,” *Revista do Instituto de Direito Constitucional e Cidadania*, vol. 5, no. 2, p. e006, Dec. 2020, doi: 10.48159/revistadoidcc.v5n2.souzafileho.pereira.
- [16] W. J. Reynolds, “Compliance,” in *Safety and Health for the Stage*, Routledge, 2020, pp. 66–108.
- [17] N. Rosli and S. M. Nayan, “Why customer first?,” *Journal of Undergraduate Social Science and Technology*, vol. 2, no. 2, pp. 505–524, 2020.
- [18] A. Al-Hawari, H. Najadat, and R. Shatnawi, “Classification of application reviews into software maintenance tasks using data mining techniques,” *Software Quality Journal*, vol. 29, no. 3, pp. 667–703, 2021, doi: 10.1007/s11219-020-09529-8.
- [19] J. Dąbrowski, E. Letier, A. Perini, and A. Susi, “Finding and analyzing app reviews related to specific features: a research preview,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11412 LNCS, pp. 183–189, 2019, doi: 10.1007/978-3-030-15538-4_14.




- [20] J. Hendrickx, A. Van Remoortere, and M. Opgenhaffen, "News packaging during a pandemic: a computational analysis of news diffusion via Facebook," *Discourse & Communication*, vol. 17, no. 6, pp. 701–720, Dec. 2023, doi: 10.1177/17504813231177280.
- [21] J. Mishra, "Twitter sentiment analysis," *Interantional Journal of Scientific Research in Engineering and Management*, vol. 07, no. 06, Jun. 2023, doi: 10.55041/IJSREM24071.
- [22] P. Assiroj, A. Kurnia, and S. Alam, "The performance of Naïve Bayes, support vector machine, and logistic regression on Indonesia immigration sentiment analysis," *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 6, pp. 3843–3852, Dec. 2023, doi: 10.11591/eei.v12i6.5688.
- [23] A. Jabbar, S. Iqbal, M. I. Tamimy, S. Hussain, and A. Akhunzada, "Empirical evaluation and study of text stemming algorithms," *Artificial Intelligence Review*, vol. 53, no. 8, pp. 5559–5588, Dec. 2020, doi: 10.1007/s10462-020-09828-3.
- [24] J. Xie, Y. Li, Q. Sun, and Y. Lin, "Enhancing sentence embedding with dynamic interaction," *Applied Intelligence*, vol. 49, no. 9, pp. 3283–3292, Sep. 2019, doi: 10.1007/s10489-019-01456-x.
- [25] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using siamese BERT-networks," in *EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference*, 2019, pp. 3982–3992, doi: 10.18653/v1/d19-1410.
- [26] B. Wang, F. Chen, Y. Wang, and C.-C. Jay Kuo, "Efficient sentence embedding via semantic subspace analysis," in *2020 25th International Conference on Pattern Recognition (ICPR)*, Jan. 2021, pp. 119–125, doi: 10.1109/ICPR48806.2021.9412169.
- [27] H.-W. Cho, "Topic modeling," *Osong Public Health and Research Perspectives*, vol. 10, no. 3, pp. 115–116, Jun. 2019, doi: 10.24171/j.phrp.2019.10.3.01.

BIOGRAPHIES OF AUTHORS






Dr. Priati Assiroj, S.Kom., M.Kom    received her Ph.D. in Computer Science from Bina Nusantara University, Jakarta, Indonesia in 2022. Currently she is a lecturer in Politeknik Imigrasi, Ministry of Law and Human Rights, Republic of Indonesia. Since. Her research fields are data mining, high-performance computing, and evolutionary algorithms. She can be contacted at email: priati.assiroj@poltekim.ac.id or tie.assiroj@gmail.com.



Sirojul Alam, S.Kom    received a Bachelor's of Computer in Information Technology from Buana Perjuangan University, Indonesia. Currently, he is a Data Governance Officer in the Digital Business of Perum Peruri Indonesia. He also enrolled as a student of Master of Cyber Defense Engineering at The Republic of Indonesia Defense University. His research interests are machine learning, artificial intelligence, and Python programming. He can be contacted at: sirojul.alam@tp.idu.ac.id, sirojmu@gmail.com, or sirojul.alam@peruri.co.id.



Prof. Harco Leslie Hendric Spits Warnars, S.Kom, M.T.I., Ph.D    received his Ph.D. in Computer Science from Manchester Metropolitan University, United Kingdom. He is a Professor in Computing and Head of Information Systems Concentration in the Department of Doctor of Computer Science at Bina Nusantara University (<https://dcs.binus.ac.id>), working on some project research with my doctoral Computer Science students in the research area of computer science, such as intelligent information systems, intelligent tutoring systems, technology for the disabled, games, artificial intelligence implementation including data mining, machine learning, and DSS. He has published over 331 papers cited by 2,810 papers with h-index of 28 and i10-index of 78. He has Scopus publication with 249 documents with 1,036 citations cited by 786 documents with h-index of 16. He can be contacted at: spits.hendric@binus.ac.id or shendric@binus.edu.