



Spell Correction for the Minangkabau Language Using BERT-Based Embeddings

Dewi Soyusiawaty¹, Abdul Fadlil², Sunardi³

¹Departement of Informatics, Industry Technology, Universitas Ahmad Dahlan, Yogyakarta, Indonesia

²Departement of Electrical Engineering, Industry Technology, Universitas Ahmad Dahlan, Yogyakarta, Indonesia

*dewi.soyusiawaty@tif.uad.ac.id

Abstract

Spell checking is an essential component of natural language processing, as it directly influences applications such as sentiment analysis, text classification, and machine translation. Developing a reliable system for low-resource languages like Minangkabau is challenging due to frequent spelling variations and limited annotated data. This study proposes a contextual spell correction model using pre-trained IndoBERT and multilingual BERT (mBERT) embeddings applied without additional training. The method masks misspelled words, extracts the contextual embedding of the [MASK] token, and compares it with candidate embeddings generated through dictionary filtering and Levenshtein Distance. Evaluation was conducted on the Spell Error Corpus for Minangkabau Language (SPEML), which includes insertion errors, deletion errors, substitution errors, transposition errors, punctuation errors, real-word errors, and loanword errors. Results show that mBERT consistently outperformed IndoBERT, achieving an average F1-score of 0.83 compared to 0.75. Statistical validation using paired t-test and Wilcoxon signed-rank test further confirmed that the performance difference between the two models was significant. Both models reached perfect scores (1.0) in real-word and loanword categories, and strong results in insertion_medium (0.97 for mBERT and 0.95 for IndoBERT). The lowest performance occurred in deletion_short (0.52 for IndoBERT) and long words cases (0.57 for mBERT). In addition, a small-scale external validation using 100 Twitter/X sentences was conducted to assess the applicability of the proposed method to real-world social media text. Overall, the findings confirm the effectiveness of contextual embeddings for Minangkabau spelling correction while highlighting challenges in long misspelled words, deletion errors, and informal real-world text.

Keywords: cosine similarity; IndoBERT; low-resource language; mBERT; minangkabau language

How to Cite: D. Soyusiawaty, Abdul Fadlil, and Sunardi, "Spell Correction for the Minangkabau Language Using BERT-Based Embeddings", *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 10, no. 3, pp. 689 - 701, Jun. 2026.

Permalink/DOI: <https://doi.org/10.29207/resti.v10i3.7182>

Received: September 16, 2025

Accepted: May 23, 2026

Available Online: June 16, 2026

This is an open-access article under the [CC BY 4.0 License](#)

Published by [Ikatan Ahli Informatika Indonesia](#)

1. Introduction

Natural Language Processing (NLP) has become a key component in the development of intelligent systems, supporting applications such as machine translation, sentiment analysis, and information retrieval [1]. However, technological advancement in NLP has not been evenly distributed across languages, particularly for regional languages in Indonesia that are still categorized as low-resource languages. One of these is the Minangkabau language, spoken by millions of users but lacking adequate computational resources and linguistic tools. Early work such as MinangNLP developed a Minangkabau-Indonesian corpus consisting of 5,000 sentiment sentences and 16,000 parallel sentence pairs from Twitter and Wikipedia [2]. The application of various classical and modern models

including machine learning methods and sequence-to-sequence architectures such as Long Short Term Memory (LSTM) and Transformer has been explored for sentiment analysis and Minangkabau-Indonesian translation [2]. In their machine translation experiments, the word-to-word (W2W) model built using a bilingual dictionary achieved BLEU scores of 64.54 on the Wikipedia corpus and 60.99 on the sentiment analysis corpus for the MIN→ID task. For neural models in the MIN→ID (Wikipedia) task, the LSTM model obtained a BLEU score of 63.77, while the Transformer model achieved 56.25.

Spelling correction is an essential component in text processing pipelines. Misspelled words whether at the character or word level can degrade performance in various NLP applications, including clinical text

interpretation, summarization, sentiment analysis, and machine translation [3-5]. BERT-based language models also produce different outputs when encountering misspelled tokens [5]. These challenges are amplified in human-generated regional corpora that often contain spelling inconsistencies, non-standard morphology, and informal grammatical structures.

In general, spelling errors are divided into two types: non-word errors, where tokens are not found in the corpus (out-of-vocabulary or OOV) and are therefore not considered vocabulary, and real-word errors (RWE), where tokens can be found in the corpus and are therefore considered syntactically valid but contextually inappropriate [6-8]. Classical approaches such as Levenshtein Distance, Jaro-Winkler, N-Gram, and Norvig's algorithm have been widely used for character-based correction [9-12]. Although efficient, these approaches do not consider semantic context and therefore struggle with real-word errors requiring contextual understanding [13].

Deep learning approaches, including LSTM and GRU, began to be used for sequence-based spelling correction [5], but they require large annotated datasets, which are unavailable for low-resource languages like Minangkabau [14-16]. The introduction of Transformer-based architectures accelerated progress in contextual modeling [17], followed by BERT, XLNet, RoBERTa, ALBERT, and GPT-2, which demonstrated strong performance on tasks such as sentiment analysis, question answering, summarization, NER, and topic modeling [18-20].

Recent developments in NLP have shown that Transformer-based models such as BERT provide strong performance in understanding sentence context [17], [21-23]. Several studies have applied BERT for spell checking in English and Indonesian, mostly through fine-tuning. Although effective, fine-tuning requires considerable computational resources and has not been widely adopted for regional languages. BERT's contextual representations support context-based spell checking by identifying misspelled words and generating corrections that fit the sentence context [17], [24]. BERT can produce correction candidates through masked language modeling, while edit distance algorithms help rank and select the most suitable candidates [5].

Soft-Masked BERT extends this approach through a dual-network architecture consisting of a BERT-based error detector and a BERT-based corrector connected via soft masking which shows improved accuracy, particularly for real-word errors in the SIGHAN Chinese benchmark [24]. Other hybrid methods integrate BERT with Levenshtein distance [9] or combine BERT with character similarity and edit distance, as in BEDSpell [10], leading to better context-appropriate corrections than classical approaches. Recent studies also introduce adaptive context-based real-time spelling correction for 24 languages [11], and

analyses of BERT's robustness to natural typos highlight challenges in handling noisy or informative words [12].

Various studies have examined spell checking and text correction for local and low-resource languages using NLP-based approaches [6], [25]. In Tamil, morphology-based and n-gram methods are employed to address its complex word structures [7]. For Indonesian, Peter Norvig- and Levenshtein-Distance-based spell checkers have been evaluated using the SPECIL corpus [26], [27]. The SinSpell work on Sinhala demonstrates the effectiveness of combining lexicon-based and probabilistic methods [8]. A survey on Greek further highlights the challenges in developing linguistic tools for Greek and other low-resource languages [28]. In Thai business information systems, query correction has become an important topic, utilizing domain-context approaches to improve search relevance [29], [30]. Meanwhile, studies on English and Arabic show that the Levenshtein Distance remains flexible for handling character-level errors in languages with diverse phonetic characteristics [31]. In addition, recent research in Amazigh [9], Kazakh [12], Persian [13], Vietnamese [15], and Turkish [17] also reports similar patterns, reflecting a broader shift toward contextual and hybrid spell-correction approaches across low-resource languages.

In the Indonesian context, the IndoBERT model [32] was developed using a large Indonesian-language corpus and evaluated through the IndoLEM benchmark, achieving state-of-the-art performance in tasks such as Named Entity Recognition (NER), POS tagging, and text coherence. IndoBERT was later extended to the social media domain through IndoBERTTweet, which incorporates domain-specific vocabulary from Twitter data [33]. Alongside IndoBERT, multilingual BERT (mBERT) is also relevant because it is trained on 104 languages, including Malay, which is linguistically close to Minangkabau. Indonesian and Minangkabau belong to the same Malayo-Polynesian Austronesian language family and share structural and lexical similarities [34], suggesting that both IndoBERT and mBERT have potential applicability for Minangkabau spell-checking tasks. However, no prior research has explicitly utilized BERT's contextual embeddings directly—without fine-tuning—for semantic-similarity-based spelling correction in Minangkabau [29], [35-38].

To provide a more comprehensive overview of recent developments in spelling correction, Table 1 summarizes some studies from the last five years across multiple languages, methods, and datasets. This expanded table highlights the evolution from classical algorithms to contextual and hybrid approaches, and further clarifies the research gap for Minangkabau contextual spelling correction.

Table 1. Summary of Recent Studies on Spelling Correction (2020–2025)

No	Study	Language	Method	Key Findings
1	Hu et al. (2020) [5]	English	Pre-trained BERT LM	Contextual LMs significantly improve correction accuracy.
2	Zhang et al. (2020) [21]	Chinese	Soft-Masked BERT	High accuracy but requires computationally expensive fine-tuning.
3	Ji et al. (2021) [23]	Chinese	SpellBERT	Lightweight BERT variant achieving competitive performance.
4	Liyanapathirana et al. (2021) [8]	Sinhala	Lexicon + probabilistic	Effective for spelling correction in low-resource.
5	Goonawardena et al. (2022) [30]	Sinhala	Spell + grammar correction	Improves overall text quality for Sinhala.
6	Chaabi & Ataa Allah (2022) [9]	Amazigh	Damerau-Levenshtein + N-gram	Strong for edit-distance error types.
7	Selvaraj et al. (2022) [7]	Tamil	NLP methods survey	Deep models recommended for morphologically complex Tamil.
8	Yanfi et al. (2022) [27]	Indonesian	Review of classical approaches	Classical methods insufficient for context-based errors.
9	Aziz et al. (2023) [10]	Urdu	Hybrid LM + rules	Improved detection and correction of real-word errors.
10	Laaroussi et al. (2023) [11]	Arabic	Context-based LM	Strong performance on long-sentence contextual dependencies.
11	Mukazhanov et al. (2023) [12]	Kazakh	Augmented Damerau-Levenshtein	Effective for character transpositions but limited for semantic errors.
12	Kasmaiee et al. (2023) [14]	Persian	Rule-based + deep learning	High performance with large annotated datasets.
13	Rahman et al. (2023) [22]	Bangla	CNN-blended BERT	Robust to natural Bangla typos; outperforms classical models.
14	Trung et al. (2024) [38]	Vietnamese	BERT + Transformer	Effective hybrid approach for Vietnamese.
15	Phukan et al. (2024) [16]	Assamese	Deep learning based detection	Performs well in low-resource Assamese spelling detection tasks.
16	Turhan (2025) [39]	Turkish	LLM-based correction	LLM achieves superior correction over classical methods.

The studies summarized in Table 1 show that recent approaches have increasingly adopted contextual and hybrid methods for spelling correction. However, these developments remain concentrated on high-resource languages or Indonesian, whose linguistic resources are considerably richer than Minangkabau. No prior work has examined the direct use of pretrained BERT embeddings without fine-tuning for contextual spelling correction in the Minangkabau language, nor evaluated performance across diverse error types such as insertion, deletion, substitution, transposition, real-word, and loanword errors as represented in SPEML. This gap highlights the need for a lightweight, context-aware method that can function effectively in low-resource settings.

To address this gap, this study proposes a contextual spelling correction approach that leverages the pretrained IndoBERT and multilingual BERT (mBERT) models without fine-tuning. Misspelled words are masked, contextual embeddings of the [MASK] token are obtained, and candidate corrections generated through lexicon filtering and Levenshtein Distance are ranked using cosine similarity. The SPEML corpus is employed to evaluate the system comprehensively across all error categories. In addition, statistical validation is conducted to support the comparison between IndoBERT and mBERT, and a small-scale external validation using Twitter/X data is added to examine the applicability of the method to real-world social media text.

The contributions of this study include: (1) introducing the first contextual-embedding-based spelling

correction framework for the Minangkabau language; (2) proposing a candidate ranking mechanism that integrates contextual similarity with edit-distance filtering; and (3) conducting a broad evaluation of IndoBERT and mBERT across all SPEML error types, supported by statistical validation and supplementary external validation on Twitter/X data. These contributions are expected to advance the development of NLP tools for low-resource regional languages.

2. Methods

2.1 Overview of the Proposed Method

The proposed approach is a contextual spell correction system for the Minangkabau language that utilizes two BERT models, namely IndoBERT and mBERT, to generate context-sensitive word embeddings in Minangkabau-language texts. The use of contextual BERT embeddings directly as the basis for correction is combined with dictionary-based matching and edit distance techniques. Thus, the system can handle spelling errors at the character level and consider the context of the sentence as a whole. Broadly speaking, the process includes: (1) tokenization of sentences and detection of suspected misspelled words, (2) masking of misspelled tokens and extraction of contextual embeddings using BERT, (3) generating correction candidates based on character similarity with the misspelled word (using a dictionary and the Levenshtein algorithm), and (4) selecting the best candidate by calculating the cosine similarity between the candidate embedding and the context embedding [MASK].

2.2 SPEML Dataset

The dataset used in this research is SPEML. This corpus includes pairs of misspelled sentences and their corrections [40], [41]. Each misspelled sentence contains exactly one word that is misspelled. SPEML is a Minangkabau language misspelling corpus grouped into seven types of spelling errors, namely: character insertion errors, character deletion errors, character substitution errors, character transposition errors, real-word errors, loanword errors, and punctuation errors [26].

The corpus also includes words with more than one character error, up to three-character errors. Table 2 shows the grouping of words according to word length and the number of typos in the word. For words with a length between 2 and 8 characters, there is one random character that causes the word to be misspelled. Furthermore, for words with a length between 8 characters up to a maximum of 12 characters, there are two random characters, and likewise, for words with a length of more than 12 characters, there are three random characters.

Table 2. Word Grouping and Number of Typo Characters

Word Length (L)	Total of Randomized Characters	Example of correct words	Example of misspelled words
$2 \leq L < 8$	1	lai	mai
$8 \leq L < 12$	2	bacakak	baacakakk
$L \geq 12$	3	pabandiangnan	apbandiagnna

Table 3 presents the list of files categorized by error type and word length within each error type (excluding punctuation errors, real-word errors, and loanword errors).

Table 3. List of Files Categorized by Error Type and Word Length

Error type by word length	Number of misspelled words
insertion_short	16,334 out of 16,334 sentences
insertion_long	16,334 out of 16,334 sentences
insertion_medium	16,334 out of 16,334 sentences
deletion_short	16,334 out of 16,334 sentences
deletion_medium	16,334 out of 16,334 sentences
deletion_long	16,334 out of 16,334 sentences
substitution_short	16,334 out of 16,369 sentences
substitution_medium	16,334 out of 16,369 sentences
substitution_long	16,334 out of 16,369 sentences
transposition_short	16,334 out of 16,369 sentences
transposition_medium	16,334 out of 16,369 sentences
transposition_long	16,334 out of 16,369 sentences
punctuation	16,334 out of 16,369 sentences
loan_word	1302 out of 16369 sentences

Each file consists of 16,334 sentences for the respective error type. For instance, insertion_short contains 16,334 sentences, each including one short misspelled word ($2 \leq L < 8$). Specifically, insertion_short denotes insertion errors in short words ($2 \leq L < 8$), insertion_medium denotes insertion errors in medium-length words ($8 \leq L < 12$), and insertion_long denotes insertion errors in long words (≥ 12 characters). In loanword errors, there are only 1,302 loanwords. This is due to the limited use of loanwords in sentences. Table 3 presents examples of character insertion errors, character deletion errors, character substitution errors, and character transposition errors.

Table 4 consists of four columns: Minang, which contains sentences without spelling errors; M_short, which contains sentences with one word containing one misspelled character; M_med, which contains sentences with one word containing two misspelled characters; and M_long, which contains sentences with one word containing three misspelled characters.

Table 4. Example of SPEML Corpus Content

Minang	M short	M med	M long
Insertion Error: animo masyarakaik dalam pacu itiak dikelola profesional sarato dibantuaknyo persatuan olahraga pacu itiak	animo masyarakaik dalam padcu itiak dikelola profesional sarato dibantuaknyo persatuan olahraga pacu itiak	animo masyarakaik dalam pacu itiak dikelola profesional sarato dibantuaknyo persatuaxn olahraga pacu itiak	animo masyarakaik dalam pacu itiak dikelola profesional sarato dibrantuainknyo persatuan olahraga pacu itiak
Substitution Error: mampajuangan tagaknyo syariat islam, isa mamilih bajuang parlemen	mampajuangan tagaknyo syariat islam, zsa mamilih bajuang parlemen	mampajuangan lagakmyo syariat islam, isa mamilih bajuang parlemen	mampajuayhan tagaknyo syariat islam, isa mamilih bajuang parlemen
Deletion Error: maliek sumangaiknyo animo masyarakaik dalam pacu itiak iko mako dikelola profesional sarato dibantuaknyo persatuan olahraga pacu itiak (porti)	mliiek sumangaiknyo animo masyarakaik dalam pacu itiak iko mako dikelola profesional sarato dibantuaknyo persatuan olahraga pacu itiak (porti)	maliek sumangaiknyo animo asyarkaik dalam pacu itiak iko mako dikelola profesional sarato dibantuaknyo persatuan olahraga pacu itiak (porti)	maliek sumangaiknyo animo masyarakaik dalam pacu itiak iko mako dikelola profesional sarato dibtuanyo persatuan olahraga pacu itiak (porti)
Transposition Error: maliek sumangaiknyo animo masyarakaik dalam pacu itiak iko mako dikelola profesional sarato dibantuaknyo persatuan olahraga pacu itiak (porti)	maliek sumangaiknyo animo masyarakaik dalam pacu itiak kio mako dikelola profesional sarato dibantuaknyo persatuan olahraga pacu itiak (porti)	maliek sumangaiknyo animo masyarakaik dalam pacu itiak iko mako dikelola profesional sarato dibantuaknyo nprsatae olahraga pacu itiak (porti)	maliek iumsngaaknyo animo masyarakaik dalam pacu itiak iko mako dikelola profesional sarato dibantuaknyo persatuan olahraga pacu itiak (porti)

Table 5 presents examples of punctuation errors, real-word errors, and loanword errors. Punctuation errors are errors in a word that contains punctuation marks, real-word errors are errors in real words that cause the

formation of other words, and loanword errors are errors in borrowed words.

Table 5. Punctuation Errors, Real-Word Errors, and Loan-Word Errors

Minang	Minang modified
Punctuation Error: taman nasional bunaken adalah taman lauik nan barado di sulawesi utara, indonesia	taman nasional bunaken adalah taman lauik nan barado di sulawesi utara: indonesia
Real-word Error: ibrahim sinik adalah seorang padagang asongan yang manjadi wartawan hebat	ibrahim sinik adalah seorang padagang asongan yang manjadi wartawan debat
Loanword Error: panerapan sistem praktik nan manyamoi sistem di industri marupkan langkah konsistensi manyiapkan tanago ahli madya	panerapan sistem praktek nan manyamoi sistem di industri marupkan langkah konsistensi manyiapkan tanago ahli madya

2.3. Contextual BERT Embedding for Error Detection

Each input sentence undergoes a tokenization process first. Tokenization breaks the sentence into token units that are ready for processing by the BERT model. The model generates a 768-dimensional embedding vector for each token based on the sentence context. The system detects spelling errors at the token level by comparing each token against a compiled Minangkabau word lexicon. The lexicon contains a list of verified standard Minangkabau vocabulary, which serves as a reference for checking whether a token is a valid word.

The total number of Minangkabau vocabulary words is 31,412. If a token is not found in the Minang dictionary (for example, “*malk*” is not a standard word), the token is marked as a candidate for a misspelled word (error). After determining the incorrect token, the system masks the token. The token is replaced with a special token [MASK] in its position in the sentence. BERT then reprocesses the masked sentence to generate context-based [MASK] embeddings. Next, these embeddings are compared with those of several candidate corrections [19].

2.4 Candidate Generation and Selection

Correction candidates for incorrect tokens are sourced from the Minang dictionary (verified), and the results of string matching using Levenshtein Distance (top-5 closest words) are provided. Each candidate is converted into an embedding using BERT (by inserting it into the same sentence to maintain context). Then, the cosine similarity between the [MASK] embedding and the candidate embedding is calculated. The candidate with the highest similarity is selected as the correction. To select the best correction, cosine similarity was computed between the contextual embedding of the [MASK] token A and the embedding of each candidate word B. Cosine similarity is defined in Equation 1.

$$\begin{aligned} \text{Cosine Similarity}(A, B) &= \frac{A \cdot B}{||A|| ||B||} \\ &= \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \cdot \sqrt{\sum_{i=1}^n B_i^2}} \end{aligned} \quad (1)$$

A is embedding vector of the [MASK] token. B is embedding vector of the candidate word, and n is dimension of the embedding vector (typically 768 for BERT). The cosine similarity value ranges from [0, 1] where a value close to 1 indicates that the candidate is highly consistent with the context.

Figure 1 shows the flow of contextual embedding-based spelling detection and correction.

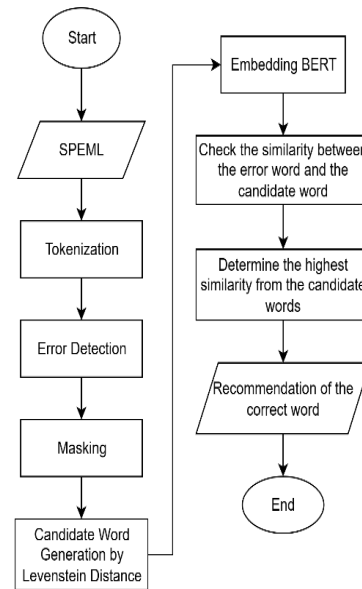


Figure 1. Flowchart of the Minang Language Spelling Correction Based on Contextual Embedding from BERT

2.5 Evaluation Metrics

System performance is evaluated by calculating accuracy, precision, recall, and F1-score in the process of detecting and correcting misspelled words. The evaluation is carried out separately for each spelling error category and word length level (short, medium, long) according to the division in the corpus. In this context, a case is considered correctly corrected if the system successfully identifies the incorrect word and replaces it with the correct one. Conversely, if the system does not correct the error or suggests an incorrect correction, it is counted as a failure in that category.

Accuracy is defined as the percentage of misspelled sentences that are entirely and correctly corrected by the system. Precision for a given category is calculated as the ratio of the number of correct corrections made by the system for errors in that category to the total number of corrections (correct + incorrect) proposed for that category. High precision indicates that when the system makes corrections in that category, most of them are correct. Recall is the ratio of the number of errors in that category that were correctly corrected by the system to the total number of cases of errors in that category in the dataset. Low recall means that many cases are missed and not corrected. F1-score is the harmonic mean of precision and recall, reflecting the balance of the system's ability to correct as many errors as possible

while minimizing incorrect corrections. All metrics are calculated per error class (insertion, deletion, etc., with each short/medium/long) to provide a detailed overview of performance. Table 6 shows the pseudocode of the correction system.

Table 6. Pseudocode of The Spelling Correction System

1	BEGIN
2	INPUT: Misspelled sentence (e.g. “Inyo indak mao datang”)
3	OUTPUT: Corrected sentence (e.g. “Inyo indak mau datang”)
4	Error Detection Find words not in vocab → incorrect_word = “mao”
5	Masking Replace the incorrect word with the token [MASK] masking_sentence = “Inyo indak [MASK] datang”
6	Candidate Generation Get a list of candidates based on Levenshtein Distance candidates = [“mau”, “mo”, “moa”]
7	Embedding Extraction Get BERT embedding vector from the token [MASK] → E_mask For each candidate -Insert candidate to the sentence → “Inyo indak mau datang” -Get embedding of the candidate token → E_candidate[i]
8	Cosine Similarity Calculation For each candidate: similarity[i] = cosine_similarity(E_mask, E_candidate[i])
9	Selecting the Best Candidate Take the candidate with the highest similarity as the correction
10	Sentence Correction Replace the incorrect word with the best candidate
11	RETURN corrected_sentence
12	END

2.6 External Validation Dataset

To evaluate the external validity of the proposed method, an additional validation was conducted using authentic Minangkabau text collected from Twitter/X posts. A total of 100 sentences were selected to represent natural and informal writing conditions commonly found in social media. Unlike SPEML, which provides controlled spelling-error categories, Twitter/X data contain naturally occurring spelling variations, non-standard word forms, informal morphology, inconsistent punctuation, and grammatical structures that may differ from the controlled corpus.

The collected sentences were manually reviewed to identify one target misspelled word in each sentence and its corresponding reference correction. This annotation was used to provide a gold-standard reference for evaluating whether the model could produce the expected correction. During preprocessing, the text was cleaned through case folding and punctuation handling before being processed by the proposed correction model. Table 7 presents examples of the external validation data. The highlighted tokens indicate the misspelled words and their corresponding reference corrections. This dataset was used as a supplementary validation set, while SPEML remained the main benchmark for systematic evaluation.

Table 7. Examples of Twitter/X Data for External Validation

text	reference
Kaluarga Banuaran Nan XX Lubuk Begalung Padang	Kaluarga Banuaran Nan XX Lubuak Bagaluang Padang
Baitu barieh balabiahnyo dari luhak maso dahulu kok tidak disigi dipanyato lipuah lah jajak nan dahulu	Baitu barieh balabiahnyo dari luhak maso dahulu kok indak disigi dipanyato lipuah lah jajak nan dahulu
Kalio jariah otw rendang Lawang Jawa Timur Indonesia	Kalio jariah otw rendang Lawang Jawa Timur Indonesia

3. Results and Discussions

Recent work on spelling correction in other low-resource languages provides a meaningful basis for situating the performance of our approach. Zhang et al. (2020) reported character-level F1-scores ranging from 0.66 to 0.73 for Chinese spelling correction using Soft-Masked BERT across the SIGHAN and News Title datasets, reflecting the difficulty of handling character distortions in logographic languages. Rahman et al. (2023) achieved an overall accuracy of approximately 0.80 for Bangla using the BSpell model, which integrates CNN features with BERT embeddings. Meanwhile, Trung et al. (2024) reported 81.5% correction accuracy and 86.8% detection accuracy for Vietnamese using a hybrid BERT Transformer architecture. Although these systems rely on fine-tuning or task-specific model adaptations, our approach—using only contextual embeddings from pretrained IndoBERT and mBERT without any additional training achieves an average F1-score of 0.83 on the SPEML dataset. This suggests that a lightweight embedding-based ranking method can provide a competitive baseline, although direct comparison with fine-tuned models remains necessary for a more complete performance-efficiency analysis.

3.1 Spell Detection and Correction Results

Figure 2 shows the spelling detection and correction process. For example, the incorrect sentence “Inyo indak mao datang” will be changed to “Inyo indak [MASK] datang”. Next, the system generates some correction word candidates, such as “mau”, “mo”, and “mao”, which are obtained from a dictionary or Levenshtein Distance-based selection results. For each candidate, a new sentence is created by replacing [MASK] with the candidate, such as “Inyo indak mau datang”, and the candidate token embedding in the new sentence is extracted using BERT. The final correction is determined by calculating the cosine similarity between the [MASK] embedding and the embedding of each candidate, then selecting the candidate with the highest similarity score. In this example, the system selects “mau” as the correction because it has the highest similarity score to the sentence context. This approach enables the system to select corrections based on the overall sentence context, resulting in more accurate handling of ambiguous or similar character spelling errors.

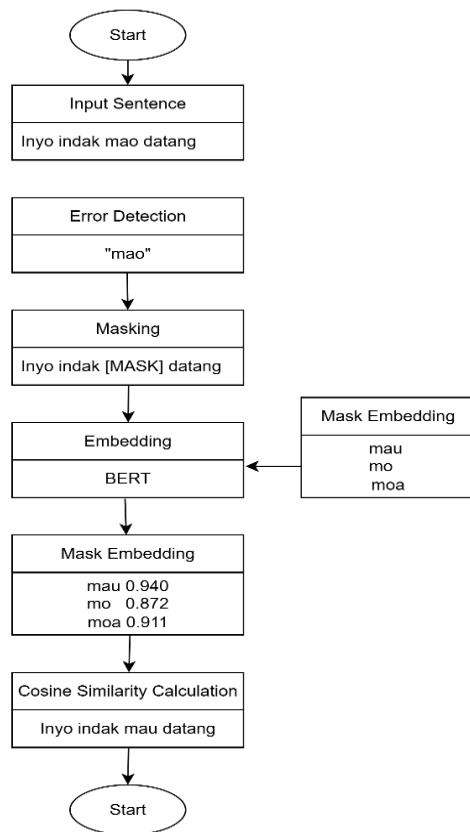


Figure 2. Spelling Detection and Correction Process

Table 8 presents the incorrect sentence data, the words detected as incorrect, and the correction results for all types of errors.

For example, in an insertion error, the correct word is “*mampajuangan*”, the word detected as misspelled is “*mampdajuanganna*”. In the spelling correction process, it is successfully corrected to “*mampajuangan*”. The corrected word is the same as the correct word, which means the system successfully corrected the word. This research evaluates the

performance of two contextual embedding models, IndoBERT and mBERT, in Minangkabau spelling correction tasks using the SPEML dataset. The evaluation was conducted on various types of spelling errors, including insertion, deletion, substitution, transposition, real-word errors, loanwords, and punctuation, and analyzed based on word length (short, medium, and long). Performance assessment encompasses four primary metrics: accuracy, precision, recall, and F1-score. The evaluation results are shown in Table 9.

Figure 3 presents a comparison graph of F1-score values per error type and per word length in the IndoBERT and mBERT models.

Table 8. Spelling Detection and Correction Results

Correct Word	Misspelled word	Corrected Word	T/F
Insertion Error:			
mampajuangan	mampdajuanganna	mampajuangan	T
kadalamannyo	kryadalamannjnyo	kadalamannyo	T
manghilangan	qmanghilakngban	carobongnyo	F
Deletion Error:			
pabandiangn	pabadiann	manyiaran	F
disempurnakan	dsmpunakan	disempurnakan	T
pamarintahan	amrinthan	ateh	F
Substitution Error:			
pabandiangn	pabangilygan	pabandiangn	T
penerjemahan	peneryemehn	goethe	F
mangggong	maksyenggong	mangggong	T
Transposition Error:			
malanjuikkan	malaanjakikun	malanjuikkan	T
ekstrakurikuler,	ekstrakurirulek	ekstrakurikuler,	T
pesantrennyo	nesyptrennao	barakhiany	F
Real-word Error:			
tidur	timur	Timur	T
dekan	pekan	Pekan	T
basabuik	tasabuik	tasabuik	T
Loanword Error:			
aktivis	aktifis	aktivis	T
renovasi	renofasi	renovasi	T
fasilitas	fasilities	fasilitas	T
Punctuation Error:			
program	program(	program	T
tiang	-tiang	tiang	T
'davros'	?davros'	'davros'	T
ka-22	ka[22	ka-22	T

Table 9. Spelling correction evaluation results on SPEML

	Accuracy		Precision		Recall		F1-score	
	IndoBERT	mBERT	IndoBERT	mBERT	IndoBERT	mBERT	IndoBERT	mBERT
Insertion_short	0.78	0.95	0.85	0.96	0.78	0.95	0.77	0.95
Insertion_med	0.95	0.97	0.96	0.97	0.95	0.97	0.95	0.97
Insertion_long	0.63	0.63	0.79	0.79	0.63	0.63	0.57	0.57
Deletion_short	0.60	0.79	0.78	0.85	0.60	0.79	0.52	0.78
Deletion_med	0.80	0.94	0.85	0.95	0.80	0.94	0.79	0.94
Deletion_long	0.61	0.63	0.78	0.79	0.61	0.63	0.55	0.57
Substitution_short	0.72	0.92	0.82	0.93	0.72	0.92	0.70	0.92
Substitution_med	0.93	0.96	0.94	0.97	0.93	0.96	0.93	0.96
Substitution_long	0.63	0.63	0.79	0.79	0.63	0.63	0.57	0.57
Transposition_short	0.73	0.91	0.82	0.93	0.73	0.91	0.70	0.91
Transposition_med	0.94	0.96	0.94	0.96	0.94	0.96	0.94	0.96
Transposition_long	0.63	0.63	0.79	0.79	0.63	0.63	0.57	0.57
Punctuation	0.74	0.84	0.83	0.88	0.74	0.84	0.72	0.83
Real-word	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Loanword	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Average	0.78	0.85	0.86	0.90	0.78	0.85	0.75	0.83

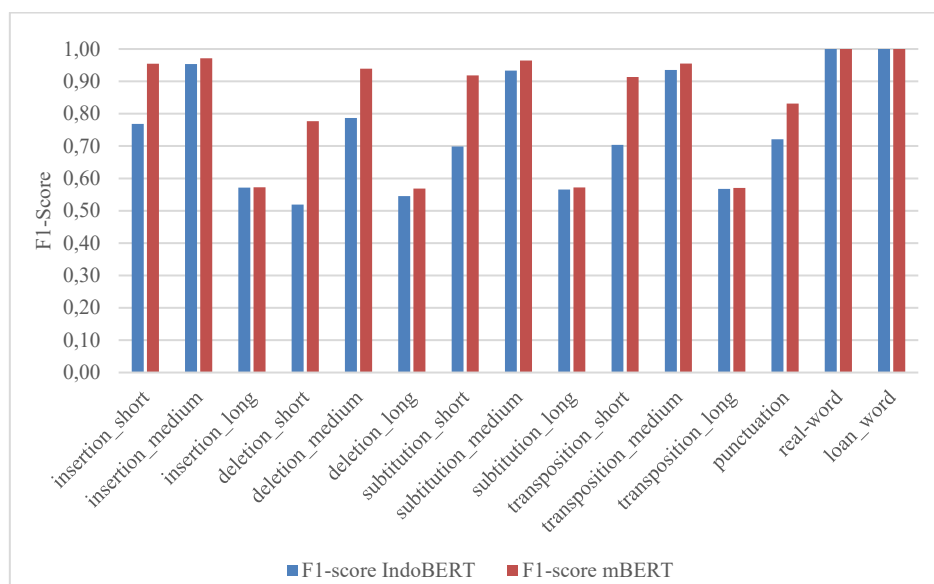


Figure 3. Comparison of IndoBERT and mBERT F1-score per Error Type

3.2 Comparison of IndoBERT and mBERT Model Performance

In the SPEML evaluation, the experimental results show that mBERT outperformed IndoBERT across almost all evaluation metrics, including accuracy, precision, recall, and F1-score. This result indicates that multilingual contextual representations provide advantages for Minangkabau spelling correction in the controlled SPEML setting. The highest performance for mBERT was achieved in the insertion_medium category, where it recorded an accuracy, precision, recall, and F1-score of 0.97. IndoBERT also performed strongly in this category, although slightly lower, with an F1-score of 0.95. On average, mBERT obtained an F1-score of 0.83, compared to 0.75 for IndoBERT. This category benefits from sufficient contextual and lexical information, which allows the models to more reliably select the correct candidate based on semantic cues.

A substantial performance gap is observed in the deletion_short category. IndoBERT records an F1-score of only 0.52, while mBERT achieves 0.78. Short tokens provide limited contextual and structural information, making their correction particularly challenging. mBERT's superior performance in this category suggests that it possesses stronger generalization capabilities when handling short and highly ambiguous Minangkabau word forms. To further clarify the structural reasons behind these performance differences, an additional linguistic and subword level analysis is presented. This explanation is important to demonstrate that mBERT's advantage is not solely the result of numerical performance, but also stems from broader vocabulary coverage and cross-linguistic representations that directly influence how each model interprets Minangkabau words.

As a multilingual model trained on 104 languages including Malay, which is linguistically close to Minangkabau, mBERT benefits from a shared

WordPiece vocabulary that encodes subword patterns common to Malayic languages. This multilingual exposure allows mBERT to recognize Minangkabau morphemes and variant forms even when they do not appear in its pretraining corpus. In contrast, IndoBERT is trained solely on Indonesian texts. Although Indonesian and Minangkabau share many lexical similarities, IndoBERT lacks exposure to the broader cross linguistic cues needed to handle Minangkabau-specific word variations and orthographic inconsistencies. These structural advantages are illustrated in Figure 4, which shows the shared subword vocabulary coverage between mBERT, Malay family languages, and Minangkabau variants.

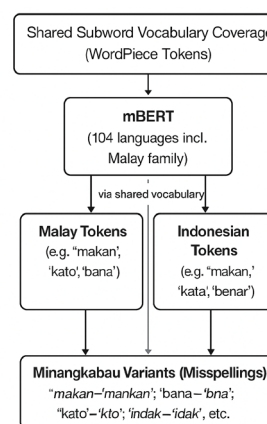


Figure 4. Illustration of Shared Subword Vocabulary in mBERT for Malayic Languages

Table 10 further provides real tokenization examples, demonstrating that mBERT can segment Minangkabau variants such as mankan, bna, and kto into meaningful subword units, whereas IndoBERT often generates [UNK] tokens or unstable segmentations. These examples provide qualitative evidence that aligns with the quantitative results reported in the SPEML evaluation.

Table 10. Tokenization Comparison Between IndoBERT and mBERT for Minangkabau Variants.

Word Variant	IndoBERT Tokenization	mBERT Tokenization
kato	[ka], [##to]	[ka], [##to]
kuto	[UNK]	[ku], [##to]
kato	[UNK]	[ka], [##too]
mankan (typo of makan)	[UNK]	[man], [##kan]
bna (typo of bana)	[UNK]	[b], [##na]

Overall, the stronger performance of mBERT in the SPEML evaluation can be attributed to its multilingual training signals, richer subword coverage, and greater sensitivity to lexical variations typical of regional languages. These structural strengths enable mBERT to capture contextual and morphological cues more effectively across most SPEML spelling error categories. These qualitative patterns reinforce the earlier quantitative findings on SPEML, confirming that mBERT's multilingual subword representations provide advantages in handling controlled Minangkabau spelling variations. This structural robustness aligns with the superior performance observed in the SPEML evaluation and supports the interpretation that mBERT is more suitable for the controlled SPEML spelling correction setting. However, this conclusion should be interpreted within the SPEML evaluation context, since real-world social media data may show different performance patterns due to informal spelling, abbreviations, code-mixing, and non-standard morphology.

3.3 Statistical Validation of Model Performance

To provide stronger evidence for the performance difference between IndoBERT and mBERT, a paired statistical analysis was conducted using the category-level F1-scores reported in Table 9. The analysis used 15 paired observations, where each pair consisted of the F1-score of IndoBERT and the corresponding F1-score of mBERT for the same error category. The difference for each pair was computed as $d = F1_{mBERT} - F1_{IndoBERT}$.

The total difference across all categories was 1.22, resulting in a mean difference of 0.0813. This indicates that, on average, mBERT improved the F1-score by approximately 0.081 compared with IndoBERT. The standard deviation of the paired differences was 0.09635, and the standard error was 0.02488. Based on these values, the paired t-test produced $t(14) = 3.269$ with $p = 0.0056$. Since the p-value is below 0.05, the difference between IndoBERT and mBERT can be considered statistically significant.

In addition, a Wilcoxon signed-rank test was conducted as a non-parametric validation because the number of paired observations was relatively small. After excluding zero-difference pairs, all non-zero differences favored mBERT, and the Wilcoxon test also showed a significant result with $p = 0.0050$. These findings confirm that mBERT's superiority over IndoBERT is not only observed descriptively in the

average F1-score, but is also supported by statistical validation.

3.4 The Influence of Word Length and Error Type on Correction Performance

In addition to differences between models, the experimental results show that word length and error type significantly affect the performance of the spelling correction system. In general, both IndoBERT and mBERT achieved the best performance on medium-length words, and performance declined for very short and very long words. This is evident from the highest average F1-scores for medium words (0.96 for mBERT and 0.90 for IndoBERT), compared to short words (0.89 vs 0.67) and long words (0.57 vs 0.56). Table 11 summarizes the comparative F1-score of IndoBERT and mBERT based on word length and error type.

Table 11. Comparative F1-score of IndoBERT and mBERT by Word Length and Error Type

Word Length	Error Type	IndoBERT	mBERT
Short	insertion_short	0.77	0.95
	deletion_short	0.52	0.78
	substitution_short	0.70	0.92
	transposition_short	0.70	0.91
	Average	0.67	0.89
Medium	insertion_med	0.95	0.97
	deletion_med	0.79	0.94
	substitution_med	0.93	0.96
	transposition_med	0.94	0.96
	Average	0.90	0.96
Long	insertion_long	0.57	0.57
	deletion_long	0.55	0.57
	substitution_long	0.57	0.57
	transposition_long	0.57	0.57
	Average	0.56	0.57

Figure 5 presents the average F1-score for each word length.

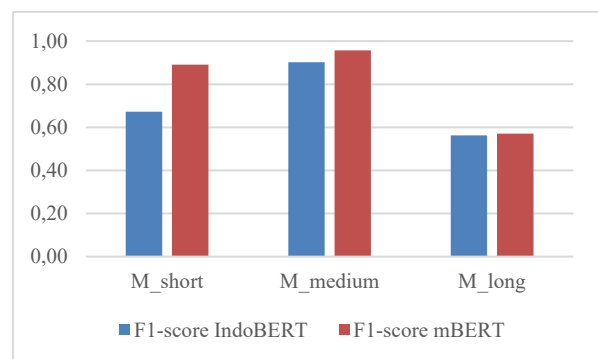


Figure 5. The Average F1-Score for Each Word Length

mBERT tends to maintain high performance across all word lengths, while IndoBERT appears to struggle more, particularly with the shortest words. The largest performance gap occurs in the short-word category, indicating that mBERT is more robust in handling very brief inputs with minimal context compared to IndoBERT. Meanwhile, medium-length words appear to provide sufficient contextual information without causing high ambiguity, making this category the ideal range for embedding models to capture patterns and generate accurate correction predictions.

In terms of error types, model performance varied across insertion, deletion, substitution, and transposition categories depending on word length. Consistently, medium-length words with any error type were easier to correct than very short or very long words. However, the most drastic performance declines occurred in specific combinations of error type and word length. The *deletion_long* category (character deletion errors in long words) showed the lowest performance: both IndoBERT and mBERT only achieved F1-scores around 0.5 (0.55 and 0.57, respectively). This reflects the difficulty of handling words whose structures become excessively long due to excessive character insertions, often resulting in out-of-pattern words. Conversely, in the *deletion_short* category (character deletion errors in very short words), IndoBERT achieved only 0.52, far below mBERT's 0.78. This indicates that short words losing characters are very difficult to recover because of weak semantic signals and limited available context. This decline in performance is consistent with previous research showing that standard BERT models encounter significant challenges in detecting insertion and deletion errors, particularly when the edit distance or number of altered characters is high [5], [42]. IndoBERT's much lower performance compared to mBERT in *deletion_short* reinforces the finding that mBERT is more sensitive to spelling errors in short words, likely due to its broader multilingual training corpus that better captures cross-linguistic contexts.

The analysis of word length, defined according to the error generation rules in SPEML, provides further insight into correction performance. Short words are defined as words containing 2–8 characters with one random error. For example, “*mai*” derived from the correct word “*lai*”. Such short tokens provide very limited semantic cues, and the contextual embedding of the [MASK] token relies heavily on surrounding words. This lack of internal structure causes multiple candidates to appear equally plausible, which explains the relatively low F1-scores in the short-word category, particularly for IndoBERT. Medium-length words, defined as words containing 8–12 characters with two random errors, show markedly better performance. For instance, “*mancaliakk*” derived from the correct “*mancaliak*” still retains much of its recognizable structure. In a sentence context, the embedding of the [MASK] token is shaped by sufficient contextual information, allowing the model to identify “*mancaliak*” as the most appropriate correction. Cosine similarity scores clearly favor the correct word over alternatives, which is why this category achieves the highest accuracy across both IndoBERT and mBERT.

In contrast, long words, defined as words with more than 12 characters and three random errors, pose the greatest difficulty. An example is “*mavandiangam*” derived from the correct “*pabandiangam*”. Errors of this type result in unnatural word forms that deviate strongly from standard vocabulary. Furthermore, long words are often segmented into multiple subwords in

BERT, and three random alterations can severely distort the embedding representation. As a result, the cosine similarity between the [MASK] token and the correct candidate is only marginally higher than for other alternatives, leading to the lowest F1-scores in categories such as *insertion_long* and *deletion_long*.

Overall, these results demonstrate that contextual embeddings are most effective for medium-length words, where contextual and structural information is sufficient to guide correction. Short words lack internal and contextual cues, while long words are disproportionately affected by subword tokenization and structural distortions. This confirms that word length is a critical factor in determining the success of contextual embedding-based spelling correction.

Beyond the four main categories, it is noteworthy that both models achieved very high performance in the *real-word* and *loanword* error categories. Both models even obtained perfect F1-scores (1.0) for real-word and loanword errors. Theoretically, these errors are among the most challenging, since the erroneous tokens are still valid lexical words, making them difficult to detect using dictionary-based approaches. However, these perfect results can be explained by the characteristics of the SPEML corpus: the distribution patterns of real-words and loanwords in Minangkabau sentences may be distinctive enough to be easily recognized by the BERT model, which relies on contextual information. In other words, the unusual contextual usage of these real-words or loanwords in the dataset is sufficiently strong to be differentiated by BERT embeddings, enabling the models to correct them accurately.

The implications of these findings are significant, both in theory and in practice. From a theoretical perspective, these results reinforce the view that contextual embeddings such as BERT are highly effective in spelling correction tasks for regional languages, especially when the type of error has strong contextual support. From a practical implementation perspective, this research demonstrates that an automatic spell-checking system for the Minangkabau language can be constructed using an embedding-based approach without the need for fine-tuning, by utilizing [MASK] token representations and cosine similarity calculations for candidates.

3.5 External Validation Dataset

To further examine the external validity of the proposed method, an additional evaluation was conducted using real-world Minangkabau sentences collected from Twitter/X. Unlike SPEML, which provides controlled and systematically generated spelling-error categories, the Twitter/X dataset represents more natural and informal writing conditions. The data contain spelling inconsistencies, abbreviations, non-standard word forms, punctuation variation, and mixed Indonesian–Minangkabau expressions. Table 12 presents External validation results on Twitter/X dataset.

Table 12. External Validation Results on Twitter/X Dataset

Model	Accuracy	Precision	Recall	F1-score
IndoBERT	0.72	1.00	0.72	0.84
mBERT	0.73	0.82	0.73	0.78

The external validation results show that mBERT achieved slightly higher accuracy than IndoBERT, with scores of 0.7375 and 0.7289, respectively. This indicates that mBERT was able to identify a slightly larger proportion of correct corrections in the Twitter/X dataset. However, a different pattern was observed in terms of precision and F1-score. IndoBERT achieved a precision score of 1.0000, indicating that the corrections proposed by IndoBERT in the evaluated Twitter/X data were highly precise. Nevertheless, its recall score of 0.7289 suggests that some misspelled words were still not successfully corrected.

Meanwhile, mBERT achieved a precision score of 0.8279 and a recall score of 0.7375. Although mBERT showed slightly higher accuracy and recall, its F1-score was lower than IndoBERT's. This result differs from the main SPEML evaluation, where mBERT achieved higher overall performance. The difference may be influenced by the characteristics of the external validation dataset, which contains naturally occurring spelling variations, informal expressions, code-mixing, abbreviations, and non-standard morphology. Therefore, the Twitter/X external validation is interpreted as supplementary evidence of model applicability beyond the controlled SPEML corpus, rather than as a replacement for the main SPEML evaluation. These findings confirm that the proposed method can be applied to real-world Minangkabau social media text.

3.6 Limitations and Future Work

Although the proposed method achieved promising results, this study still has several limitations. First, the main evaluation was conducted on SPEML, a controlled spelling-error corpus. Although a small-scale external validation using Twitter/X data was added to improve external validity, SPEML remains the primary benchmark in this study. Therefore, the results may not yet fully represent the broader complexity of real-world Minangkabau texts. Social media data often contain abbreviations, code-mixing, repeated characters, inconsistent punctuation, non-standard morphology, and multiple errors in a single sentence. These characteristics make real-world spelling correction more challenging than evaluation on a controlled corpus.

Second, IndoBERT and mBERT were used without additional fine-tuning. This design makes the proposed approach lightweight and computationally efficient, which is suitable for low-resource languages such as Minangkabau. It also avoids the need for large annotated training data, long training time, and expensive computational resources. However, without task-specific fine-tuning, the models may not fully capture Minangkabau-specific morphological, lexical,

syntactic, and semantic patterns. This limitation is reflected in the lower performance on short words, long words, and deletion-related errors.

Third, the current system is designed to correct one target misspelled word in each sentence. This setting is consistent with the structure of SPEML and the external validation setting used in this study. However, real-world texts may contain more than one spelling error in a sentence, along with informal abbreviations and mixed Indonesian–Minangkabau expressions. Therefore, the current system may still be limited when applied to more complex natural writing conditions.

A comparison with fine-tuned models would provide a clearer view of the trade-off between performance and efficiency. Fine-tuned models may improve correction performance by adapting to Minangkabau-specific spelling patterns and contextual usage. However, they require annotated data, additional training time, computational resources, and hyperparameter tuning. In contrast, the proposed no fine-tuning approach is simpler, more efficient, and more practical for early-stage development in low-resource settings. Therefore, this study positions the no fine-tuning method as a practical baseline rather than as a replacement for fine-tuned models.

Future work should explore fine-tuning IndoBERT, mBERT, or other multilingual models using larger Minangkabau corpora. Further research should also evaluate the method on larger annotated real-world datasets, especially social media texts, to strengthen external validity. In addition, combining contextual embeddings with phonetic rules, N-gram weighting, morphology-aware candidate ranking, and Minangkabau-specific spelling rules may improve performance for difficult cases such as short words, long words, deletion errors, and highly distorted spelling forms. Future studies should also report not only correction performance but also computational cost, training time, and data requirements when comparing no fine-tuning and fine-tuning approaches.

4. Conclusions

This study presented a contextual spell correction approach for the Minangkabau language using IndoBERT and mBERT embeddings without additional fine-tuning. Correction is performed by calculating the cosine similarity between the [MASK] token embedding and the correction candidate embedding generated through lexical selection. The dataset used as the main benchmark is SPEML, which covers various types of linguistic and non-linguistic spelling errors, including insertion, deletion, substitution, transposition, punctuation, real-word, and loanword errors. The results show that mBERT outperformed IndoBERT across almost all SPEML error categories, achieving an average F1-score of 0.83 compared to 0.75 for IndoBERT. Statistical validation using paired t-test and Wilcoxon signed-rank test further confirmed that the performance difference between the two models was

significant. The best performance was achieved in the real-word and loanword error categories, with an F1-score of 1.0. In addition, medium-length words provided the highest correction performance, especially in the insertion_medium category. Conversely, performance declined significantly for words of extreme length and errors of the deletion short or insertion_long type, indicating challenges in handling errors with limited contextual information or complex word forms.

A supplementary external validation using Twitter/X data further showed that the proposed method can be applied to real-world Minangkabau social media text, although the performance pattern differed from the controlled SPEML evaluation. This indicates that real-world informal text introduces additional challenges, such as code-mixing, abbreviations, non-standard spelling forms, inconsistent punctuation, and naturally occurring spelling variations. The findings confirm that contextual embeddings, even without fine-tuning, can serve as an effective and practical foundation for spelling correction in low-resource languages. However, the system still has limitations in handling complex spelling errors, multiple errors in a sentence, and informal real-world text. Although the no fine-tuning approach is computationally efficient and suitable for low-resource settings, future work should compare it directly with fine-tuned models to better understand the trade-off between performance and efficiency. Further development can also integrate phonetic rules, N-gram weighting, morphology-aware candidate ranking, and larger annotated real-world Minangkabau datasets to improve the robustness of the system.

Acknowledgements

The authors would like to express their sincere gratitude to the Ministry of Research, Technology, and Higher Education of the Republic of Indonesia (Kemristekdikti) for providing the research funding. The authors also wish to extend their appreciation to the Institute for Research and Community Service (LPPM) of Universitas Ahmad Dahlan for its role as facilitator, as well as to Universitas Ahmad Dahlan for providing the facilities and academically supportive environment that contributed to the successful completion of this study.

References

- [1] W. B. Yanfi Yanfi, Haryono Soeparno, Reina Setiawan, "Multi-Head Attention Based Bidirectional LSTM for Spelling Error Detection in the Indonesian Language.pdf," 2024, *Ieee Access*. [Online]. Available: <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=10580948>
- [2] F. Koto and I. Koto, "Towards Computational Linguistics in Minangkabau Language: Studies on Sentiment Analysis and Machine Translation." [Online]. Available: <https://id.wikimedia.org/wiki/>
- [3] P. Etoori, M. Chinnakotla, and R. Mamidi, "Automatic spelling correction for resource-scarce languages using deep learning," *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Student Research Workshop*, pp. 146–152, 2018. [Online]. Available: <https://doi.org/10.18653/v1/P18-3021>
- [4] D. Hládek, J. Staš, and M. Pleva, "Survey of automatic spelling correction," *Electronics (Switzerland)*, vol. 9, no. 10, pp. 1–29, 2020. [Online]. Available: <https://doi.org/10.3390/electronics9101670>
- [5] Y. Hu, X. Jing, Y. Ko, and J. T. Rayz, "Misspelling correction with pre-trained contextual language model," *Proceedings of 2020 IEEE 19th International Conference on Cognitive Informatics and Cognitive Computing, ICCI*CC 2020*, pp. 144–149, 2020. [Online]. Available: <https://doi.ieeecomputersociety.org/10.1109/ICCC50026.2020.9450253>
- [6] Ç. Çöltekin, A. S. Doğruöz, and Ö. Çetinoğlu, *Resources for Turkish natural language processing: A critical survey*, vol. 57, no. 1. Springer Netherlands, 2023. [Online]. Available: <https://doi.org/10.1007/s10579-022-09605-4>
- [7] P. A. Selvaraj, M. Jagadeesan, M. Harikrishnan, R. Vijayapriya, and K. Jayasudha, "Survey on Spell Checker for Tamil Language Using Natural Language Processing," *Journal of Pharmaceutical Negative Results*, vol. 13, no. 7, p. 2022. [Online]. Available: <https://www.pnrjournal.com/index.php/home/article/view/1697>
- [8] U. Liyanapathirana, K. Gunasinghe, and G. Dias, "SinSpell: A Comprehensive Spelling Checker for Sinhala," *Accelerating Higher Education Expansion and Development (AHEAD)*, 2021. Doi: [10.48550/arXiv.2107.02983](https://doi.org/10.48550/arXiv.2107.02983)
- [9] Y. Chaabi and F. Ataa Allah, "Amazigh spell checker using Damerau-Levenshtein algorithm and N-gram," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 8, pp. 6116–6124, 2022. doi: <https://doi.org/10.1016/j.jksuci.2021.07.015>
- [10] R. Aziz et al., "Real Word Spelling Error Detection and Correction for Urdu Language," *IEEE Access*, vol. 11, no. September, pp. 100948–100962, 2023. doi: [10.1109/ACCESS.2023.3312730](https://doi.org/10.1109/ACCESS.2023.3312730)
- [11] S. Laaroussi, A. Yousfi, S. L. Aouragh, and S. O. El Alaoui, "Global spelling correction in context using language models: Application to the Arabic Language," *International Journal of Computing and Digital Systems*, vol. 13, no. 1, pp. 361–370, 2023. doi: [10.12785/ijcds/130129](https://doi.org/10.12785/ijcds/130129)
- [12] N. Mukazhanov, Z. Alibiyeva, A. Yerimbetova, A. Kassymova, and N. Alibiyeva, "Development of an augmented Damerau-Levenshtein method for correcting spelling errors in Kazakh texts," *Eastern-European Journal of Enterprise Technologies*, vol. 5, no. 2(125), pp. 23–33, 2023. doi: [10.15587/1729-4061.2023.289187](https://doi.org/10.15587/1729-4061.2023.289187)
- [13] T. Zesch, D. Gardner, and M. Bexte, "Transformer-Based Real-Word Spelling Error Feedback with Configurable Confusion Sets," *Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2025)*, pp. 375–383, 2025. doi: [10.18653/v1/2025-bea-1.29](https://doi.org/10.18653/v1/2025-bea-1.29)
- [14] S. Kasmaiee, S. Kasmaiee, and M. Homayounpour, "Correcting spelling mistakes in Persian texts with rules and deep learning methods," *Sci Rep*, vol. 13, no. 1, pp. 1–21, 2023. doi: <https://doi.org/10.1038/s41598-023-47295-2>
- [15] N. Madi and H. Al-Khalifa, "Error detection for Arabic text using neural sequence labeling," *Applied Sciences (Switzerland)*, vol. 10, no. 15, pp. 1–14, 2020. doi: <https://doi.org/10.3390/app10155279>
- [16] R. Phukan, M. Neog, P. J. Goutom, and N. Baruah, "Automated Spelling Error Detection in Assamese Texts using Deep Learning Approaches," *Procedia Comput Sci*, vol. 235, pp. 1684–1694, 2024. doi: <https://doi.org/10.1016/j.procs.2024.04.159>
- [17] M. Arslan, H. Ghanem, S. Munawar, and C. Cruz, "A Survey on RAG with LLMs," *Procedia Comput Sci*, vol. 246, no. C, pp. 3781–3790, 2024. doi: <https://doi.org/10.1016/j.procs.2024.09.178>
- [18] H. Zhang and M. O. Shafiq, "Survey of transformers and towards ensemble learning using transformers for natural

- language processing,” *J Big Data*, vol. 11, no. 1, 2024. doi: <https://doi.org/10.1186/s40537-023-00842-0>
- [19] C. Bryant, Z. Yuan, M. R. Qorib, H. Cao, H. T. Ng, and T. Briscoe, “Grammatical Error Correction: A Survey of the State of the Art,” *Computational Linguistics*, vol. 49, no. 3, pp. 643–701, 2023. doi: [10.48550/arXiv.2211.05166](https://doi.org/10.48550/arXiv.2211.05166)
- [20] S. Singh and S. Singh, “Systematic review of spell-checkers for highly inflectional languages,” *Artificial Intelligence Review*, vol. 53, p. 4051–4092, 2020. doi: [10.1007/s10462-019-09787-4](https://doi.org/10.1007/s10462-019-09787-4)
- [21] S. Zhang, H. Huang, J. Liu, and H. Li, “Spelling error correction with soft-masked BERT,” *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 882–890, 2020. doi: [10.18653/v1/2020.acl-main.82](https://doi.org/10.18653/v1/2020.acl-main.82)
- [22] C. R. Rahman, M. D. H. Rahman, S. Zakir, M. Rafsan, and M. E. Ali, “BSpell: A CNN-Blended BERT Based Bangla Spell Checker,” *BLP 2023 - 1st Workshop on Bangla Language Processing, Proceedings of the Workshop*, pp. 85–93, 2023. doi: [10.18653/v1/2023.banglalp-1.2](https://doi.org/10.18653/v1/2023.banglalp-1.2)
- [23] T. Ji, H. Yan, and X. Qiu, “SpellBERT: A Lightweight Pretrained Model for Chinese Spelling Check,” *EMNLP 2021 - 2021 Conference on Empirical Methods in Natural Language Processing, Proceedings*, pp. 3544–3551, 2021. doi: [10.18653/v1/2021.emnlp-main.287](https://doi.org/10.18653/v1/2021.emnlp-main.287)
- [25] A. Khattak, M. Z. Asghar, A. Saeed, I. A. Hameed, S. Asif Hassan, and S. Ahmad, “A survey on sentiment analysis in Urdu: A resource-poor language,” *Egyptian Informatics Journal*, vol. 22, no. 1, pp. 53–74, 2021. doi: <https://doi.org/10.1016/j.eij.2020.04.003>
- [26] Y. Yanfi, R. Setiawan, H. Soeparno, and W. Budiharto, “SPECIL: Spell Error Corpus for the Indonesian Language,” *IEEE Access*, vol. 11, pp. 93227–93237, 2023. doi: [10.1109/ACCESS.2023.3307712](https://doi.org/10.1109/ACCESS.2023.3307712)
- [27] H. L. H. S. W. Yanfi, Ford Lumban Gaol, Benfano Soewito, “Spell Checker for the Indonesian Language Extensive Review.pdf,” 2022, *International Journal of Emerging Technology and Advanced Engineering*. doi: [10.46338/ijetae0522_01](https://doi.org/10.46338/ijetae0522_01)
- [28] J. Bakagianni, K. Pouli, M. Gavrilidou, and J. Pavlopoulos, “A systematic survey of natural language processing for the Greek language,” *Patterns*, p. 101313, 2025. doi: <https://doi.org/10.1016/j.patter.2025.101313>
- [29] P. Kittiworapanya, N. Saelek, A. Lertpiya, and T. Chalothorn, “Survey of Query correction for Thai business-oriented information retrieval,” *Proceedings - 2020 15th International Joint Symposium on Artificial Intelligence and Natural Language Processing, iSAI-NLP 2020*, pp. 3–8, 2020. doi: [10.1109/iSAI-NLP51646.2020.9376809](https://doi.org/10.1109/iSAI-NLP51646.2020.9376809)
- [30] M. Goonawardena, A. Kulatunga, R. Wickramasinghe, T. Weerasekara, H. De Silva, and S. Thelijjagoda, “Automated Spelling Checker And Grammatical Error Detection And Correction Model for Sinhala Language,” *Proceedings - International Research Conference on Smart Computing and Systems Engineering, SCSE 2022*, vol. 5, pp. 184–189, 2022. doi: [10.1109/SCSE56529.2022.9905126](https://doi.org/10.1109/SCSE56529.2022.9905126)
- [31] S. Abdulmalek, “Levenstein ’ s Algorithm On English and Arabic .,” 2021 *International Conference of Technology, Science and Administration (ICTSA)*. 2021. doi: [10.1109/ICTSA52017.2021.9406547](https://doi.org/10.1109/ICTSA52017.2021.9406547)
- [32] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, “IndoLEM and IndoBERT: A Benchmark Dataset and Pre-trained Language Model for Indonesian NLP,” *COLING 2020 - 28th International Conference on Computational Linguistics, Proceedings of the Conference*, pp. 757–770, 2020. doi: [10.48550/arXiv.2011.00677](https://doi.org/10.48550/arXiv.2011.00677)
- [33] F. Koto, J. H. Lau, T. Baldwin, “IndoBERTweet: A Pretrained Language Model for Indonesian Twitter with Effective Domain-Specific Vocabulary Initialization,” *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 10660–10668, 2021. doi: [10.18653/v1/2021.emnlp-main.833](https://doi.org/10.18653/v1/2021.emnlp-main.833)
- [34] S. S. Batais, “Consonantal Debuccalization and Deletion in Minangkabau,” *Sage Open*, vol. 14, no. 2, pp. 1–13, 2024. doi: <https://doi.org/10.1177/21582440241240576>
- [35] Z. Guo, Y. Ni, K. Wang, W. Zhu, and G. Xie, “Global Attention Decoder for Chinese Spelling Error Correction,” *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 1419–1428, 2021. doi: [10.18653/v1/2021.findings-acl.122](https://doi.org/10.18653/v1/2021.findings-acl.122)
- [36] and N. K. Vasyl Lytvyn, Petro Pukach , Victoria Vysotska, Myroslava Vovk, “Identification and Correction of Grammatical Errors in Ukrainian Texts Based on Machine Learning Technology,” *Mdpi*, vol. 11, no. 904, 2023. doi: <https://doi.org/10.3390/math11040904>
- [37] M. Dony and S. Yasin, “Koreksi Teks Untuk Kesalahan Penulisan Pada Artikel Berita Menggunakan Model Bahasa,” *JITLIT*, vol. 1, no. 2, pp. 33–38, 2024.
- [38] H. N. Trung, D. T. Ham, T. Huynh, and K. Hoang, “A Combination of BERT and Transformer for Vietnamese Spelling Correction,” *Lecture Notes in Computer Science. Intelligent Information and Database Systems*, pp. 545–558, May 2024. doi: [10.1007/978-3-031-21743-2_43](https://doi.org/10.1007/978-3-031-21743-2_43)
- [39] C. G. Turhan, “Leveraging large language models for spelling correction in Turkish,” *PeerJ Comput Sci*, vol. 11, pp. 1–26, 2025. doi: [10.7717/peerj-cs.2889](https://doi.org/10.7717/peerj-cs.2889)
- [40] D. Soyusiawaty, A. Fadlil, and S. Sunardi “Spell Error Corpus for Minang Language.” [Online]. Available: <https://www.kaggle.com/datasets/dewisoyus/spell-error-corpus-for-minang-language>
- [41] D. Soyusiawaty, A. Fadlil, and S. Sunardi, “Pengembangan Korpus Bahasa Minang pada Spell Error Corpus for Minang Language (SPEML),” *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 11, no. 1, pp. 17–26, 2025. doi: <https://doi.org/10.25077/TEKNOSI.v11i01.2025.17-26>
- [42] C. Zhou et al., “Pretraining Chinese BERT for Detecting Word Insertion and Deletion Errors,” 2022, [Online]. Available: <https://arxiv.org/abs/2204.12052>