

Topic Modelling of TikTok Application User Complaints in Google Play Store Reviews Using Latent Dirichlet Allocation

Ghevira Devanda^{1*}, Inaya Zehan Kalyzta², Zaqi Kurniawan³

Information System^{1*, 2, 3}
Budi Luhur University, Jakarta, Indonesia^{1, 2, 3}
<https://www.budiluhur.ac.id/>^{1, 2, 3}
2412502003@student.budiluhur.ac.id¹, 2412500627@student.budiluhur.ac.id², zaqi.kurniawan@budiluhur.ac.id³

Abstract. TikTok is a short-video-sharing platform with an extremely large user base in Indonesia, and as a result it also receives numerous negative reviews concerning both technical performance and platform policies on the Google Play Store. This study aims to identify and cluster the main topics of TikTok user complaints using a *Machine Learning*-based approach built on the *Latent Dirichlet Allocation* (LDA) algorithm. Data were collected through a quota-based *web scraping* procedure using the *google-play-scraper* library, targeting Indonesian-language reviews with a *rating* below three (rating 1 and 2) up to a target of 1,000 reviews, yielding a raw sample of 1,000 reviews (814 one-star and 186 two-star). The *preprocessing* stage included *cleaning*, *case folding*, normalization using a colloquial lexicon enriched with a TikTok domain-specific correction dictionary, *stopword* removal, and *stemming* using Sastrawi, which reduced the data to 814 clean documents. Document representation was constructed using a *Bag of Words* (BoW) approach through the Gensim library, with vocabulary filtering (*no_below* = 3, *no_above* = 0.6) that produced a *dictionary* consisting of 397 unique tokens. The optimal number of topics was determined through a multi-metric evaluation combining *Coherence Score* (*u_{mass}*), *Coherence Score* (*c_v*), and *Perplexity* for candidate models ranging from two to five topics, complemented by a manual check of topic interpretability. Despite mixed signals from the *c_v* and *Perplexity* metrics, the two-topic model was selected based on its *u_{mass}* score (-8.7713) and its more interpretable, non-fragmented thematic structure. The document distribution results show that Topic 1 is more dominant, comprising 445 documents (54.67%), representing technical application complaints such as *bugs*, slow performance, failure to open the application, and excessive advertisements. Meanwhile, Topic 2, with 369 documents (45.33%), relates to complaints about content moderation and restrictions on the *live* streaming feature. Visualization using *pyLDAvis* confirmed that the two topics are clearly separated without overlap, reinforcing the validity of the modelling results. As the sample is restricted to Indonesian-language, one- and two-star reviews collected at a single point in time, the findings are intended to reflect this specific segment of TikTok users rather than the full population of TikTok reviews. This study provides objective insights for application developers in prioritizing system improvements and evaluating content moderation policies more effectively.

Key words: Coherence Score; User Complaints; Latent Dirichlet Allocation; Topic Modelling; Text Mining; TikTok.

1. INTRODUCTION

The development of digital technology has driven social media and *mobile* application platforms to become the primary space for the public to express opinions, including complaints about a product or service. One form of such opinion is the user review available on application distribution platforms such as the Google Play Store. TikTok, as a short-video-sharing application with an extremely large user base in Indonesia, also receives numerous reviews covering opinions, technical complaints,

and responses to the platform's service policies. Such large-scale textual data is highly valuable for developers in evaluating and improving the quality of their services [1]. However, manually reading and categorizing such a large and unstructured volume of user-generated text, whether in the form of app reviews or public complaints, is clearly inefficient and prone to human error and subjectivity [2]. To address this challenge, a computational approach based on *Text Mining* combined with *Topic Modelling* is required [3]. Topic modelling is an *unsupervised machine learning*



method designed to extract information and uncover hidden thematic structures from a collection of texts automatically, without requiring prior data labelling. This approach has proven effective in processing large-scale textual data, ranging from mapping public complaints [2], to modelling government policy documents [4], and analyzing consumer opinions on e-commerce platforms [5]. Among the various topic modelling algorithms, *Latent Dirichlet Allocation* (LDA) is one of the most widely used generative probabilistic models, as it can effectively identify hidden topic structures within a collection of documents [6]. This algorithm operates on the assumption that each document is composed of a mixture of several hidden topics, while each topic is represented as a probability distribution over a set of words [7]. LDA has a notable advantage in that it can summarize, cluster, and process unstructured document collections into meaningful topics without requiring labelled data [6], as has been successfully implemented in modelling Indonesian national news texts [8].

The successful implementation of LDA is also evident in numerous studies analyzing social media and digital platform texts. Several previous studies have applied LDA to cluster discussion trends on Twitter, although comparative evaluations indicate that LDA's performance on such short, unstructured text is often surpassed by newer embedding-based methods [7]. On longer, review-style text, however, LDA has been shown to map topics in mobile application reviews on the Google Play Store effectively [9]. Narrowing the focus to app-store user complaints, the LDA method has proven effective in identifying user-experience themes through app review topic modelling across major app distribution platforms [10], analyzing user satisfaction with the BSI Mobile digital banking application on the Google Play Store [11], and mapping reviews of the PeduliLindungi health application [12].

Based on this review, most previous studies have applied LDA to the Twitter platform or to applications other than TikTok, while in-depth analyses specifically examining the characteristics of reviews with an extremely negative rating (1 and 2 stars) for the Indonesian-language TikTok application remain limited, particularly studies that combine domain-specific normalization of non-standard vocabulary (slang) with a topic-count selection procedure validated through more than a single *Coherence Score* metric. This limitation represents a research gap that needs to be filled, given that extracting topics from public complaints is essential for formulating policy insights and evaluating system services more effectively. The novelty of this study therefore lies in three aspects: (1) its specific focus on Indonesian-language TikTok reviews rated one and two stars, a segment not previously examined in detail by prior LDA-based app-review research; (2) the use of a TikTok domain-specific correction dictionary layered on top of a general Indonesian colloquial lexicon during

preprocessing; and (3) the determination and validation of the optimal number of topics through three complementary metrics, namely *Coherence Score* (u_mass), *Coherence Score* (c_v), and *Perplexity*, combined with a manual check of topic interpretability, rather than relying on a single coherence metric as in most prior LDA-based app-review studies.

Unlike previous studies, this research specifically uses Indonesian-language reviews with a *rating* below three for the TikTok application on the Google Play Store, with the aim that the resulting topics better represent user complaints as material for evaluating service quality. This study aims to implement the LDA algorithm to identify the main topics of TikTok users' negative complaints, determine and validate the optimal number of topics through a multi-metric *coherence* evaluation (u_mass , c_v , and *Perplexity*) supported by a manual interpretability check, and interpret the modelling results through *word cloud*, document distribution, and *pyLDavis* visualizations, so that they can serve as a data-driven basis for system improvement among developers, within the scope of the sampled one- and two-star Indonesian-language reviews analyzed.

2. RELATED WORK

2.1. Topic Modelling in Public and Academic Text Data

The application of *Text Mining* with the LDA algorithm has been widely used to extract information from various public and institutional documents. In the public service sector, LDA has been applied to identify topics and trends in public complaints in Tomohon City, making it easier for local governments to map problem priorities [2]. This algorithm has also been shown to map topics in Indonesian national news texts effectively [8], while at a broader institutional scale it has been used to trace the evolution of government policy attention during the COVID-19 pandemic in China [4], and to extract the underlying factors driving customer satisfaction from large-scale online consumer reviews in e-commerce settings [5].

The capability of LDA is also relied upon beyond formal government text, although its performance is known to vary with text length and structure. A systematic review of topic modelling on short social media text cautions that standard evaluation metrics such as coherence scores can be unreliable estimates of topic quality for tweet-length content, and that determining the appropriate number of topics for such corpora remains methodologically challenging [3]. In contrast, comparative studies on longer, review-style user-generated text show that a well-tuned LDA pipeline, built on a Bag-of-Words representation with coherence-based hyperparameter selection, can reliably surface interpretable themes [9]. This series of studies indicates that LDA's reliability depends heavily on the nature of the text being modelled, which is one reason this study focuses on Google Play Store reviews, which are



generally longer and more structured than microblog posts, rather than on short social media text.

2.2. Topic Extraction in the Social Media Ecosystem

Topic modelling methods show a more mixed level of accuracy when applied to social media text, which generally uses colloquial, short, and non-standard language. A direct comparison of four topic modelling techniques on Twitter posts, namely LDA, Non-negative Matrix Factorization (NMF), Top2Vec, and BERTopic, found that BERTopic and NMF were more effective than LDA at surfacing coherent and interpretable themes from short, unstructured Twitter text [7]. This finding reinforces that LDA's effectiveness is closely tied to the length and structure of the text corpus being modelled, a consideration this study addresses by focusing on comparatively longer application reviews rather than short social media posts.

2.3. Application Review Analysis on Digital Distribution Platforms

In the evaluation of software on digital distribution platforms such as the Google Play Store, user reviews serve as an important benchmark for assessing the level of consumer satisfaction. A systematic literature review of app review analysis for software engineering confirms that mining techniques such as topic modelling are among the most common approaches used to convert large volumes of app reviews into actionable insight for developers [1]. Building on this, topic modelling combined with sentiment analysis has been applied to online meeting application reviews to identify functional complaint themes during a period of rapid remote-work adoption [13], to reviews of digital-wellbeing ("digital detox") applications to uncover the emotional themes underlying user feedback [14], and to reviews of mobile medical applications to construct a user-satisfaction evaluation index [15]. In a sector that demands high system reliability, a comparative study of topic modelling techniques on a dataset of Nigerian bank customer tweets found that traditional LDA produced somewhat interpretable but inconsistent topics (coherence scores of 0.39 on the Cv measure and -9.12 on the u_{mass} measure), and that an embedding-based BERTopic approach ultimately achieved higher topic coherence, illustrating both the continued relevance of coherence-based LDA evaluation and its limitations on short, sparse social-media text [16]. Similarly, an analysis of Google Play Store reviews for a customer-relationship-management application identified negative user-experience themes through coherence-optimised LDA modelling [9], while a dimension-based topic and sentiment analysis of a Chinese mobile-health (m-Health) application's user reviews found that feature-related functionality and reliability were the dimensions with the greatest positive impact on user satisfaction, while bug-related complaints were associated with the lowest satisfaction scores [17]. In the Indonesian context specifically, this approach has also proven effective

in evaluating the service quality of the BSI Mobile digital banking application [11], while an analysis of 13,731 reviews of the PeduliLindungi health application identified five optimal topics based on the *coherence* value, covering registration issues, vaccine certificates, date-of-birth errors, difficulty opening the application, and general user complaints [12]. Unlike these studies, which generally identified many specific subtopics across varied application domains, this study focuses on TikTok reviews with an extremely negative rating in order to explore a more concise topic structure validated through *Coherence Score* evaluation.

3. METHODS

3.1. Research Stages

This research was carried out through several systematically organized stages, including the collection of TikTok review data from the Google Play Store, *preprocessing* of the text, the construction of a *dictionary* and *corpus* using a *Bag of Words* (BoW) approach, determination and validation of the optimal number of topics, training of the final *Latent Dirichlet Allocation* (LDA) model, and interpretation of the results through visualization. The entire process was carried out using the Python programming language, utilizing the *google-play-scraper*, *Sastrawi*, *Gensim*, *WordCloud*, and *pyLDAvis* libraries.

Before the final model was built, an evaluation process was conducted to determine the most optimal number of topics using three complementary metrics, namely *Coherence Score* (*u_{mass}*), *Coherence Score* (*c_v*), and *Perplexity*, together with a manual check of topic interpretability. The model selected using this combined procedure was then used as the final LDA model. The final stage involved visualizing the modelling results using a *word cloud*, a *Top-10* chart of dominant words for each topic, document distribution by topic, and an *Intertopic Distance Map* visualization using *pyLDAvis* to facilitate interpretation of the modelling results. The overall research flow is shown in Figure 1.

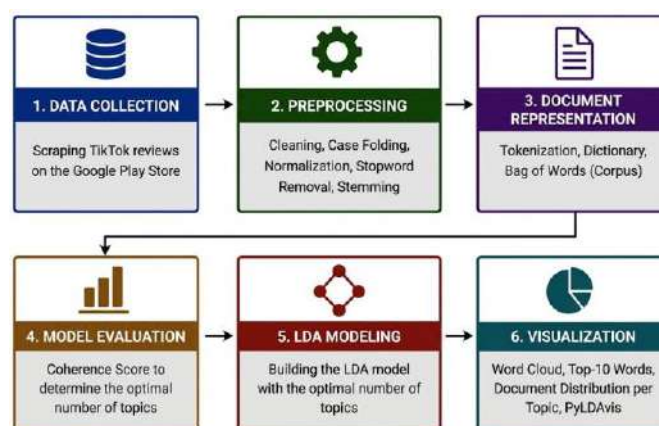


Fig. 1. Research Model



3.2. Data Collection

Research data were collected through a *scraping* process of TikTok application reviews (*com.ss.android.ugc.trill*) on the Google Play Store using the *google_play_scraper* library with Indonesian language parameters (*lang='id'*) and Indonesia as the country (*country='id'*). Sampling followed a purposive, criterion-based (non-probability) strategy with a fixed quota: reviews were retrieved in successive batches of 200 through the library's continuation-token pagination mechanism, sorted by the most recent reviews first (*Sort.NEWEST*); at each batch, only reviews with a *rating* below three (*rating* 1 or 2) were retained, and the process was repeated until a predetermined quota of 1,000 negative reviews was reached, which required 17 retrieval batches. No additional filtering by review length, language detection, or user account was applied beyond these rating and language parameters. Because this procedure targets the most recent negative reviews available on the Google Play Store at the time of retrieval rather than a random or time-stratified sample drawn across the application's entire review history, the resulting dataset should be interpreted as a snapshot of recent Indonesian-language, one- and two-star reviews rather than as representative of all TikTok reviews across every *rating* level and time period. The final raw sample consisted of 1,000 reviews in total, comprising 814 reviews with a *rating* of 1 and 186 reviews with a *rating* of 2, which were then passed on to the *preprocessing* stage.

3.3. Preprocessing

The *preprocessing* stage aims to improve data quality prior to text mining and analysis, so that the analyzed words are more representative and free of *noise* [1], [3]. The steps carried out include: (1) *case folding*, namely converting all text to lowercase; (2) *cleaning*, namely the removal of URLs, *mentions* (@username), the clitic suffix 'nya', and non-alphabetic characters; (3) normalization of non-standard words using an Indonesian colloquial dictionary (*colloquial-indonesian-lexicon*), which contains 4,331 slang word entries along with their standard equivalents, enriched with a TikTok domain-specific correction dictionary, for example 'eror'/'ngebug' converted to 'bug', 'ngelag'/'lemot' converted to 'lambat' (slow), and 'app'/'apk' converted to 'aplikasi' (application); (4) *stopword* removal using an Indonesian *stopword* list from NLTK, extended with common colloquial particles and filler words (for example 'sih', 'dong', 'deh', 'kayak', 'kalo', 'banget'), as well as several dominant words with the potential to cause overlap between topics, namely 'aplikasi' (application), 'akun' (account), 'video', 'bagus' (good), and 'update'; (5) filtering of profanity; and (6) *stemming* using the Sastrawi library to return each word to its root form. As an illustration, the review 'seru sih seru tapi banyak iklan' (fun, but there are a lot of ads) was normalized to 'seru seru iklan' after going through all of these stages. Documents that became empty or contained fewer than two tokens after *preprocessing* were subsequently removed from the

corpus, reducing the number of documents from 1,000 to 814 clean documents.

3.4. Document Representation

Documents resulting from *preprocessing* were subsequently tokenized and represented in the form of a *dictionary* and *corpus* using the Gensim library. This Bag-of-Words representation, combined with coherence-based hyperparameter tuning to determine the optimal number of topics, follows a similar general approach to previous coherence-based LDA topic-modelling studies, including app-review research [9] and short-text customer-discussion analysis [16]. Vocabulary filtering was then carried out using the *no_below* = 3 (a word must appear in at least three documents) and *no_above* = 0.6 (a word must not appear in more than 60% of documents) parameters, to reduce the influence of words that appear either too rarely or too frequently. This process produced a *dictionary* with 397 unique tokens, while the document representation was constructed using a *Bag of Words* (BoW) approach comprising 814 documents as input for the LDA model.

3.5. Model Evaluation and Topic Validation

The number of topics was determined by building several LDA models using topic counts ranging from $k = 2$ to $k = 5$. Determining the optimal number of topics is a well-recognised methodological challenge in LDA research, and various evaluation approaches have been proposed to address it [18]. In this study, each candidate model was evaluated using three complementary metrics: *Coherence Score* (*u_{mass}*), *Coherence Score* (*c_v*), and *Perplexity* (log-likelihood bound), an approach consistent with comparative topic-modelling studies that report more than one *coherence* measure for short, user-generated text [16]. The *u_{mass}* metric relies only on internal word *co-occurrence* within the *corpus* and does not require an external reference *corpus*, with values closer to zero indicating better topic *coherence*; the *c_v* metric instead uses a *sliding window* and normalized pointwise mutual information approach, with higher values indicating better *coherence*; and *Perplexity* approximates how well the model predicts the *corpus* on which it was trained, computed here as Gensim's log-*perplexity* bound. Beyond these quantitative metrics, the interpretability of the candidate topic solutions was also checked qualitatively: for the selected model, the researchers examined the *Top-10* keyword list of each topic together with the word-cloud and document-distribution outputs described below, to assess whether the resulting keyword clusters formed clearly distinguishable and semantically coherent complaint themes before the topic labels were finalized. This check draws on the model's own keyword and distributional outputs rather than on an independent reading of individual reviews; no manual or expert human annotation of individual reviews was performed to externally verify the topic assignments, a limitation discussed further in the



Conclusions. The model best supported jointly by these quantitative metrics and this qualitative coherence check was selected as the final model used in this study.

3.6. LDA Modelling

The final LDA model was built using the Gensim library based on the best number of topics from the evaluation, with *passes* = 20 (number of training iterations), *alpha* = 'auto', and *eta* = 'auto' (automatic estimation of the *prior* parameters for the topic and word distributions), as well as *random_state* = 42 to maintain result consistency across each run. This model was used to identify the word distribution for each topic and determine the topic distribution for each document based on the working principle of the *Latent Dirichlet Allocation* algorithm, namely modelling each document as a combination of several topics, while each topic is represented as a probability distribution of words [6], [7].

3.7. Visualization

The modelling results were visualized using several techniques, namely a *word cloud* to display the dominant words for each topic, a *Top-10* word chart to show word weights, a document distribution chart per topic to observe document spread, and *pyLDAvis* to visualize the relationships and degree of separation between topics through dimensionality reduction using *multidimensional scaling* (MDS). These visualizations assist in interpreting the characteristics of the topics and the relationships between the topics produced by the LDA model, consistent with the emphasis on visual topic interpretation found in comparable app-review topic-modelling studies [9].

4. RESULTS AND DISCUSSIONS

4.1. Preprocessing and Document Representation Results

The *scraping* process produced 1,000 negative reviews for the TikTok application, which subsequently went through the *preprocessing* stage, including *case folding*, *cleaning*, colloquial lexicon and TikTok domain normalization, *stopword* removal, and *stemming* using the Sastrawi library. After this stage, the number of documents decreased to 814 clean documents, a reduction of 186 documents due to the removal of reviews that were empty or contained fewer than two tokens after *preprocessing*. The cleaned documents were then tokenized and represented in the form of a *dictionary* and *corpus Bag of Words* (BoW) using the Gensim library. After the vocabulary filtering process with the *no_below* = 3 and *no_above* = 0.6 parameters, a *dictionary* consisting of 397 unique tokens was obtained, with the number of documents in the *corpus* remaining at 814 documents. This result indicates that the *preprocessing* process successfully filtered out low-quality and overly short reviews while retaining a sufficiently large and

representative document collection for the subsequent topic modelling stage.

4.2. Evaluation of the Optimal Number of Topics

TABLE 1. Coherence Score (*u_{mass}*), Coherence Score (*c_v*), and Perplexity Evaluation Results (*k* = 2–5)

Metric	<i>k</i> = 2	<i>k</i> = 3	<i>k</i> = 4	<i>k</i> = 5
<i>u_{mass}</i>	-8.7713	-9.7901	-10.5869	-11.1566
<i>c_v</i>	0.3021	0.3879	0.3521	0.3929
Perplexity	-5.6896	-5.7655	-5.7944	-5.8914

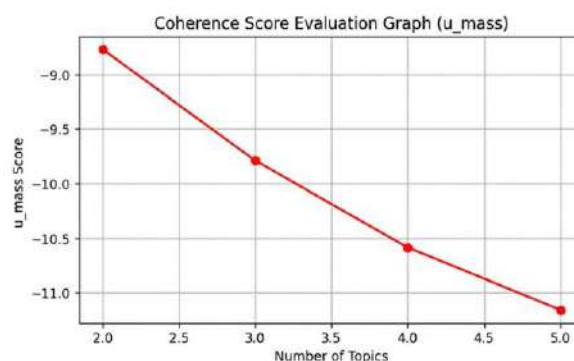


Fig. 2. Coherence Score (*u_{mass}*) Evaluation Chart

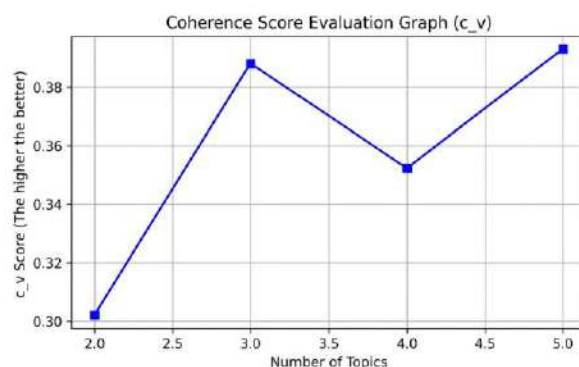


Fig. 3. Coherence Score (*c_v*) Evaluation Chart

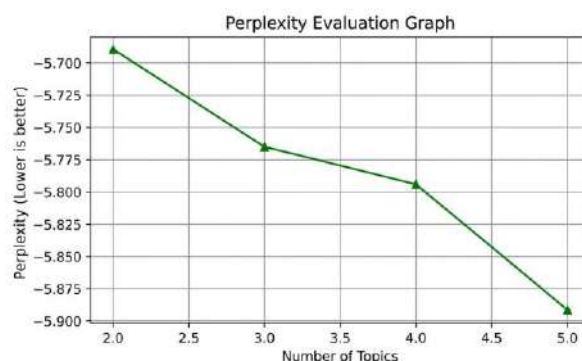


Fig. 4. Perplexity Evaluation Chart

Determination of the optimal number of topics was carried out by training several LDA models using varying numbers of topics, namely *k* = 2 to *k* = 5, and then evaluating each model using three complementary metrics: *Coherence Score*



(u_mass), *Coherence Score* (c_v), and *Perplexity*. Table 1 and Figure 2 show that the u_mass score decreased as the number of topics increased, from -8.7713 at $k = 2$ to -9.7901 at $k = 3$, -10.5869 at $k = 4$, and -11.1566 at $k = 5$, indicating that $k = 2$ produced the most internally coherent word *co-occurrence* structure among the candidates. The c_v score, shown in Figure 3, followed a less monotonic pattern, rising from 0.3021 at $k = 2$ to 0.3879 at $k = 3$, decreasing to 0.3521 at $k = 4$, and reaching its highest value of 0.3929 at $k = 5$; taken alone, this metric would favor a larger number of topics. The *Perplexity* score, shown in Figure 4, likewise became more negative as k increased, from -5.6896 at $k = 2$ to -5.8914 at $k = 5$, a pattern commonly observed when *perplexity* is measured on the same *corpus* used for training, since additional topics allow the model to fit the training data more closely without necessarily producing more meaningful or generalizable themes [18]. Because the three metrics did not point unanimously to the same k , the final choice of $k = 2$ was based on a combination of considerations. First, the u_mass metric, which is computed directly from the *corpus*'s own word *co-occurrence* statistics rather than from an external reference *corpus* or a within-sample likelihood bound, showed the clearest and most decisive preference for $k = 2$, with a substantial and consistent decline at every larger value of k . Second, both the c_v and *Perplexity* trends are computed on the same *corpus* used for model training and, particularly for a *corpus* with a still-limited vocabulary (397 unique tokens after filtering), can reward finer-grained partitions of the data without this necessarily reflecting more meaningful or generalizable themes, a known limitation of automatic *coherence* metrics on limited-vocabulary, user-generated text [3], [18]. Third, a qualitative check of the $k = 2$ solution's interpretability confirmed that it already produces two clearly distinguishable and internally consistent complaint themes (technical performance issues and content-moderation/policy issues). This check involved examining the Top-10 keyword lists together with the word-cloud and document-distribution outputs; it corroborates the model's own outputs rather than an independent reading of individual reviews, but still provides a defensible basis for selecting this model over the larger candidates despite their marginally higher c_v scores. Based on this combined quantitative and qualitative evidence, two topics were retained as the optimal number for building the final LDA model. The disagreement between metrics is acknowledged as a limitation of automatic topic-number selection and is revisited in the Conclusions.

4.3. LDA Topic Modelling Results

TABLE 2. Top 10 Words with the Highest Probability Weight Per Topic

No	Topic 1 (Weight)	Topic 2 (Weight)
1	aplikasi (0.063)	aplikasi (0.049)
2	buka (0.030)	live (0.029)
3	iklan (0.030)	suka (0.024)
4	lambat (0.028)	fitur (0.023)
5	hilang (0.022)	langgar (0.021)

No	Topic 1 (Weight)	Topic 2 (Weight)
6	unduh (0.021)	konten (0.019)
7	bug (0.019)	blokir (0.017)
8	lag (0.019)	tonton (0.014)
9	foto (0.019)	kasih (0.013)
10	muncul (0.018)	komentar (0.013)

The final LDA model was trained using $k = 2$ with the *alpha* and *eta* parameters automatically optimized (*auto*), *passes* = 20, and *random_state* = 42. Table 2 presents the ten words with the highest probability weight for each topic. Topic 1 is dominated by the words 'aplikasi' (application) (0.063), 'buka' (open) (0.030), 'iklan' (advertisement) (0.030), and 'lambat' (slow) (0.028), indicating that complaints in this topic center on application opening failures, slow performance, the occurrence of *bugs* and lag, and excessive advertising disruptions. Meanwhile, Topic 2 is dominated by the words 'aplikasi' (0.049), 'live' (0.029), 'suka' (like) (0.024), and 'fitur' (feature) (0.023), together with 'langgar' (violate) (0.021), 'konten' (content) (0.019), and 'blokir' (block) (0.017), which contextually represent user complaints related to content moderation, restrictions on the *live* streaming feature, and moderation decisions perceived by users as lacking transparency. These keyword-based interpretations are internally consistent with the word-cloud and document-distribution results reported below, supporting the use of 'technical performance issues' and 'content-moderation/policy issues' as the working labels for Topic 1 and Topic 2, respectively, in the remainder of this paper. This differs from LDA applications in other digital service domains, such as banking-sector customer discussions, where traditional LDA has been reported to produce comparatively less consistent topics than embedding-based methods on short, sparse text [16]. In this study, by contrast, LDA applied to the comparatively longer TikTok review text produced two clearly separated topics, with the dominant topic relating to the technical aspects of the application, reflecting differences in service characteristics between transaction-based financial platforms and video-sharing platforms based on user-generated content.

4.4. Document Distribution Per Topic

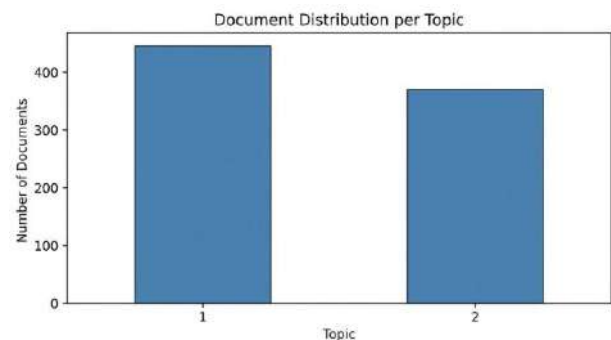


Fig. 5. Document Distribution Per Topic

Each document was subsequently classified into its dominant topic based on the highest topic probability generated by the model. Figure 5 shows that, of the total



814 documents, the document distribution is comparatively more balanced between the two topics than in the raw scraping composition: Topic 1 (technical complaints) is the more dominant topic, with 445 documents (54.67%), while Topic 2 (content moderation and policy) covers 369 documents (45.33%). This finding indicates that, although users who gave a low rating complained somewhat more often about technical application performance issues than about content moderation policies, the two complaint types occur at a broadly comparable rate, making both technical stability and policy transparency important evaluation priorities for developers.

4.5. Word Cloud and Top-10 Word Visualization



Fig. 6. Word Cloud for Each Topic

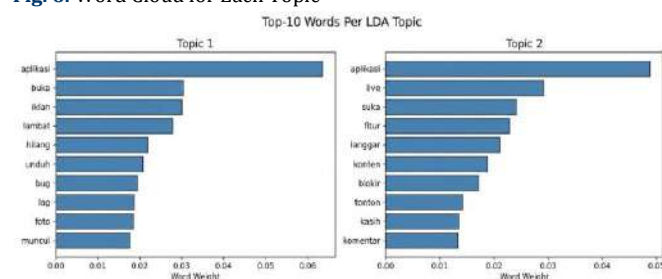


Fig. 7. Top-10 Word Chart Per LDA Topic

To facilitate visual interpretation, the modelling results were visualized in the form of a *word cloud* (Figure 6) and a *Top-10* word bar chart (Figure 7) for each topic. These visualizations clearly show the dominance of the words 'aplikasi', 'buka', and 'iklan' in Topic 1, as well as the words 'aplikasi', 'live', and 'suka' in Topic 2, reinforcing the topic interpretation described earlier. The size of the words in the *word cloud*, proportional to their probability weight, makes it easier for readers to quickly recognize the main theme of each topic without having to read through the entire numerical word distribution, while the *Top-10* word chart provides a more precise quantitative picture of the weight differences between words within a single topic.

4.6. Intertopic Distance Map Visualization

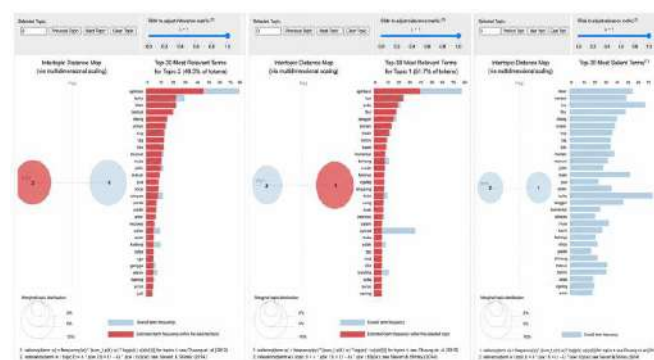


Fig. 8. Intertopic Distance Map from pyLDavis Visualization

The *Intertopic Distance Map* visualization using *pyLDavis* is used to assess the degree of separation between topics through *multidimensional scaling* (MDS) dimensionality reduction, where the size of the *bubble* represents the proportion of tokens in each topic. It should be noted that *pyLDavis* rennumbers the topics based on the order of token proportion from largest to smallest, so that *Topic 1* in the *pyLDavis* visualization (51.68% of the total tokens) corresponds to Topic 2 in the Gensim model (cluster 'live', 'suka', 'fitur', 'langgar', 'konten'), while *Topic 2* in *pyLDavis* (48.32% of the total tokens) corresponds to Topic 1 in the Gensim model (cluster 'aplikasi', 'buka', 'iklan', 'lambat'). Both *bubbles* of the topics on the *intertopic distance map*, shown in Figure 8, appear to form two clusters that are clearly separated without overlap (*non-overlapping*) and of broadly comparable size, indicating that the two topics have significantly different word characteristics without one topic being negligibly small. This clear topic separation further supports the validity of selecting $k = 2$ as the optimal number of topics, corroborating the u_mass -based and manual interpretability evidence discussed in the Evaluation of the Optimal Number of Topics section.

4.7. Discussion

The results of this study are consistent with previous international research on app-review topic modelling for user-experience evaluation [10] and with previous local research on the PeduliLindungi [12] application, which together demonstrate that the LDA method is highly effective for clustering user complaints on the Google Play Store. The difference is that this study successfully divided TikTok user complaints into two very distinct major groups: technical issues and policy issues (content moderation). This division will greatly assist *developers* in determining which areas should be addressed first. In practical terms, since Topic 1 (technical complaints) is complained about more frequently, the development team's main priority should be technical improvements, such as fixing *bugs*, making the application more stable, and adjusting the frequency of advertisement appearances.



Meanwhile, complaints in Topic 2 (content moderation and policy), as identified by this study's own LDA modelling, can serve as important input for TikTok's moderation team. Independent research on TikTok's content moderation practices in Southeast Asia argues that the platform's moderation decisions are often perceived as opaque and inconsistent, and calls for greater procedural transparency and fairness in enforcing platform rules [19], particularly those related to restrictions on the *live* feature and the removal of content owned by users.

5. CONCLUSIONS

This study has successfully applied the *Latent Dirichlet Allocation* (LDA) algorithm to cluster topics from 814 clean Indonesian-language reviews with a *rating* of 1 or 2 of the TikTok application on the Google Play Store, drawn from a raw sample of 1,000 recently posted negative reviews. Based on a combined evaluation using *Coherence Score* (u_{mass}), *Coherence Score* (c_v), *Perplexity*, and a manual check of topic interpretability, the two-topic model was selected as the most defensible option, with a u_{mass} coherence score of -8.7713.

From this modelling, two main user complaint themes were identified. Topic 1 (54.67%) is the more dominant theme and contains technical issues, such as application opening failures, *bugs*, lag, slow performance, and advertisement disruptions. Meanwhile, Topic 2 (45.33%) contains complaints about platform rules, such as moderation policy, blocking, violations, and restrictions on the *live streaming* feature. The separation between these two topics has also been examined through *pyLDAvis* visualization, which shows the two topics occupying non-overlapping positions, and the resulting keyword-based labels are corroborated by their internal consistency across the Top-10 keyword lists, word cloud, and document-distribution outputs; no independent manual or expert annotation of individual reviews was carried out to externally verify these assignments.

Because Topic 1 (technical issues) is discussed somewhat more frequently than Topic 2, it can be concluded that, within this sample, TikTok users who leave one- and two-star Indonesian-language reviews complain slightly more often about the application's technical stability than about content-moderation rules, although the two complaint types occur at a broadly comparable rate. This result can serve as a guide for TikTok developers to prioritize technical improvements (*bug* and system fixes) while also treating policy transparency as a comparably important area for the moderation team to address.

Nevertheless, this study still has several limitations that should be considered when interpreting its findings. First, model evaluation, although now supported by three complementary metrics (u_{mass} , c_v , and *Perplexity*) together with a manual interpretability check, still relies on automatic *coherence* measures whose reliability is known to vary for limited-vocabulary, user-generated text, and the

c_v and *Perplexity* metrics did not fully agree with the u_{mass} -based choice of $k = 2$. Second, the data analyzed was limited to a single, real-time scraping snapshot of 1,000 Indonesian-language reviews rated one or two stars, sorted by recency rather than drawn through random or stratified sampling; consequently, the topics identified and the conclusions drawn in this study should be understood as describing this specific sample of recent, low-rated, Indonesian-language TikTok reviews, and should not be generalized to TikTok users at large, to reviews in other languages, or to reviews with a neutral or positive rating. Third, topic labels ('technical performance issues' and 'content-moderation/policy issues') were assigned by the researchers based on the Top-10 keyword lists, word cloud, and document-distribution outputs of the model itself; no independent manual or expert annotation of individual reviews was performed to externally verify these assignments. This step was not carried out because the analyzed sample constitutes a single, non-archived snapshot of live Google Play Store data collected in real time (see Section 3.2); as review listings on the platform are continuously updated, an independent re-collection at a later date would draw a different, non-identical set of reviews, ruling out manual verification against the exact document-topic assignments reported here. As a result, the qualitative check reported above corroborates internal consistency rather than providing independent validation, which limits the strength of the topic-validation evidence. Future research is recommended to incorporate manual or crowd-sourced annotation of a sample of individual reviews to independently validate topic assignments, to validate the topic-count selection using held-out data or formal inter-annotator agreement procedures, to extend the data collection period and sampling frame to reduce time-of-collection bias, and to combine topic modelling with sentiment analysis to produce more complete and generalizable conclusions.

REFERENCES

- [1] J. Dąbrowski, E. Letier, A. Perini, and A. Susi, "Analysing app reviews for software engineering: a systematic literature review," *Empir. Softw. Eng.*, vol. 27, p. 43, 2022, doi: 10.1007/s10664-021-10065-7.
- [2] C. F. A. F. Polii and V. P. Rantung, "Analisis Topik Dan Tren Pengaduan Masyarakat Kota Tomohon Menggunakan Latent Dirichlet Allocation," *EduTIK J. Pendidik. Teknol. Inf. dan Komun.*, vol. 6, no. 1, pp. 348–358, 2026.
- [3] C. D. P. Laureate, W. Buntine, and H. Linger, "A systematic review of the use of topic models for short text social media analysis," *Artif. Intell. Rev.*, vol. 56, pp. 14223–14255, 2023, doi: 10.1007/s10462-023-10471-x.
- [4] Q. Cheng, J. Kang, and M. Lin, "Understanding the evolution of government attention in response to COVID-19 in China: A topic modeling approach," *Healthc.*, vol. 9, no. 7, p. 898, 2021, doi: 10.3390/healthcare9070898.
- [5] X. Zhao, L. Gao, and Z. Huang, "Customer satisfaction evaluation for drugs: A research based on online reviews and PROMETHEE-II method," *PLoS One*, vol. 18, no. 6, p. e0283340, 2023, doi: 10.1371/journal.pone.0283340.
- [6] A. Fuad and M. Al-Yahya, "Analysis and Classification of Mobile



- Apps Using Topic Modeling: A Case Study on Google Play Arabic Apps," *Complexity*, vol. 2021, p. 6677413, 2021, doi: 10.1155/2021/6677413.
- [7] R. Egger and J. Yu, "A Topic Modeling Comparison Between LDA, NMF, Top2Vec, and BERTopic to Demystify Twitter Posts," *Front. Sociol.*, vol. 7, p. 886498, 2022, doi: 10.3389/fsoc.2022.886498.
- [8] F. M. Hidayat, Cahyadi, H. Sanjaya, D. Purnomo, and H. Wiranto, "Pemodelan Topik Berita Nasional Indonesia Menggunakan Latent Dirichlet Allocation," *INFOTECH J.*, vol. 12, no. 1, pp. 33–39, 2026, doi: 10.31949/infotech.v12i1.16926.
- [9] O. Grljević, M. Marić, and R. Božić, "Exploring Mobile Application User Experience Through Topic Modeling," *Sustain.*, vol. 17, no. 3, p. 1109, 2025, doi: 10.3390/su17031109.
- [10] F. Ishfaq, S. N. K. Marwat, W. U. Khan, S. Shahzad, S. Khan, and Q. H. Abbasi, "A machine learning based empathy mapping framework for enhancing user experience through app review analysis," *Sci. Rep.*, vol. 16, p. 335, 2026, doi: 10.1038/s41598-025-30729-4.
- [11] R. Yulianti, "Implementasi Latent Dirichlet Allocation (LDA) Dalam Pemodelan Topik Ulasan Pengguna Bsi Mobile Di Google Play Store," Universitas Islam Indonesia, 2023.
- [12] J. Akbar, T. A. M., Y. Tolla, A. E. Ahmad, A. Yaqin, and E. Utami, "Pemodelan Topik Menggunakan Latent Dirichlet Allocation pada Ulasan Aplikasi PeduliLindungi," *InComTech J. Telekomun. dan Komput.*, vol. 13, no. 1, pp. 40–50, 2023, doi: 10.22441/incomtech.v13i1.15572.
- [13] N. Aslam, K. Xia, F. Rustam, E. Lee, and I. Ashraf, "Self voting classification model for online meeting app review sentiment analysis and topic modeling," *PeerJ Comput. Sci.*, vol. 8, p. e1141, 2022, doi: 10.7717/PEERJ-CS.1141.
- [14] N. F. Khan and M. N. Khan, "Analyzing user reviews on digital detox apps: A text mining and sentiment analysis approach," *J. Consum. Behav.*, vol. 24, no. 1, pp. 392–404, 2025, doi: 10.1002/cb.2424.
- [15] Y. Zhai, X. Song, Y. Chen, and W. Lu, "A Study of Mobile Medical App User Satisfaction Incorporating Theme Analysis and Review Sentiment Tendencies," *Int. J. Environ. Res. Public Health*, vol. 19, no. 12, p. 7466, 2022, doi: 10.3390/ijerph19127466.
- [16] B. Ogunleye, T. Maswera, L. Hirsch, J. Gaudoin, and T. Brunsdon, "Comparison of Topic Modelling Approaches in the Banking Context," *Appl. Sci.*, vol. 13, no. 2, p. 797, 2023, doi: 10.3390/app13020797.
- [17] F. Fang, Y. Zhou, S. Ying, and Z. Li, "A Study of the Ping An Health App Based on User Reviews with Sentiment Analysis," *Int. J. Environ. Res. Public Health*, vol. 20, no. 2, p. 1591, 2023, doi: 10.3390/ijerph20021591.
- [18] J. Gan and Y. Qi, "Selection of the optimal number of topics for LDA topic model—Taking patent policy analysis as an example," *Entropy*, vol. 23, no. 10, p. 1301, 2021, doi: 10.3390/e23101301.
- [19] D. Liu, "Borderline content and platformised speech governance: Mapping TikTok's moderation controversies in South and Southeast Asia," *Policy and Internet*, vol. 16, no. 3, pp. 543–566, 2024, doi: 10.1002/poi3.388.

