LMU

LUDWIG-
MAXIMILIANS-
UNIVERSITÄT
MÜNCHEN

CENTRUM FÜR INFORMATIONS- UND SPRACHVERARBEITUNG
STUDIENGANG COMPUTERLINGUISTIK

Bachelorarbeit

im Studiengang Computerlinguistik

an der Ludwig- Maximilians- Universität München

Fakultät für Sprach- und Literaturwissenschaften

Evaluating the Robustness of Sentiment Analysis for Indonesian Using Behavioral Testing

vorgelegt von
Maria Patricia Viannisa

Betreuer:	Andreas Säuberli
Prüfer:	Prof. Dr. Barbara Plank
Bearbeitungszeitraum:	02. April - 11. Juni 2025

Selbstständigkeitserklärung

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbständig angefertigt, alle Zitate als solche kenntlich gemacht sowie alle benutzten Quellen und Hilfsmittel angegeben habe.

München, den 11. Juni 2025

. . . .
Maria Patricia Viannisa

Erklärung der verwendeten KI-Tools

Ich versichere, dass ich diese Arbeit eigenständig, ohne jede externe Unterstützung, außer den unten aufgeführten Ressourcen, angefertigt habe.

Purpose	**Section(s)**	**Tool**
Grammar check	All document	Grammarly
Translations	All document	Google Translate

München, den 11. Juni 2025

....
Maria Patricia Viannisa

Abstract

English

This thesis aims to evaluate the robustness of the IndoBERT and mBERT models in handling linguistic variations in sentiment analysis for Indonesian text. We also attempt to identify limitations in the IndoBERT and mBERT models of handling linguistic variations through behavioral testing inspired by the CheckList framework by Ribeiro et al. (2020). We created four behavioral tests to accomplish our adjective: sentence insertion, negation handling, orthographical errors, and formality levels. Each of them assesses a different linguistic capability that is present in Indonesian. For our evaluation process, we used one established benchmark dataset by Wilie et al. (2020), the SmSA dataset, and we also manually constructed three formality-level datasets from native Indonesian speakers. The results of our research provide a comprehensive analysis of the performance of the models and the language capabilities in which they fail to analyze correctly. Our key findings indicate that IndoBERT is more robust than mBERT, although both models continue to face challenges in analyzing sentiment in Indonesian, especially in informal contexts.

Deutsch

Ziel dieser Arbeit ist es, die Robustheit der IndoBERT- und mBERT-Modelle im Umgang mit linguistischen Variationen in der Sentimentanalyse indonesischer Texte zu bewerten. Darüber hinaus versuchen wir, die Grenzen der IndoBERT- und mBERT-Modelle im Umgang mit linguistischen Variationen durch Verhaltenstests zu identifizieren, die vom CheckList-Framework von Ribeiro et al. (2020) inspiriert sind. Wir haben vier Verhaltenstests entwickelt, um unser Adjektiv zu erreichen: Satzeinfügung, Negationsbehandlung, Rechtschreibfehler und Formalitätsstufen. Jede von ihnen bewertet eine andere im Indonesischen vorhandene linguistische Fähigkeit. Für unseren Evaluierungsprozess verwendeten wir einen etablierten Benchmark-Datensatz von Wilie et al. (2020), den SmSA-Datensatz, und erstellten manuell drei Formalitätsstufen-Datensätze von indonesischen Muttersprachlern. Die Ergebnisse unserer Forschung liefern eine umfassende Analyse der Leistungsfähigkeit der Modelle und der Sprachfähigkeiten, bei denen sie nicht korrekt analysieren. Unsere wichtigsten Erkenntnisse zeigen, dass IndoBERT robuster ist als mBERT, obwohl beide Modelle weiterhin vor Herausforderungen bei der Analyse der Stimmung auf Indonesisch stehen, insbesondere in informellen Kontexten.

Contents

1 Introduction

Social media and the internet are crucial in people's lives in the modern digital era. Almost everyone has access to these platforms and uses them for communication daily. People can freely express their opinions and emotions using various forms and languages on online communities, forums, and review websites. There are no restrictions on their writing, whether they express positive, neutral, or negative sentiments.

For example, customers can rate and write reviews on websites after visiting a restaurant or staying at a hotel. These ratings and reviews have become a valuable source of information for service providers, businesses, and even governments, influencing product development, marketing strategies, and decision-making processes. They also shape the public image of establishments online, especially since most review websites are publicly accessible.

Rahman et al. (2025) describe sentiment analysis as the process of extracting actual sentiment or underlying meaning from comments and plays a crucial role in understanding public thinking. It has become one of the most widely discussed areas in Natural Language Processing (NLP). Despite recent advancements, several challenges in sentiment analysis remain unresolved. These include individuals' informal writing styles, sarcasm, irony, and language-specific issues (Wankhade et al., 2022), which are even more evident in low-resource languages.

Despite being widely used online and thus having a lot of raw text data available, low-resource languages like Indonesian and Hebrew still lack annotated data collection (Joshi et al., 2020). As a result, NLP systems for these languages are often less accurate and unreliable than those developed for high-resource languages like English and Japanese.

The main problem addressed in this study is the lack of robustness evaluation methods for Indonesian sentiment analysis models. While both IndoBERT and Multilingual BERT (mBERT) have achieved high scores on conventional evaluation metrics, their ability to handle linguistic variations has not been systematically tested. This creates a gap between model performance in controlled experiments and their reliability when deployed in real-world applications. Without evaluating models under diverse linguistic conditions, their generalization capabilities remain unclear.

This research uses a behavioral evaluation approach based on controlled data perturbations to address this gap. We can systematically observe model performance changes by introducing specific linguistic variation types into input data, such as inserting additional sentences and introducing spelling errors. This provides a more targeted way to identify weaknesses that may not be visible through conventional evaluation metrics such as accuracy, precision, recall, and F1-score.

We also introduce three newly constructed Indonesian datasets in this thesis, developed based on a survey conducted via Google Forms with native Indonesian speakers. The respondents were asked to create three sentences conveying an identical meaning but expressed at different levels of formality: formal, semi-formal, and informal. In addition, they were instructed to annotate their sentences with a single sentiment label: positive, neutral, or negative. Using these datasets, we aim to compare the performance of IndoBERT and mBERT across different formality levels while controlling for semantic content.

The research questions addressed in this thesis are as follows:

1. How effectively do the IndoBERT and mBERT models handle linguistic variations in sentiment analysis for Indonesian text?

2. What limitations in the IndoBERT and mBERT models of handling linguistic variations can be identified through behavioral testing?

This section outlines the structure and main content of the thesis. After the introduction in Chapter 1, Chapter 2 discusses the theoretical foundations and related work that form the base of this thesis, including the Transformer model, sentiment analysis, behavioral testing, and characteristics of Indonesian. Next, Chapter 3 explains the methodology used in this thesis, including the experimental tools and frameworks employed, the selected datasets, the language models, the fine-tuning procedures, and the evaluation approaches. Following that, Chapter 4 shows the results of the experiments, while Chapter 5 analyzes the results. Finally, Chapter 6 concludes the thesis and provides suggestions for future work.

2 Theoretical Foundation and Related Work

In this chapter, we explain the theoretical foundations and related work relevant to this thesis and highlight four key areas: the Transformer model, sentiment analysis, behavioral testing, and the characteristics of the Indonesian language.

2.1 The Transformer Model

The Transformer model is a deep learning model that has become very prominent in NLP. Vaswani et al. (2017) introduced the Transformer architecture in their research paper *Attention is All You Need.* Preceding the emergence of the Transformer, previous models like Recurrent Neural Networks (RNNs) (Rumelhart et al., 1986) and Long Short-Term Memory networks (LSTMs) (Hochreiter and Schmidhuber, 1997) use sequential data processing, which read text word-by-word and record information in the order it appears.

The Transformer employs a self-attention mechanism, which can simultaneously look at all words in a sentence. This state-of-the-art mechanism enables more efficient model training and better handling of relations between words, even when they are far apart in the sentence. For example, in the sentence “The cat sat tried to eat the mouse after it died,” the word “it” refers to “the mouse.” The self-attention mechanism helps the Transformer to figure this out by paying attention and giving importance to the right words.

Additionally, the Transformer uses positional encoding to keep track of word order because it does not process the text in order like RNNs and LSTMs. This encoding assigns a number to each word’s vector to indicate its position in the sentence. Then, the Transformer uses layers of self-attention and feedforward networks to process the input multiple times and learn patterns in the text. Figure 2.1 illustrates the Transformer architecture.

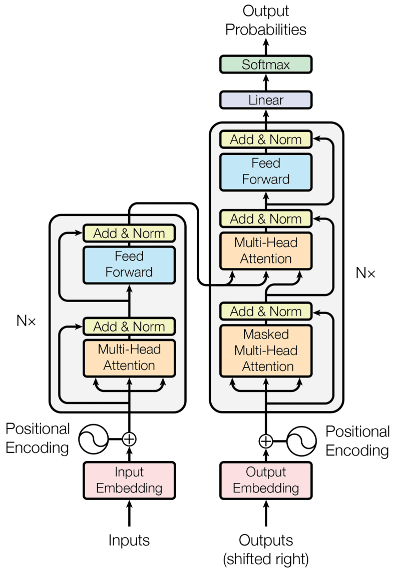

Figure 2.1: The Transformer architecture, as introduced by Vaswani et al. (2017).

After its release, the Transformer becomes the foundation of many state-of-the-art NLP models, such as the Bidirectional Encoder Representations from Transformers (BERT) model by Devlin et al. (2018). BERT looks at a word's left and right context, hence the name bidirectional. This process helps the model understand meaning more accurately. BERT is trained using two tasks: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). In MLM, some words in the sentence are hidden or masked, and the model is tasked to predict these hidden words. In NSP, the model has to determine whether two sentences logically follow one another, considering their contexts. These tasks enable BERT to learn deep contextual representations, which are helpful in many downstream tasks, including sentiment analysis.

The Transformer also becomes the base of A Robustly Optimized BERT Pre-training Approach (RoBERTa) by Liu et al. (2019). Liu et al. (2019) modifies BERT by removing NSP and using dynamic masking. A Lite BERT for Self-supervised Learning of Language Representations (ALBERT) (Lan et al., 2019) is also based on the Transformer. ALBERT reduces its model size by factorizing embeddings. Besides RoBERTa and ALBERT, GPT-2 by Radford et al. (2019) is also based on the Transformer model.

Recent research has introduced contextual pre-trained language models based on the Transformer architecture for languages other than English, particularly BERT-based models. For instance, Martin et al. (2020) presented CamemBERT for French, while de Vries et al. (2019) introduced BERTje for Dutch. This line of research has also been extended to Asian languages, such as PhoBERT for Vietnamese (Nguyen and Nguyen, 2020) and the Japanese Sentence-BERT model (Shibayama and Shinnou, 2021).

Concerning Indonesian NLP, Wilie et al. (2020) introduced the IndoBERT model and benchmark Indonesian datasets in their paper *IndoNLU: Benchmark and Resources for Evaluating Indonesian Natural Language Understanding*. IndoBERT is a monolingual BERT model pre-trained on a large Indonesian corpus that includes both formal and colloquial Indonesian. Besides IndoBERT, Multilingual BERT (mBERT) (Devlin et al., 2018) can also process Indonesian and multiple other languages within a single model. mBERT uses the same architectural design as BERT, but it was simultaneously pre-trained on Wikipedia text from 104 different languages, including Indonesian. Although mBERT's training data includes Indonesian texts, its corpus is not as extensive as the ones used for IndoBERT.

Building on IndoBERT, IndoBERTweet (Koto et al., 2021) trained a BERT variant specifically on Indonesian Twitter data, which helped capture the informal, noisy language common on social media platforms. More recently, NusaBERT (Wongso et al., 2024) addressed the challenge of linguistic diversity within Indonesia by training models on a broader set of Indonesian regional languages. This approach was used to reduce bias and improve performance across the country's many language varieties, reflecting a growing focus on inclusivity in Indonesian NLP.

2.2 Sentiment Analysis

Sentiment analysis, or opinion mining, is the computational study of people's opinions, sentiments, evaluations, appraisals, attitudes, and emotions from written language (Liu, 2012). The most traditional sentiment analysis approach is the lexicon-based approach (Taboada et al., 2011), which relies heavily on lists of words marked with sentiment labels. However, this approach often fails when the words have multiple meanings and thus also have multiple sentiment labels.

Then, machine learning methods like Naive Bayes and Support Vector Machines (SVM) became popular for sentiment analysis (Pang et al., 2002). However, these approaches struggled with complex linguistic features, including sarcasm, context, and domain-specific usage.

With the development of deep learning techniques, there have been rapid improvements in the performance of sentiment analysis (Zhang et al., 2018). Models like Convolutional Neural Networks (CNNs) (O'Shea and Nash, 2015) and RNNs could learn from raw text without manual feature engineering. These models, however, struggled with processing long sentences and understanding word meanings and dependencies with each other.

In recent years, pre-trained Transformer models fine-tuned on labeled sentiment data can effectively generalize across domains and sentence structures (Sun et al., 2019). They can also learn the meaning of each word in its full context. Several studies have applied Transformer-based models to Indonesian sentiment analysis in real-world settings. Geni et al. (2023) analyzed tweets before the 2024 elections using BERT-based models, showing how sentiment analysis can track changes in public sentiments. Another study by Putri et al. (2024) evaluated IndoBERT on political tweets, confirming its usefulness in informal settings. In a related study, Irianti et al. (2025) combined BERT embeddings with SVM classifiers to analyze sentiment around court decisions, demonstrating how hybrid methods can boost performance.

Despite these advancements, sentiment analysis for Indonesian still faces several challenges, particularly the lack of annotated datasets, even though large volumes of raw text data are available from online platforms (Wankhade et al., 2022). This shortage of annotated data limits the ability of supervised learning models to generalize effectively to new or diverse inputs. Additionally, Indonesian sentiment data often originates from social media platforms, where that data is filled with non-standard language, spelling variations, and diverse levels of formality are common. These linguistic variations present significant challenges for existing models, often not trained on such diverse linguistic patterns.

Although IndoBERT (Wilie et al., 2020) and mBERT (Devlin et al., 2018) have performed well on Indonesian datasets according to standard evaluation metrics, their robustness to linguistic variations remains underexplored. Therefore, this thesis aims to address this gap by evaluating the robustness and identifying limitations of the IndoBERT and mBERT models in handling linguistic variations in sentiment analysis for Indonesian text by using behavioral testing inspired by the CheckList framework (Ribeiro et al., 2020).

2.3 Behavioral Testing

Although accuracy metrics are commonly used to evaluate NLP models, they do not always reveal weaknesses in real-world use. Ribeiro et al. (2020) introduced CheckList, a behavioral testing framework that evaluates NLP models on specific linguistic phenomena such as negation handling, orthographical errors, and vocabulary variations. This method uncovers gaps in robustness that conventional evaluation metrics might miss. Rather than relying on random performance samples from test sets, CheckList systematically constructs test cases to probe how models behave in controlled scenarios.

According to Ribeiro et al. (2020), behavioral testing breaks down a model's expected language understanding capabilities into smaller linguistic capabilities. These capabilities are then evaluated using three primary test types:

- Minimum Functionality Test (***MFT***) is a collection of simple examples and labels to check a basic behavior within a capability. This test detects when language models use shortcuts to handle complex inputs without understanding the capability.

- Invariance Test (***INV***) assess model consistency under small label-preserving changes to inputs. The model prediction is expected to stay the same despite the perturbations.

- Directional Expectation Test (***DIR***) checks whether the output changes align with known input changes. The model prediction is expected to change in a certain way.

Table 2.1 shows examples of tested English linguistic capabilities, the behavioral testing types, their descriptions, example test cases, and their expected test behavior from CheckList (Ribeiro et al., 2020).

Language Capability	Test Type	Description	Example Test Case	Expected Behavior
Temporal	***MFT***	Sentiment change over time, present should prevail	I used to hate this airline, but now I like it.	Positive Sentiment
Robustness	***INV***	Swap one character with its neighbor (typo).	@SouthwestAir no thanks to **@SouthwestAir no thakns**	INV. (no change)
Vocab. +POS	***DIR***	Add negative phrases, fails if sentiment goes up by over 0.1	@JetBlue all day. **I abhor you**	A decrease in sentiment

Table 2.1: Examples of CheckList behavioral test types with their descriptions, example test cases, and the expected model behavior from Ribeiro et al. (2020).

This approach has proven helpful for sentiment analysis tasks in multiple languages. For example, Ribeiro et al. (2020) applied the CheckList framework to analyze sentiment models for English and showed that existing language models still fail to handle negation and minor lexical variation despite high performance on conventional evaluation metrics. In the context of multilingualism, K et al. (2022) introduced the Multilingual CheckList, a framework for systematically evaluating multilingual models across languages and linguistic phenomena such as negation and morphological changes, highlighting significant variability in model performance across languages.

Despite the availability of pre-trained language models like IndoBERT and mBERT for Indonesian, their robustness against linguistic variations remains underexplored. Existing evaluations typically rely on conventional metrics such as accuracy or F1-score, failing to reveal how models behave under controlled perturbations. Meanwhile, behavioral testing enables a more nuanced evaluation by systematically probing a model's response to realistic variations in input.

2.4 Characteristics of Indonesian

Indonesian is a language spoken by more than 270 million people. Although its grammar is quite simple compared to other languages, Indonesian also possesses unique linguistic characteristics that influence NLP tasks such as sentiment analysis. The language has relatively regular phonology and opaque orthography, both of which make the tokenization process easier. However, Indonesian morphological richness and lexicographic borrowings present challenges for model generalization.

Indonesian employs both derivational and inflectional morphology. Common affixes include prefixes (me-, ber-, ter-), suffixes (-kan, -i), and circumfixes (ke- -an, per- -an), which change both syntactic category and sentiment polarity (Sneddon, 2003). For instance, the adjective *baik* ('good') can turn into a noun *kebaikan* ('kindness').

Lexicographically, Indonesian also has many variations due to the many words it borrows (Tadmor, 2009). It has words from Dutch (*kantor* = 'office'), Arabic (*dunia* = 'world'), and English (*komputer* = 'computer'). The words introduce synonyms and near-synonyms that may express sentiment differently depending on context.

Another aspect is formality levels. Indonesian culture strongly emphasizes politeness, which is reflected in formal or professional settings and everyday interactions. This includes respecting elders, addressing people of higher social or economic status with care, and treating seniors in educational institutions with notable deference. Politeness is deeply embedded within formal and informal social hierarchies, and this cultural trait is mirrored in the Indonesian vocabulary. The language contains many honorifics and respectful forms of address that vary according to age, gender, and family relationships. Hence, Indonesian has multiple ways of saying 'I,' such as *saya*, *aku*, and even *gua*, depending on how formal the context is (Djenar, 2008). Similarly, there are many ways to say 'you,' like *Anda*, *kamu*, and *loe*. These variations can change the tone of a sentence, which is important for detecting sentiment.

Syntactically, Indonesian usually follows a subject-verb-object (SVO) order for transitive verb (Stack, 2005), like English. However, it often omits the subject or object when understood from context. For example, *Makan siang dulu* ('Have lunch first') does not define the person who is eating, but the listener understands. This can confuse models trained on complete sentence structures.

Negation words also play a critical role in shaping sentence sentiments. In a formal context, Indonesian often uses the negation words *tidak* ('not') and *bukan* ('not'). *Tidak* is typically used preceding verbs, adjectives, and adverbs. For example, *tidak baik* means ('not good'). On the other hand, *bukan* is used to negate nouns or noun phrases. For example *bukan makanan* means ('not food').

Other than *tidak* and *bukan*, there also exists the negation words *tak*, *jangan*, and *belum*. Therefore, incorrectly identifying the scope or type of negation can result in sentiment misclassification. Furthermore, there are also colloquial variants of the negation word *tidak*, such as *enggak*, *nggak*, *gak*, *engga*, *ngga*, *ga*, and *ndak* (Kushartanti et al., 2019). These variants of negation words create challenges for pre-trained models. Since negation can reverse sentiment polarity, accurately modeling its form and function is essential for building robust sentiment analysis systems in Indonesian.

3 Methodology

This chapter describes the methodology and experimental setup used in this thesis. It consists of three main stages: dataset construction, model fine-tuning, and evaluation using conventional metrics and behavioral testing.

First, we selected the IndoNLU SmSA dataset for the fine-tuning and evaluation process in this study. We also created three new datasets with varying levels of formality for additional behavioral testing methods. Then, we fine-tuned two pre-trained masked language models, IndoBERT and mBERT, on the SmSA dataset.

After that, we evaluated the models using conventional classification metrics, including accuracy, precision, recall, and F1-score. Next, we implemented behavioral testing methods adapted from the CheckList framework (Ribeiro et al., 2020). These tests target specific Indonesian linguistic capabilities, including sentence insertion (***DIR*** test), negation handling (***INV*** test), orthographical errors (***INV*** test), and formality levels (***INV*** test). For the ***DIR*** test, we evaluated whether the model's predictions align with our expected labels after the sentence addition process. Meanwhile, for the ***INV*** test, we evaluated whether the model's prediction matches the gold labels of the datasets after the perturbation process for negation handling and orthographical errors. For formality-level behavioral testing, we compare the test results between the three datasets.

The remainder of this chapter describes the experimental tools and frameworks employed, the datasets selected, the language models evaluated, the fine-tuning procedures, and the evaluation approaches applied.

3.1 Experimental Tools and Frameworks

We implemented the experiments in this thesis using Python and several key libraries: Pandas, NLTK, Scikit-learn, Random, PyTorch, and Transformers.

Pandas *Pandas*[1] is a Python library commonly used for data manipulation and analysis. We used Pandas to handle and process the datasets, mainly to read TSV and CSV files, filter content, and manage label comparisons.

NLTK *Natural Language Toolkit (NLTK)*[2] is a Python library for processing human language data. We used NLTK to construct new datasets specially designed for this research, the formality-level datasets. Specifically, we used the NLTK tokenizer package to segment sentences into word tokens and cover them in lowercase to ensure consistency in formatting. Before further analysis and evaluation, this pre-processing step is crucial in normalizing the input across different politeness levels.

Scikit-learn *Scikit-learn*[3] is a widely used Python library for machine learning. We used Scikit-learn to calculate conventional evaluation metrics such as accuracy, precision, recall, and F1-score. These functions are useful for quantitatively comparing the gold labels from the datasets and the model predictions.

[1] https://pandas.pydata.org/
[2] https://www.nltk.org/
[3] https://scikit-learn.org/

Random *Random*[4] is a built-in Python library for generating pseudo-random values. We used Random to create orthographical errors from the original dataset by randomly selecting words and applying character-level modifications, such as swaps or deletions.

PyTorch *PyTorch*[5] is a deep learning framework that supports flexible model development and efficient computation. We used PyTorch to load and run the sentiment classification models, including handling tensors and managing the inference process.

Transformers The *Transformers* library[6] by Hugging Face provides easy access to pre-trained language models. Transformers is the leading resource needed to download the IndoBERT and mBERT language models and tokenizer configurations.

3.2 Datasets

To assess the models' robustness in handling different aspects of language variation, we selected one established benchmark dataset, the IndoNLU SmSA dataset. In addition to the SmSA dataset, we created three original custom datasets designed for Indonesian robustness evaluation in this thesis. Each of these datasets contains 30 sentences with identical meanings; however, they are rewritten in different levels of formality.

3.2.1 IndoNLU SmSA Dataset

SmSA dataset, first introduced in the IndoNLU benchmark (Wilie et al., 2020), is a sentence-level sentiment analysis dataset consisting of comments and reviews in Indonesian that are collected from various internet sources. The text sources were crawled and subsequently annotated manually by several Indonesian linguists. The sentences primarily reflect informal, colloquial Indonesian, capturing the linguistic characteristics of everyday digital communication. Hence, they are abundant with nonstandard Indonesian grammar, slang usage, additional Indonesian regional vocabularies, and lexical variations.

The dataset is split into three subsets: training, validation, and test sets, containing 11,000, 1,260, and 500 sentences, respectively. Each sentence is annotated with a single sentiment label from one of three sentiment classes: positive, neutral, or negative. These labels reflect the emotional tone expressed in the sentence.

While all three subsets are later used in fine-tuning, only the test subset is used for conventional evaluation metrics and behavioral testing. To facilitate easier analysis and model processing, we converted the sentiment labels of the SmSA test subset into numerical values: positive (0), neutral (1), and negative (2).

Example sentences of each sentiment label, their original labels in the test subset, and the new labels after modification are explained in Table 3.1.

[4] https://docs.python.org/3/library/random.html

[5] https://pytorch.org/

[6] https://huggingface.co/docs/transformers/en/index

Sentence	Original Label	Numerical Label
tiket.com sangat terpercaya . harga murah . lancar , praktis , efektif , dan efisien . ('tiket.com is very trustworthy . cheap price . smooth , practical , effective , and efficient .')	positive	0
indonesia itu ada di benua asia . ('indonesia is on the asian continent .')	neutral	1
takdir politik ahy belum bisa ikut kontestasi pilpres 2019 - 2024 . walaupun ahy legowo , kami tetap kecewa . ('ahy's political destiny is that he has not been able to participate in the 2019 - 2024 presidential election . even though ahy is resigned , we are still disappointed .')	negative	2

Table 3.1: Examples of sentences from the test subset of SmSA with their corresponding original and modified gold labels.

3.2.2 Formality-Level Datasets Construction

Given Indonesia's unique and important formality levels, we constructed three parallel datasets designed exclusively for Indonesian robustness evaluation in this thesis. The datasets represent formal, semi-formal, and informal vocabulary varieties of Indonesian, where each set contains sentences with equivalent meaning but expressed in different levels of formality. The parallel structure of these datasets allows direct comparison of model performance across different formality levels while controlling for semantic content.

We conducted a survey using Google Forms, asking native Indonesian speakers to write sentences in different formality styles. Each respondent was asked to write one formal, one semi-formal, and one informal sentence, with all three expressing the same sentiment, and to assign a single sentiment label to the sentences they created.

In the description of the Google Forms, we first introduced the topic and the objective of our research. We provided a brief explanation of sentiment analysis and an example of a negatively labeled sentence: *Aku sangat benci Ibuku yang memanjakan adikku hingga ia dewasa.* ('I really hate my mother who spoiled my little brother until he grew up.'). Then, we described the different levels of formalities, including their example uses. We also included a disclaimer stating that the examples in the Google Forms are intended solely for academic purposes within this thesis and do not reflect any real political views, real events, or personal opinions.

We presented six example sentences to guide respondents in filling out the form: three expressing negative sentiment and three expressing positive sentiment, each representing a different level of formality. Table 3.2 explains further details and examples given to the annotators in Google Forms.

Formality	Description	Positive Sentence Example	Negative Sentence Example
Formal	Commonly found in news articles or official statements. Typically uses “saya” for the first person and “anda” for the second person.	Pemerintah menunjukkan kinerja yang baik dalam meningkatkan kualitas layanan kesehatan masyarakat. (‘The government has shown good performance in improving the quality of public health services.’)	Presiden menyampaikan bahwa pemerintah berkomitmen untuk menurunkan angka pengangguran melalui program-program strategis. (‘The President said that the government is committed to reducing unemployment rates through strategic programs.’)
Semi-Formal	Example use in group chats with family, extended family (including older relatives), or workplace WhatsApp groups. The tone is polite and conversational but not too formal. Typically uses “aku” for the first person and “kamu” for the second person.	Menurutku, pemerintah udah lumayan oke sekarang soal layanan kesehatan. (‘In my opinion, the government is pretty good now regarding health services.’)	Pak Presiden bilang pemerintah bakal fokus ngurangin pengangguran lewat beberapa program baru. (‘The President said the government will focus on reducing unemployment through several new programs.’)
Informal	Typically used when texting close friends of the same peer group; very casual, sometimes emotional, and may include slang, abbreviations, or mild curses. For this style, the first and second-person pronouns vary depending on the region in Indonesia.	Anjir, sekarang rumah sakit makin bagus. Pemerintah akhirnya ngasih perubahan yang bener juga nih! (‘Damn, now the hospital is getting better. The government is finally making real changes!’)	Pemerintah omdo mulu, pengangguran makin numpuk aja. Goblok banget sumpah. (‘The government is just ranting, unemployment is piling up. They’re so stupid, I swear.’)

Table 3.2: Descriptions of different formality levels, their descriptions, and their sentence examples.

From the total of 37 responses collected, we extracted 30 sentences per dataset, evenly divided into 10 sentences for each sentiment class as labeled by the respondents. The balanced number of sentences per sentiment class within each dataset helps to avoid bias during evaluation. Each sentence is manually assigned a sentiment label represented in numbers: positive (0), neutral (1), or negative (2).

To ensure data quality, we manually reviewed all survey responses. First, we verified that the three sentences provided in each response conveyed the same meaning to support a consistent comparison across different formality levels. Next, we conducted a follow-up review of the sentiment labels to confirm their accuracy despite being initially assigned by native Indonesian speakers. We discovered that some formal sentences, despite being labeled as positive or negative by the respondents, actually convey a neutral sentiment; these labels instead more accurately reflect the sentiment expressed in the corresponding semi-formal or informal sentences. To address this issue, we inserted a phrase to adjust the sentiment of the sentences in line with the labels provided by the respondents. For example, adding *Saya mendukung ...* (‘I support ...’) to shift the sentiment toward positive, and adding *Saya ragu ...* (‘I doubt ...’) to shift the sentiment toward negative.

We exported the Google Sheets of responses as a CSV file. Then, we used the Pandas library to process the CSV file, including reading, modifying, and saving the datasets. Using the NLTK library, we tokenized the sentences and converted all text to lowercase to facilitate better processing and analysis. Minimal additional pre-processing was applied to preserve natural linguistic variations. This process resulted in three CSV files: one containing formal sentences with corresponding sentiment labels, one with semi-formal sentences and labels, and one with informal sentences and labels.

Due to resource and time constraints, the datasets are relatively small. They may not fully capture the diversity of Indonesian dialects and social contexts, which should be considered when interpreting the results.

Example sentences for each dataset are in Table 3.3.

Formal	Semi-Formal	Informal	English Translation	Gold Label
Program-program pemerintah dalam mensubsidi beasiswa kepada mahasiswa Indonesia yang berkeinginan untuk melanjutkan pendidikan di sekolah tinggi di luar negeri sangatlah baik.	Program pemerintahnya untuk subsidi beasiswa ke anak” Indonesia yg pengen kuliah di luar sih bagus ya	y sbnrny program2 pemerintah buat subsidi beasiswa buat anak2 yg mo kuliah di luar sih bgs bgt y	Government programs in subsidizing scholarships for Indonesian students who want to continue their education at universities abroad are very good.	positive (0)
Menurut saya, Anda sangatlah cantik.	Sayang kamu cantik banget sihhh	lo cantik banget dah gila bisa ya secakep itu	I think you are very beautiful.	positive (0)
Saat ini media sosial dikejutkan dengan grup facebook bernama “Fantasi Sedarah” yang mengarah pada tindakan asusila terkait ketertarikan seksual dengan anggota keluarganya.	Baru-baru ini medsos lagi dikejutkan dengan grup facebook “Fantasi Sedarah” yang mengarah pada tindakan pelecehan pada anggota keluarga.	Blkgn kasus ttg grup fb fantasi sedarah lagi bikin heboh.	Currently, social media is shocked by a Facebook group called “Fantasi Sedarah” which refers to immoral acts related to sexual attraction to family members.	neutral (1)
Mobil memiliki empat roda.	Rodanya mobil ada empat.	Mobil tu rodanya empat	The car has four wheels.	neutral (1)
Saya setuju bahwa kurangnya akses masyarakat terhadap ruang terbuka hijau adalah salah satu indikator kegagalan perencanaan tata ruang kota.	Orang bisa nilai perencanaan kota jelek atau engga itu dari ada gaknya ruang terbuka hijau.	Njirr, mall di mana-mana tapi taman kota minim! Kureng bgt sih ini pas ngeremcanain bangun kotanya	Lack of public access to green open spaces is one indicator of the failure of urban spatial planning.	negative (2)
Teman-teman terdekat Anda, seperti saya, yang peduli dengan Anda, juga kesal.	Temen kamu yang peduli sama kamu kayak aku juga kesel.	Tmn2 deket lo aja yg PEDULI sm lo (aka GUE) aja kesel	Your closest friends, like me, who care about you, are also upset.	negative (2)

Table 3.3: Example sentences from the formality-level datasets with corresponding sentiment labels.

3.3 Language Models

We evaluated two language models: IndoBERT and mBERT. We applied the same conventional metrics and behavioral testing approaches to both models to analyze their vulnerabilities and performance differences.

3.3.1 IndoBERT

IndoBERT is a monolingual BERT-based language model pre-trained exclusively on large-scale pre-processed Indonesian text data, the Indo4B Dataset. The dataset consists of around 4 billion words, with around 250 million sentences, both in formal and colloquial Indonesian. The dataset is compiled from 12 different datasets. Both the model and the dataset were first introduced by the IndoNLU team (Wilie et al., 2020), who aimed to advance Indonesian NLP research further.

In this thesis, we used the **indobenchmark/indobert-base-p1** model, which is a base-sized model with 124.5 million parameters, 12 layers, 12 attention heads, an embedding size of 768, a hidden size of 768, a Feedforward Network size of 3072, and contextualized pre-training embedding type.

3.3.2 Multilingual BERT

Multilingual BERT (mBERT) is a multilingual version of the original BERT model released by Devlin et al. (2018). It is pre-trained on Wikipedia texts from 104 different languages, including Indonesian. Therefore, mBERT can process different languages in one model without being trained separately or given a customized vocabulary for each language.

The architecture of mBERT is a multilayer bidirectional Transformer encoder consisting of 167.4 million parameters, 12 layers, 12 attention heads, an embedding size of 768, a hidden size of 768, a Feedforward Network size of 3072, and contextualized pre-training embedding type. In this thesis, we used the **google-bert/bert-base-multilingual-cased** model.

Table 3.4 shows the detailed comparison between IndoBERT and mBERT.

Property	IndoBERT	mBERT
Model Type	Monolingual BERT-based	Multilingual BERT-based
Pre-training Data	Indo4B Dataset (250M Indonesian sentences)	Wikipedia texts from 104 languages
Vocabulary Size	30,000 WordPieces (Indonesian-focused)	119,547 WordPieces (shared across 104 languages)
Training Objective	Masked Language Modeling (MLM)	Masked Language Modeling (MLM)
Number of Layers	12	12
Hidden Size	768	768
Attention Heads	12	12
Parameter Count	124.5 million	167.4 million

Table 3.4: Comparison of key architectural and training properties of IndoBERT and mBERT.

3.4 Fine-tuning

Before evaluating the IndoBERT and mBERT models using conventional metrics or behavioral testing, we first fine-tuned both models on the SmSA dataset. We used all three SmSA subsets originally from the IndoNLU benchmark: training, validation, and test subsets. We conducted the fine-tuning process on Google Colab using a T4 GPU runtime to speed up training. Fine-tuning is necessary to adapt the pre-trained models, which were pre-trained initially on the masked language modeling task, to the specific sentiment classification task in Indonesian.

We followed the SmSA fine-tuning tutorial and example provided in the IndoNLU GitHub repository[7]. The training was conducted over five epochs with a 3×10^{-6} learning rate. We

[7] https://github.com/IndoNLP/indonlu/blob/master/examples/finetune_smsa.ipynb

monitored the training loss and the key classification metrics during each epoch, including accuracy, precision, recall, and F1-score. After each epoch, we evaluated the model on the validation set to track generalization performance. The Tables 3.5 and 3.6 below summarizes the epoch-wise training and validation results.

Epoch	Train Loss	Valid Loss	Accuracy	Precision	Recall	F1
1	0.3349	0.1863	0.93	0.91	0.89	0.90
2	0.1579	0.1799	0.94	0.93	0.89	0.90
3	0.1195	0.1886	0.93	0.92	0.91	0.91
4	0.0929	0.1710	0.94	0.93	0.91	0.92
5	0.0657	0.1771	0.94	0.92	0.92	0.92

Table 3.5: Epoch-wise training and validation performance metrics for IndoBERT on the SmSA dataset.

Epoch	Train Loss	Valid Loss	Accuracy	Precision	Recall	F1
1	0.6222	0.4488	0.83	0.81	0.76	0.78
2	0.3909	0.3596	0.86	0.85	0.80	0.82
3	0.3113	0.3072	0.88	0.85	0.83	0.84
4	0.2483	0.2931	0.89	0.86	0.84	0.85
5	0.2054	0.2985	0.90	0.87	0.84	0.86

Table 3.6: Epoch-wise training and validation performance metrics for mBERT on the SmSA dataset.

We manually created three original test sentences for each sentiment class (positive, neutral, and negative) to check whether the model had learned to recognize sentiment correctly. We chose to create these sentences manually to evaluate the model's capability in classifying new sentences that are not present in its training data.

We compared the results between the baseline and fine-tuned models. Firstly, we tested the sentences on the baseline model and documented the results. Then, we fine-tuned the model. Lastly, we ran the test sentences on the fine-tuned model and documented the results. These manual tests aim to verify the model's ability to generalize and correctly assign sentiment labels to statements beyond the test set.

From this fine-tuning process and additional sentences, we observed that the baseline IndoBERT model labeled all of the test sentences as neutral, with the highest confidence score only at 42.373%. This shows that the baseline IndoBERT model only managed to classify two sentences with the correct label: those with neutral labels. Meanwhile, the baseline mBERT model labeled five sentences as negative and only one as positive, with the highest confidence score only at 37.718%. The baseline mBERT model managed to label three sentences correctly: two negative sentences and one positive sentence.

In contrast, the fine-tuned IndoBERT model correctly labeled all six sentences, with the lowest confidence score at 98.848% and the highest at 99.848%. However, the fine-tuned mBERT model still classified one sentence incorrectly. mBERT misclassified a sentence with negative sentiment as positive, with a confidence score of 57.712%. Besides this sentence, mBERT classified the other five sentences accurately, with the lowest confidence score at 70.266% and the highest at 98.021%. This shows that fine-tuning the model on a sentiment analysis task is necessary before the evaluation process.

The sentences and the results are shown in Table 3.7.

Sentence	Gold Label	IndoBERT Baseline	IndoBERT Fine-tuned	mBERT Baseline	mBERT Fine-tuned
Aku sangat mencintai dirinya yang sangat sempurna untukku. ('I really love him who is so perfect for me.')	positive (100%)	neutral (38.501%)	positive (99.72%)	negative (37.058%)	positive (98.021%)
Dia sangat hebat karena sudah sangat pintar dan mandiri sejak kecil. ('He is very great because he has been very smart and independent since he was little.')	positive (100%)	neutral (40.196%)	positive (99.805%)	positive (35.362%)	positive (97.975%)
Ayah memakan bubur ayam untuk sarapan tadi pagi. ('Dad ate chicken porridge for breakfast this morning.')	neutral (100%)	neutral (42.373%)	neutral (98.848%)	negative (36.917%)	neutral (70.266%)
Setiap hari, adikku berangkat sekolah pukul 7 pagi. ('Every day, my little brother goes to school at 7 am.')	neutral (100%)	neutral (38.484%)	neutral (99.226%)	negative (37.718%)	neutral (93.122%)
Temanku sangat benci ibunya yang tidak pernah ada di rumah. ('My friend really hates her mother who is never home.')	negative (100%)	neutral (39.421%)	negative (99.734%)	negative (36.669%)	positive (57.712%)
Budi anak yang sangat malas dan tidak berguna. ('Budi is a very lazy and useless child.')	negative (100%)	neutral (39.066%)	negative (99.804%)	negative (35.619%)	negative (96.028%)

Table 3.7: Fine-tuning test sentences and their results on baseline and fine-tuned IndoBERT and mBERT.

After fine-tuning the model, we saved the model's weights using PyTorch. Using Hugging Face, we also saved the tokenizer, which is essential for pre-processing input text, such as during training. Saving the tokenizer is critical to ensure consistent tokenization behavior during inference or further evaluation. The trained model files (PTH) and the tokenizer directories (compressed into a ZIP file) were downloaded from Google Colab to store everything locally. We ran this fine-tuning process twice: once for IndoBERT and once for mBERT.

3.5 `predict` Function

We implemented a prediction function to perform sentiment classification in all our evaluation approaches on the datasets. The `predict` function takes a list of input texts and processes them using the model's tokenizer. Padding and truncation are enabled to ensure uniform sequence lengths. The *return_tensors="pt"* argument transforms the tokenized outputs into PyTorch tensors, which is required to ensure that the tokenized outputs are compatible with the inference process of the language model implemented in PyTorch.

Since we used the `predict` function solely to predict the sentiment labels and not to train the model, we used the Python context manager *torch.no_grad* to turn off gradient computations, which require more memory and slower computation process. Within this context manager, we passed the tokenized inputs to the model. Then, the model returns output logits, which are unnormalized scores used to determine the final prediction. Using *torch.argmax* across the class labels, the function chooses the class with the highest logit score as its final prediction, resulting in discrete sentiment labels. Then, the resulting tensor is converted to a standard Python list and returned.

Listing 3.1 presents the implementation of the `predict` function used to generate model predictions.

```python
def predict(texts):
    inputs = tokenizer(texts, padding=True, truncation=True, return_tensors="pt")
    with torch.no_grad():
        outputs = model(**inputs)
        logits = outputs.logits
        predictions = torch.argmax(logits, dim=1).tolist()
    return predictions
```

Listing 3.1: The `predict` function

3.6 Evaluation Approaches

We utilized two evaluation approaches to assess the IndoBERT and mBRT models: standard conventional metrics and behavioral testing. While standard metrics such as accuracy and F1-score provide a quantitative measure of the overall model performance, behavioral testing offers insights into the model's robustness and reliability under specific linguistic perturbations.

For all tests, we started by loading the dataset(s) relevant to the evaluation. Then, we initialized the tokenizer and the pre-trained sequence classification model. We loaded the model weights from a fine-tuned checkpoint and set the model to evaluation mode. After that, we executed the `predict` function to obtain the model's sentiment predictions. We refer to this process as the ***prediction process*** further in this thesis.

3.6.1 Conventional Metrics

We evaluated the models using four conventional metrics: accuracy, precision, recall, and F1-score. These metrics were chosen to provide a comprehensive assessment of the model performance: accuracy is the number of correct classifications over the entire test set, precision is the ratio of the true positive over all the positives observed, recall is the ability of a model to identify the true positive correctly, and F1-score is the weighted average of precision and recall (Obi, 2023).

To start the evaluation process, we imported the `accuracy_score`, `precision_score`, `recall_score`, and `f1_score` functions from Scikit-learn. We used only the SmSA test subset for conventional evaluation metrics. Then, we executed the *prediction process*. Finally, we compared the predicted labels with the gold labels from the subset to calculate the accuracy, precision, recall, and macro-averaged F1-score, all presented as percentages.

Listing 3.2 displays the Python script that compares gold labels with predicted labels in conventional evaluation metrics.

```python
df["predicted_label"] = predict(df["sentence"].tolist())

accuracy = accuracy_score(df["gold_label"], df["predicted_label"])
precision = precision_score(df["gold_label"], df["predicted_label"], average="weighted")
recall = recall_score(df["gold_label"], df["predicted_label"], average="weighted")
f1 = f1_score(df["gold_label"], df["predicted_label"], average="macro")

print(f"Accuracy:  {accuracy * 100:.2f}%")
print(f"Precision: {precision * 100:.2f}%")
print(f"Recall:    {recall * 100:.2f}%")
print(f"F1 Score:  {f1 * 100:.2f}%")
```

Listing 3.2: The Python script used to compare the labels for conventional evaluation metrics.

3.6.2 Behavioral Testing

In addition to conventional evaluation metrics, we applied a behavioral testing approach inspired by the CheckList framework proposed by Ribeiro et al. (2020). Considering that the original CheckList tool is designed primarily for English, we adapted the underlying methodology of behavioral testing to suit the Indonesian language. This approach enables a comparative analysis of the linguistic sensitivity and generalization capabilities of IndoBERT and mBERT beyond conventional evaluation metrics.

We manually constructed Indonesian behavioral test cases based on two out of three CheckList's core test types: Directional Expectation (***DIR***) and Invariance (***INV***), focusing on four specific test types: sentence insertion, negation handling, orthographical errors, and formality levels. ***DIR*** tests assess whether the model appropriately updates its prediction when a semantically meaningful change is introduced to reverse the sentiment, such as by appending sentences with a clear positive, neutral, or negative tone. Meanwhile, ***INV*** tests evaluate model consistency by introducing minor lexical or orthographic changes, such as vocabulary variation, while keeping the expected sentiment label unchanged. We chose only two testing types from CheckList because Indonesian lacks annotated datasets required for ***MFT***.

DIR Test: Sentence Insertion

This test evaluates the model's ability to correctly predict the expected sentiment label when an additional sentence is appended at the end of the original sentences in the dataset. It assesses whether the model's prediction aligns with the expected label as influenced by the additional sentence.

We used the SmSA test subset and created two additional sentences: one representing the negative label and one representing the positive label. Using the `sentence_addition` function, we appended each sentence separately to the original sentences, resulting in two distinct datasets. Each dataset consists of 500 original sentences, with one specific additional sentence appended to all entries. This process produced two separate CSV files, each corresponding to one of the appended sentences. Table 3.8 shows the additional sentences and their corresponding sentiments.

Additional Sentence	**Sentiment Label of the Additional Sentence**	**Example Appended Sentence from Dataset**
Saya cinta matematika. ('I love mathematics.')	positive	oke . saya akan segera menuju ke cabang bca terdekat . saya cinta matematika . ('okay . i will go to the nearest bca branch soon . i love mathematics .')
Saya benci matematika. ('I hate mathematics.')	negative	oke . saya akan segera menuju ke cabang bca terdekat . saya benci matematika . ('okay . i will go to the nearest bca branch soon . i hate mathematics .')

Table 3.8: Sentences appended in the SmSA test subset in the ***DIR*** test for sentence insertion.

We loaded each dataset featuring sentences appended with the additional sentences. Then, we executed the *prediction process*. Next, we compared the predicted labels to the gold labels for every sentence and recorded whether the prediction matched the expected behavior. Based on this, we calculated the total number of samples, the number of incorrect predictions, and the failure rate, which shows the proportion of unexpected predictions.

We created two separate Python scripts to evaluate each of the sentiment classes. In each script, we loaded the dataset containing the appended sentences. Then, we executed the *prediction process*. We created the `expected_label` function to define a set of expected prediction labels for each original gold label. For instance, if the original gold sentiment is neutral and a sentence with negative sentiment is inserted, the only accepted prediction label is negative. A detailed explanation of the gold label and its expected labels can be found in Table 3.9.

Sentiment of the Additional Sentence	Gold Label	Expected Label(s)
positive	positive (0)	positive (0)
	neutral (1)	positive (0)
	negative (2)	positive (0), neutral (1)
negative	positive (0)	neutral (1), negative (2)
	neutral (1)	negative (2)
	negative (2)	negative (2)

Table 3.9: Appended sentences by sentiment class with their gold labels and corresponding expected labels used to evaluate model behavior in the ***DIR*** test for sentence insertion.

Next, we compared the predicted labels to the expected labels for every sentence and recorded whether the prediction matched the expected behavior. Based on this, we calculated the total number of samples, the number of incorrect predictions, and the failure rate, which shows the proportion of unexpected predictions. Listing 3.3 displays the Python script that evaluates whether model predictions align with expected outcomes after inserting a positive sentence into the input, and Listing 3.4 displays the Python script used for evaluation after inserting a negative sentence into the input.

```python
df["predicted_label"] = predict(df["modified_sentence"].tolist())

def expected_label(gold_label):
    if gold_label == 0: # positive
        return {0} # positive
    elif gold_label == 1: # neutral
        return {0} # positive
    elif gold_label == 2: # negative
        return {0, 1} # positive, neutral
    else:
        return set()

df["expected_label"] = df["gold_label"].apply(expected_label)

df["label_match"] = df.apply(lambda row: row["predicted_label"] in row["expected_label"], axis=1)

total = len(df)
correct_behavior = df["label_match"].sum()
failure_rate = ((total - correct_behavior) / total) * 100

print(f"DIR Test Positive Sentence Results ({model_name}):")
print(f"Total samples: {total}")
print(f"Failures (unexpected behavior): {total - correct_behavior}")
print(f"Failure rate: {failure_rate:.2f}%")

if save_path:
    df[["sentence", "gold_label", "expected_label", "predicted_label", "label_match"]].to_csv(save_path, index=False)
```

Listing 3.3: The Python script used to compare the labels for positive sentence insertion tests.

```python
df["predicted_label"] = predict(df["modified_sentence"].tolist())

def expected_label(gold_label):
    if gold_label == 0: # positive
        return {1, 2} # neutral, negative
    elif gold_label == 1: # neutral
        return {2} # negative
    elif gold_label == 2: # negative
        return {2} # negative
    else:
        return set()

df["expected_label"] = df["gold_label"].apply(expected_label)

df["label_match"] = df.apply(lambda row: row["predicted_label"] in row["expected_label"], axis=1)

total = len(df)
correct_behavior = df["label_match"].sum()
failure_rate = ((total - correct_behavior) / total) * 100

print(f"DIR Test Negative Sentence Results ({model_name}):")
print(f"Total samples: {total}")
print(f"Failures (unexpected behavior): {total - correct_behavior}")
print(f"Failure rate: {failure_rate:.2f}%")

if save_path:
    df[["sentence", "gold_label", "expected_label", "predicted_label", "label_match"]].to_csv(save_path, index=False)
```

Listing 3.4: The Python script used to compare the labels for negative sentence insertion tests.

The testing process generated four CSV files: two for IndoBERT and two for mBERT. Each file corresponds to one of the appended sentences in the dataset, containing detailed results such as the modified sentences, gold labels, expected labels, predicted labels, and whether the prediction matched the expected labels. Table 3.10 shows example appended sentences from the output file of the negative sentence insertion evaluation using the mBERT model, including their corresponding gold labels, predicted labels, and label match status.

Appended Sentence	Gold Label	Expected Label(s)	Predicted Label	Label Match
melihat komen nya 90 % negatif jadi pikir-pikir buat mencari tiket kereta di sini . mending aplikasi yang lain saja yang sudah terbukti bagus bertahun-bertahun . **saya benci matematika . saya benci matematika .** ('seeing the comments 90% negative so think twice about looking for train tickets here. better to use another application that has been proven to be good for years. **i hate mathematics .**')	negative (2)	negative (2)	positive (0)	False
ingin rekomendasi saja buat yang tertarik kuliner bandung , bacang yang ada di jalan naripan sebelah hotel ibis styles ibis yang warna hijau . sungguh , seenak itu ! buka dari jam 5 sore sampai jam 3 dinihari kalau belum sekali . harga 8 ribu . cocok buat ajak orang tersayang nya . **saya benci matematika .** ('i just want to recommend for those who are interested in bandung culinary , the bacang on jalan naripan next to the green ibis styles hotel . really , it's that delicious ! open from 5 pm to 3 am if you haven't tried it once . price 8 thousand . suitable for taking your loved ones . **i hate mathematics .**')	positive (0)	neutral (1) negative (2)	positive (0)	False
produk lokal memang cetek . saya benci matematika . **saya benci matematika .** ('local products are shallow . **i hate mathematics .**')	negative (2)	negative (2)	positive (0)	False

Table 3.10: Examples of appended sentences and their corresponding gold labels, expected labels, predicted labels, and label match status from the negative sentence insertion tests on mBERT.

INV Test: Negation Handling

This test examines the model's ability to handle various forms of negation in Indonesian. It evaluates whether the model can maintain the original sentiment correctly when different Indonesian negation words are used.

We used the SmSA test subset and extracted all sentences with the word *tidak* ('no') using the `negation_extraction` function, resulting in a total of 188 sentences. Then, we substituted *tidak* with the following negation words: *nggak* and *gak*. This procedure generated two distinct CSV files, each representing a different negation word. Listing 3.6 shows the Python script that extracts sentences containing the word *tidak* from the SmSA test subset. Further details about the differences in Indonesian negation words can be found in Table 3.11.

```
def negation_extraction(new_negation, save_path):

    df = pd.read_csv("path/to/smsa_dataset_changed_label.csv")

    df_filtered = df[df["sentence"].str.contains(r"\btidak\b", case=False, regex=True)]

    df_filtered["sentence"] = df_filtered["sentence"].str.replace(r"\btidak\b", new_negation, case=False, regex=True)

    if save_path:
        df_filtered.to_csv(save_path, index=False)
```

Listing 3.5: The `negation_extraction` function

Negation Word	English Translation	Description	Example from Dataset
tidak	no	The most formal and standard form of negation in Indonesian, typically used in written and formal spoken contexts.	keleveru terbaik memang , dari kualitas , kuantitas , pelayanan pun terbaik . **tidak** salah pilih aku . ('keleveru is the best indeed , in terms of quality , quantity , and service is also great . i did not choose wrongly')
nggak	no	Informal form of *tidak*, commonly used in everyday spoken Indonesian.	keleveru terbaik memang , dari kualitas , kuantitas , pelayanan pun terbaik . **nggak** salah pilih aku .
gak	no	A shortened version of *nggak*, frequently used in casual conversation, texting, or informal settings.	keleveru terbaik memang , dari kualitas , kuantitas , pelayanan pun terbaik . **gak** salah pilih aku .

Table 3.11: List of Indonesian negation words used in the ***INV*** tests for negation handling.

We loaded each dataset featuring sentences with the negation word *tidak* replaced by its variants. Then, we executed the *prediction process*. We directly compared the predicted labels to the original gold labels to check if the model consistently preserves the correct sentiment label despite the negation variation. We recorded whether each prediction matched the gold label and then calculated the total number of samples, the number of incorrect predictions, and the failure rate, which reflects the proportion of unexpected model outputs. Listing 3.6 shows the Python that compares gold labels with predicted labels in negation handling tests.

```
df["predicted_label"] = predict(df["sentence"].tolist())
df["label_match"] = df["gold_label"] == df["predicted_label"]

total = len(df)
correct_behavior = df["label_match"].sum()
failure_rate = ((total - correct_behavior) / total) * 100

print(f"Behavioral Test Results ({test_name}):")
print(f"Total samples: {total}")
print(f"Failures (unexpected behavior): {total - correct_behavior}")
print(f"Failure rate: {failure_rate:.2f}%")

if save_path:
    df[["sentence", "gold_label", "predicted_label", "label_match"]].
        to_csv(save_path, index=False)
```

Listing 3.6: The Python script used to compare the labels for negation handling tests.

The testing generated four CSV files: two for IndoBERT and two for mBERT. Each file represents a different negation substitution applied in the dataset and includes detailed information such as the modified sentences, gold labels, predicted labels, and whether the predictions aligned with the expected behavior. 3.12 shows example modified sentences from the output file of the *nggak* negation word substitution evaluation using the IndoBERT model, including their corresponding gold labels, predicted labels, and label match status.

Modified Sentence	Gold Label	Predicted Label	Label Match
xiaomi ram - nya mantap , baterai awet juga . kamera depan nya juga lumayan . **nggak** menyesal beli ini . ('xiaomi ram is steady , battery is also long lasting . the front camera is also quite good . no regrets buying this .')	positive (0)	positive (0)	True
kalau **nggak** suka gaperlu didukung mas , dukung saja yang menurut mas daus itu bisa bawa indonesia jauh lebih baik . yang penting kita harus menomorsatukan asas etika . ('if you don't like it , no need to support it , just support what according to mas daus can bring Indonesia much better . the important thing is we must prioritize ethical principles .')	neutral (1)	positive (0)	False
kecewa dengan specs desain galaxy s5 . **nggak** sesuai dengan harapan . terlalu biasa saja . ('disappointed with the specs design of the galaxy s5 . not according to expectations . too ordinary .')	negative (2)	negative (2)	True

Table 3.12: Examples of modified sentences and their corresponding gold labels, predicted labels, and label match status from the negation word substitution tests on IndoBERT.

INV Test: Orthographical Errors

This test analyzes the model's robustness to orthographical errors by evaluating whether it maintains the original sentiment when sentences contain errors, such as character swaps with neighboring characters or random character deletions within words.

Previous studies showed that even small perturbations can significantly lower the performance of the language model. For example, Moradi and Samwald (2021) showed that Transformer-based models are susceptible to character-level perturbations and often fail even when the input changes are minor and harmless. Náplava et al. (2021) recommended choosing noise distributions from real-world error corpora instead of arbitrarily choosing rates. Palin et al. (2019) showed that the average typing speed of people on mobile devices is 36.2 WPM with 2.3% uncorrected errors.

Based on previous research, we developed a perturbation function that alters words inside a sentence by randomly removing one letter or swapping two neighboring characters. We only took words longer than three characters into consideration in order to preserve the sentence's general readability, and this process was applied with a probability of 0.3 per word. This rate is selected to replicate noisy conditions of informal online text while maintaining the sentence's readability. We utilized the *Random* Python package to generate sentences containing orthographical errors.

The purpose of the perturbation is to simulate the types of errors that are typically found in informal or casual writing. After applying this perturbation to every sentence in the dataset, we saved the noisy version of the SmSA test subset as a new CSV file for additional testing. Table 3.13 presents examples of sentences that have been modified with orthographical errors.

Listing 3.7 shows the Python script that creates orthographical errors in the sentences from the SmSA test subset.

```python
def add_indonesian_typos(sentence, typo_prob=0.3):
    words = sentence.split()
    perturbed_words = []

    for word in words:
        if len(word) > 3 and random.random() < typo_prob:
            typo_type = random.choice(["swap", "delete"])
            i = random.randint(1, len(word)-2)

            if typo_type == "swap":
                chars = list(word)
                chars[i], chars[i+1] = chars[i+1], chars[i]
                word = ''.join(chars)
            elif typo_type == "delete":
                word = word[:i] + word[i+1:]

        perturbed_words.append(word)

    return ' '.join(perturbed_words)
```

Listing 3.7: The `add_indonesian_typos` function

Gold Label	Original Sentence	Modified Sentence
positive (0)	foto aku cuma modal kamera xiaomi : tetapi **tetap suka** ! (‘my photos are only using a xiaomi camera : but i still like them !’)	foto aku cuma modal kamera xiaomi : tetapi **ttap suak** !
neutral (1)	**kata** nya lagi banyak promo di **shopee** . (‘he said there are lots of promotions on shopee .’)	**kaat** nya lagi banyak promo di **shoepe** .
negative (2)	saya kecewa karena saya sudah **transfer** dan belum dikonfirmasi saya telepon malah pelanggan sibuk terus , **aduh** kecewa nih gue . **udah** kecewa nih gue . (‘i’m disappointed because i’ve transferred and it hasn’t been confirmed yet . i called but the customer was busy all the time. oh , i’m disappointed .’)	saya kecewa karena saya sudah **trnasfer** dan belum dikonfirmasi saya telepon malah pelanggan sibuk terus , **audh** kecewa nih gue .

Table 3.13: Examples of original and orthographically perturbed Indonesian sentences used in the ***INV*** test for orthographical errors.

We loaded the dataset containing sentences perturbed with orthographical errors from a CSV file. Then, we executed the *prediction process*. We compared each predicted label to its corresponding gold label to check if the model correctly maintained the original sentiment despite typos. We recorded whether the prediction matched the gold label, then calculated the total number of samples, the number of incorrect predictions, and the failure rate, which represents the percentage of incorrect or unexpected outputs. Listing 3.8 shows the Python script used to compare the gold labels and prediction labels for the orthographical errors test.

```
df["predicted_label"] = predict(df["sentence"].tolist())
df["label_match"] = df["gold_label"] == df["predicted_label"]

total = len(df)
correct_behavior = df["label_match"].sum()
failure_rate = ((total - correct_behavior) / total) * 100

print(f"Behavioral Test Results ({model_name}):")
print(f"Total samples: {total}")
print(f"Failures (unexpected behavior): {total - correct_behavior}")
print(f"Failure rate: {failure_rate:.2f}%")

if save_path:
    df[["sentence", "gold_label", "predicted_label", "label_match"]].
        to_csv(save_path, index=False)
```

Listing 3.8: The Python script used to compare the labels for orthographical errors test.

We saved the detailed prediction results, such as the perturbed sentences, gold labels, predicted labels, and whether the predictions matched to the CSV file. We ran this evaluation twice: once for the IndoBERT model and once for the mBERT model, each with their respective tokenizers and fine-tuned weights, generating separate CSV files for each model's typo robustness test. 3.14 shows example modified sentences from the evaluation of the orthographical error using the mBERT model, including their corresponding gold labels, predicted labels, and label match status.

Modified Sentence	Gold Label	Predicted Label	Label Match
kolam **rnang** di hotel aston bikin tidak mau berenti **berenng** . nyaman **banet** . air nya bersih . tempat bilas nya **nyamn** . ('the swimming pool at the aston hotel makes you not want to stop swimming . very comfortable . the water is clean . the place to rinse off is comfortable .')	positive (0)	positive (0)	True
kalau **laapr** ya baiknya **makna** . ('if you're hungry , it's best to eat .')	neutral (1)	positive (0)	False
kalua dibandingkan sama **jepnag** , indonesia itu **tdiak** ada **aap**-apa nya . ('when compared to japan , indonesia is nothing .')	negative (2)	neutral (1)	False

Table 3.14: Examples of modified sentences and their corresponding gold labels, predicted labels, and label match status from the negation word substitution test on mBERT.

INV Test: Formality Levels

We ran a behavioral test to evaluate the model's robustness across different politeness levels using the manually constructed formality-level datasets mentioned in Subsection 5.2.2. Then, we executed the *prediction process.*

We compared the prediction labels with the original gold labels to determine whether the model's output aligned with the original sentiment. For each sentence, we recorded whether the prediction matched the gold label and subsequently computed the total number of test samples, the number of incorrect predictions, and the failure rate, defined as the percentage of predictions that diverged from the expected behavior. Listing 3.9 shows the Python script used to compare the gold and predicted labels for the formality-level tests.

```
df["predicted_label"] = predict(df["sentence"].tolist())
df["label_match"] = df["gold_label"] == df["predicted_label"]

total = len(df)
correct_behavior = df["label_match"].sum()
failure_rate = ((total - correct_behavior) / total) * 100

print(f"Behavioral Test Results ({test_name}):")
print(f"Total samples: {total}")
print(f"Failures (unexpected behavior): {total - correct_behavior}")
print(f"Failure rate: {failure_rate:.2f}%")

if save_path:
    df[["sentence", "gold_label", "predicted_label", "label_match"]].
        to_csv(save_path, index=False)
```

Listing 3.9: The Python script used to compare the labels for formality level tests.

We executed the test three times for each model, once for each level of politeness. Finally, we saved the gold labels, predicted labels, and whether the predictions matched in separate CSV files. 3.15 shows example modified sentences from the output file of the medium-formality level evaluation using the mBERT model, including their corresponding gold labels, predicted labels, and label match status.

Sentence	Gold Label	Predicted Label	Label Match
gibran bilang kalau qris udah memberikan solusi pembayaran praktis tanpa harus bergantung sama uang tunai . bagus banget ya ! ('gibran said that qris has provided a practical payment solution without having to rely on cash . that's really great !')	positive (0)	neutral (1)	False
rodanya mobil ada empat . ('the car has four wheels .')	neutral (1)	neutral (1)	True
aku nggak suka makan donat ('i don't like eating donuts')	negative (2)	negative (2)	True

Table 3.15: Examples of sentences and their corresponding gold labels, predicted labels, and label match status from the negation word formality-level tests on mBERT.

4 Results

4.1 Conventional Metrics

As shown in Figure 4.1, IndoBERT consistently outperforms mBERT across all metrics by approximately 10 percentage points, indicating its stronger suitability for Indonesian sentiment analysis, likely due to language-specific pre-training and adaptation.

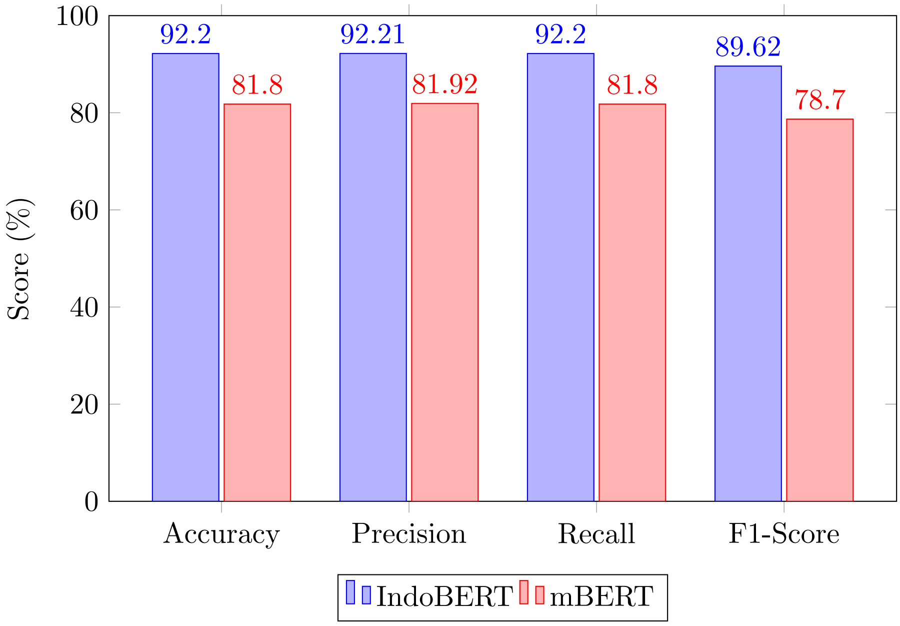

Figure 4.1: Bar chart comparing IndoBERT and mBERT across conventional evaluation metrics on the SmSA test subset.

The F1-score results from our testing differ from those presented in the IndoNLU benchmark (Wilie et al., 2020), despite strictly adhering to their fine-tuning tutorial and example code posted on their GitHub[1] without making any modifications, additions, or omissions. Possible causes of these different scores might be hardware and implementation factors, such as variations in GPU/TPU precision and the training libraries used. The learning rate might also be a contributing factor. In the IndoNLU benchmark paper, Wilie et al. (2020) used 1×10^{-5} learning rate to fine-tune both IndoBERT and mBERT. Meanwhile, the tutorial and example in their GitHub[1] used 3×10^{-6} learning rate, which we followed in the fine-tuning process of IndoBERT annd mBERT in this thesis.

Table 4.1 shows the difference of F1-scores between this thesis and the IndoNLU benchmark.

Model	F1-score in this Thesis	F1-score in the IndoNLU Benchmark
IndoBERT	89.62%	87.73%
mBERT	78.70%	84.14.%

Table 4.1: Comparison of F1-scores from this thesis and the IndoNLU benchmark (Wilie et al., 2020) for IndoBERT and mBERT on the SmSA dataset.

[1] https://github.com/IndoNLP/indonlu/blob/master/examples/finetune_smsa.ipynb

4.2 Behavioral Testing

As shown in Table 4.2, IndoBERT generally exhibits lower failure rates than mBERT across seven of eight behavioral tests. Notably, IndoBERT performed well in the ***INV*** tests (Tests 3–5), maintaining failure rates below 15%, with especially low rates of 7.98% in Tests 3 and 4. In contrast, mBERT struggled significantly with these same tests, reaching failure rates above 30%. IndoBERT also shows better performance on the orthographical errors test (Test 5) with a failure rate of 14.40%, while mBERT has a failure rate of 22.60%.

For the ***DIR*** tests (Tests 1–2), IndoBERT outperformed mBERT in Test 1, where mBERT failed 52.40% of the time compared to IndoBERT's 19.40%. However, in Test 2, mBERT achieved a lower failure rate (7.20%) than IndoBERT (37.20%), indicating better handling of added positive contexts.

Both models show varying degrees of robustness to formality variation (Tests 6-8). IndoBERT's failure rate increases gradually as politeness level decreases: from 13.33% (high politeness) to 20.00% (medium) and 23.33% (low). In contrast, mBERT exhibits more fluctuation, starting at 36.37% for high politeness, improving slightly to 26.27% for medium, but then increasing again to 40.00% for low-formality sentences. These results suggest that IndoBERT handles differences in politeness more consistently, while mBERT struggles particularly with informal input.

Test No.	Test Type and Description	Total Samples	IndoBERT Failures	IndoBERT Failure Rate (%)	mBERT Failures	mBERT Failure Rate (%)
1	***DIR***: Insert negative sentence *Saya benci matematika.* ('I hate mathematics.')	500	97	19.40	**262**	**52.40**
2	***DIR***: Insert positive sentence *Saya cinta matematika.* ('I love mathematics.')	500	**186**	**37.20**	36	7.20
3	***INV***: Substitute negation word *tidak* to *nggak*	188	15	7.98	**60**	**31.91**
4	***INV***: Substitute negation word *tidak* to *gak*	188	15	7.98	**58**	**30.85**
5	***INV***: Swap characters with its neighbor or remove characters	500	72	14.40	**113**	**22.60**
6	***INV***: Sentences with high politeness level	30	4	13.33	**11**	**36.37**
7	***INV***: Sentences with medium politeness level	30	6	20.00	**8**	**26.27**
8	***INV***: Sentences with low politeness level	30	7	23.33	**12**	**40.00**

Table 4.2: Behavioral testing test results for IndoBERT and mBERT on the SmSA test subset.

Overall, the results demonstrate that IndoBERT is generally more robust than mBERT when faced with linguistic perturbations in Indonesian. While both models perform reasonably well on unperturbed test data, behavioral tests reveal their varying abilities to handle linguistic changes. These results will be further explored in the Discussion chapter.

5 Discussion

In this chapter, we discuss the results of the four behavioral tests conducted in this study to deeply analyze the robustness of IndoBERT and mBERT in analyzing sentiments in Indonesian. To facilitate easier analysis of model failures, we extracted all misclassified sentences into separate CSV files, resulting in 16 files. Listing 5.1 presents the implementation of the `false_extraction` function used to generate model predictions.

```python
def false_extraction(data_path, save_path):

    df = pd.read_csv(data_path)

    df_filtered = df[df["label_match"].astype(str).str.contains(r"\bFALSE\b", case=False, regex=True)]

    if save_path:
        df_filtered.to_csv(save_path, index=False)
```

Listing 5.1: The `false_extraction` function

5.1 Sentence Insertion

In this test, IndoBERT and mBERT demonstrated contrasting results. While IndoBERT performed better on negative additional sentences, mBERT achieved a lower failure rate on positive additional sentences. In general, both models demonstrated low robustness against additional sentences and difficulties in differentiating between the context sentence and the distracting statement.

5.1.1 IndoBERT

IndoBERT displays a high failure rate of 37.20% on the positive directional expectation test, misclassifying 186 sentences out of 500. Although IndoBERT correctly predicted the sentiment for most neutral or positive original sentences, the model struggled particularly with sentences where a negative sentence was appended with a positive sentence, indicating sensitivity to contextual dominance. Also, IndoBERT overrode the appended positive sentiment when the original text is strongly negative, such as complaints about services or products.

This result shows that IndoBERT is only moderately robust in inserting sentences with positive sentiments. It is still over-reliant on the original sentiment and ignores the appended phrases. Additionally, it still performed poorly on service-related negativity. Table 5.1 shows an example from the positive sentence insertion test on IndoBERT.

Original Sentence	tidak bisa menggunakan kode promo di aplikasi android , telpon ke pelayanan pelanggan bilang nya tidak ada solusi nya . oke uninstall dan tidak rekomendasi ! ('can't use promo code in android app , call customer service said there is no solution . okay uninstall and not recommended !')
Appended Sentence	tidak bisa menggunakan kode promo di aplikasi android , telpon ke pelayanan pelanggan bilang nya tidak ada solusi nya . oke uninstall dan tidak rekomendasi ! **saya cinta matematika .** ('can't use promo code in android app , call customer service said there is no solution . okay uninstall and not recommended ! **i love mathematics .**')
Gold Label of the Original Sentence	negative (2)
Expected Label of the Appended Sentence	positive (0) neutral (1)
Predicted Label of the Appended Sentence	negative (2)
Label Match	False

Table 5.1: Example of an incorrectly-labeled sentence from the positive sentence insertion test on IndoBERT.

On the negative directional expectation test, IndoBERT shows a failure rate of 19.40%, misclassifying 97 sentences in total. This indicates that IndoBERT prioritized the explicit negative sentiment of the appended phrase, demonstrating robustness in handling negative sentiments. However, the model's bias towards negative statements overshadowed neutral and positive signals, which indicates over-reliance on lexical cues. The model handled negative expressions effectively, but failed to handle mixed sentiments. Table 5.2 shows an example from the negative sentence insertion test on IndoBERT.

Original Sentence	terima kasih untuk buah karya mu di jakarta yang jadi sejuk santun bersama pak sandi juga pak anies ('thank you for your work in jakarta which has become a cool and polite city with mr. sandi and mr. anies')
Appended Sentence	terima kasih untuk buah karya mu di jakarta yang jadi sejuk santun bersama pak sandi juga pak anies **saya benci matematika .** ('thank you for your work in jakarta which has become a cool and polite city with mr. sandi and mr. anies **i hate mathematics .**')
Gold Label of the Original Sentence	positive (0)
Expected Label of the Appended Sentence	neutral (1) negative (2)
Predicted Label of the Appended Sentence	positive (0)
Label Match	False

Table 5.2: Example of an incorrectly-labeled sentence from the negative sentence insertion test on IndoBERT.

5.1.2 mBERT

mBERT exhibits a lower failure rate of 7.20% on the positive directional expectation test, misclassifying 36 sentences out of 500. The model showed higher robustness than IndoBERT, despite prioritizing the dominant negative sentiment over the appended positive phrase. An example from the positive sentence insertion test on mBERT is shown in Table 5.3.

Original Sentence	saya sudah transfer ratusan ribu dan sesuai nominal transfer . tapi tiket belum muncul juga . harus diwaspadai ini aplikasi ini . bahaya . ('i have transferred hundreds of thousands and according to the nominal transfer . but the ticket has not appeared yet . must be aware of this application . dangerous .')
Appended Sentence	saya sudah transfer ratusan ribu dan sesuai nominal transfer . tapi tiket belum muncul juga . harus diwaspadai ini aplikasi ini . bahaya . **saya cinta matematika .** ('i have transferred hundreds of thousands and according to the nominal transfer . but the ticket has not appeared yet . must be aware of this application . dangerous . **i love mathematics .**')
Gold Label of the Original Sentence	negative (2)
Expected Label of the Appended Sentence	positive (0) neutral (1)
Predicted Label of the Appended Sentence	negative (2)
Label Match	False

Table 5.3: Example of an incorrectly-labeled sentence from the positive sentence insertion test on mBERT.

On the negative directional expectation test, mBERT shows a higher failure rate of 52.40%, misclassifying 262 sentences out of 500. It demonstrated strong performance on explicit negative sentiments but still struggled with implicit negativity and sarcasm, such as the sentence. Furthermore, like IndoBERT, mBERT over-prioritized the appended negative phrases over contextual understanding. It also faced challenges in classifying colloquial Indonesian sentences. An example from the negative sentence insertion test on mBERT is shown in Table 5.4.

Original Sentence	aku kecewa . tadi pergi kfc dapat wedges macam tidak niat dibuat nya . sudah begitu memberi mayones nya berantakan banget . ('i'm disappointed . i went to kfc earlier and got wedges that didn't look like they were made intentionally . then when i put mayonnaise on them , it was really messy .')
Appended Sentence	aku kecewa . tadi pergi kfc dapat wedges macam tidak niat dibuat nya . sudah begitu memberi mayones nya berantakan banget . **saya benci matematika .** ('i'm disappointed . i went to kfc earlier and got wedges that didn't look like they were made intentionally . then when i put mayonnaise on them , it was really messy . **i hate mathematics .**')
Gold Label of the Original Sentence	negative (2)
Expected Label of the Appended Sentence	negative (2)
Predicted Label of the Appended Sentence	positive (0)
Label Match	False

Table 5.4: Example of an incorrectly-labeled sentence from the negative sentence insertion test on mBERT.

5.2 Negation Handling

In these tests, IndoBERT and mBERT have quite diverse results in their performance. IndoBERT manages to maintain high accuracy when we replaced *tidak* with *nggak* or *gak*. Meanwhile, mBERT has a lower accuracy than IndoBERT, exhibiting failure rates three times as high as IndoBERT's. Here, accuracy refers to the proportion of predicted labels that match the gold labels from the dataset.

There exists a typical failure pattern in both IndoBERT and mBERT, where they sometimes made false positive predictions, which are the result of having multiple negations in the sentence, for example ***nggak*** *bisa menggunakan kode promo di aplikasi android , telpon ke pelayanan pelanggan bilang nya* ***nggak*** *ada solusi nya . oke uninstall dan* ***nggak*** *rekomendasi !* ('can't use promo code in android app , call customer service and they said there is no solution . okay uninstall and don't recommend !').

In general, the negation handling tests reveal that IndoBERT is more robust than mBERT in dealing with formal and colloquial Indonesian negation words. We assumed this is because IndoBERT was pre-trained on datasets that include formal and colloquial Indonesian, unlike mBERT, which was only pre-trained on Wikipedia.

5.2.1 IndoBERT

IndoBERT has equal failure rates of 7.98%. In particular, the model misclassified 15 identical sentences in both *nggak* and *gak* datasets. The model retained the correct sentiment label for negated sentiments and sentiment reversal. For example, for the word ***enak*** ('tasty'), which has a positive sentiment, the model labeled ***nggak enak*** ('not tasty') as negative. An example of a correctly labeled sentence from the *nggak* negation substitution test on IndoBERT is shown in Table 5.5.

Original Sentence	sudah harga mahal , nggak enak pula makanan nya , **tidak** mau lagi deh ke restoran itu . ('the price is expensive and the food is not tasty , i don't want to go to that restaurant again .')
tidak **Substitution**	sudah harga mahal , nggak enak pula makanan nya , **nggak** mau lagi deh ke restoran itu .
Gold Label	negative (2)
Predicted Label	negative (2)
Label Match	True

Table 5.5: Example of a correctly-labeled sentence from the negation substitution test on IndoBERT.

Despite that, IndoBERT still made several misclassifications, particularly where negation co-occurs with ambiguous or weak sentiment cues. This indicates that the model might rely too much on lexical negation without deeper syntactic parsing. Table 5.6 shows an example of an incorrectly labeled sentence from the *nggak* negation substitution test on IndoBERT.

Original Sentence	**tidak** perlu diragukan lagi , semua orang memang membenci dia , aku pun sama . ('there is no doubt , everyone hates him , including me .')
tidak **Substitution**	**nggak** perlu diragukan lagi , semua orang memang membenci dia , aku pun sama .
Gold Label	negative (2)
Predicted Label	positive (0)
Label Match	False

Table 5.6: Example of an incorrectly-labeled sentence from the negation substitution test on IndoBERT.

5.2.2 mBERT

mBERT shows different failure rates of *nggak* and *gak*. mBERT misclassfied 60 sentences in *nggak* dataset (31.91%) and 58 sentences in *gak* dataset (30.85%). Sentences with strong negative keywords like ***kecewa*** ('disappointed') and ***buruk*** ('bad') retained the correct sentiment label despite the substitution, demonstrating that mBERT's dependency on terms with striking sentiment weight to override the negation noise.

Errors occur when negation alters sentiment-bearing phrases such as ***tidak*** *mau lagi* ('don't want anymore') to ***gak*** *mau lagi*. Most misclassified sentences have negative sentiment as the gold label but are predicted as positive. This occurs when negation co-occurs in the sentence. An example of a correctly-labeled sentence from the negation substitution test on mBERT is shown in Table 5.7, and an example of an incorrectly-labeled one is shown in Table 5.8.

Original Sentence	saya mesan tiket tapi **tidak** dikonfirmasi dan proses pengembalian uang yang gak jelas dan belum ada sampai sekarang , pelayanan nya yang sangat buruk dan saran saya jangan pesan di sini , sangat mengecewakan . ('i ordered a ticket but it was not confirmed and the refund process was unclear and has not happened until now , the service is very bad and my advice is do not order here , it is very disappointing .')
***tidak* Substitution**	saya mesan tiket tapi **gak** dikonfirmasi dan proses pengembalian uang yang gak jelas dan belum ada sampai sekarang , pelayanan nya yang sangat buruk dan saran saya jangan pesan di sini , sangat mengecewakan .
Gold Label	negative (2)
Predicted Label	negative (2)
Label Match	True

Table 5.7: Example of a correctly-labeled sentence from the negation substitution test on mBERT.

Original Sentence	**tidak** ada enak nya makan di sana , bakso **tidak** enak , teh manis nya rasanya aneh , mi ayam nya mie nya kelembekan , **tidak** mau lagi deh ke sana . ('there's nothing good about eating there , the meatballs aren't delicious , the sweet tea tastes strange , the chicken noodles are soggy , i don't want to go there anymore .')
***tidak* Substitution**	**gak** ada enak nya makan di sana , bakso **gak** enak , teh manis nya rasanya aneh , mi ayam nya mie nya kelembekan , **gak** mau lagi deh ke sana .
Gold Label	negative (2)
Predicted Label	positive (0)
Label Match	False

Table 5.8: Example of an incorrectly-labeled sentence from the negation substitution test on mBERT.

5.3 Orthographical Errors

IndoBERT and mBERT show different results in this test. IndoBERT is robust to orthographical errors with a relatively low failure rate, but mBERT only exhibits moderate robustness due to its struggle to handle colloquial Indonesian text. IndoBERT and mBERT struggled to classify sentences where sentiment-bearing words are perturbed. For example, when the word ***kecewa*** ('disappointed'), which has a negative sentiment, was replaced with ***keceaw***.

Overall, the orthographical error tests reveal that IndoBERT is more robust than mBERT in handling orthographical errors. We assumed this is because IndoBERT was pre-trained on datasets that include formal and colloquial Indonesian, unlike mBERT, which was only pre-trained on Wikipedia.

5.3.1 IndoBERT

IndoBERT shows a low failure rate of 14.40%, misclassifying 72 sentences out of 500 from the SmSA test subset. We noted that errors arise when typos distort key sentiment-bearing words.

For example, when the word ***kecewa*** ('disappointed'), which has a negative sentiment, was replaced with ***keceaw***. However, sentences with multiple typos were still correctly classified as long as the typos do not affect clear sentiment-bearing keywords, assuming that the models rely on noticeable sentiment terms to override noise.

Table 5.9 shows an example of the correctly labeled sentences, where the errors do not occur on the sentiment-bearing word, and Table 5.10 shows an example of the mislabeled sentences, where the sentiment-bearing word is perturbed.

Original Sentence	pelayanan di **hotel salak** bogor tidak sebagus yang gue membayangkan . **fasilitas** nya juga **biasa** banget padahal **kata** nya hotel **bintang** lima . hm . kecewa . **kayak** nya sih nanti-nanti tidak mau ke sana **lagi** . ('the service at salak hotel bogor is not as good as i imagined . the facilities are also very ordinary even though they say it's a five star hotel . hm . disappointed . i don't think i want to go there again .')
Perturbed Sentence	pelayanan di **htel salk** bogor tidak sebagus yang gue membayangkan . **fasiiltas** nya juga **bisa** banget padahal **kta** nya hotel **bintng** lima . hm . kecewa . **kayk** nya sih nanti-nanti tidak mau ke sna **lgi** .
Gold Label	negative (2)
Predicted Label	negative (2)
Label Match	True

Table 5.9: Example of a correctly-labeled sentence from the orthographical error test on IndoBERT.

Original Sentence	**luar** biasa **kecewa** sangat dengan **aplikasi** ini , saya sudah hubungi pusat panggilan untuk cek pemesanan saya karena saya sudah **bayar** dan **tidak muncul** e - tiket nya , bahkan saya **sudah** kirim bukti transaksi nya ke e-mail , hingga saat ini **tidak** ada pihak pusat **panggilan yang** menghubungi , **kecewa** . ('extremely disappointed with this application , i have contacted the call center to check my order because i have paid and the e-ticket does not appear , i have even sent proof of the transaction to email , until now no party from the call center has contacted me , disappointed .')
Perturbed Sentence	**laur** biasa **keceaw** sangat dengan **aplkiasi** ini , saya sudah hubungi pusat panggilan untuk cek pemesanan saya karena saya sudah **baar** dan **tidk mucnul** e - tiket nya , bahkan saya **sudh** kirim bukti transaksi nya ke e-mail , hingga saat ini **tdiak** ada **piak** pusat **panggiln yag menghbungi** , **keceaw** .
Gold Label	negative (2)
Predicted Label	positive (0)
Label Match	False

Table 5.10: Example of an incorrectly-labeled sentence from the orthographical error test on IndoBERT.

This testing shows that IndoBERT handled most orthographical errors effectively, showing a robust pre-training process on noisy Indonesian text. However, it still struggled with multiple typos in a single sentence and misspellings that affect key sentiment-bearing words.

5.3.2 mBERT

mBERT exhibits a moderate failure rate of 22.60%, misclassifying 113 sentences out of 500 from the SmSA test subset. It struggled with sentences that contain many orthographical errors. For example, the sentence ***kmarin*** *gue* ***dtang*** *ke tempat makan* ***brau ynag*** *ada di* ***dgo aats*** *. gue kia makanan nya enak karena harga nya mahal .* ***ternyta*** *,* ***bor****-boro . tidak mau* ***lai*** *deh ke tempat itu . sudah mana tempat nya* ***juag*** *tidak* ***naman*** *bnget , terlalu sempit .* ('yesterday i came to a new place to eat in upper dago . i think the food is delicious because the price is expensive . turns out, it's a waste . i don't want to go to that place anymore . the place is also not very comfortable , it's too narrow .'). Like IndoBERT, we also noted that errors arise when typos distort key sentiment-bearing words. For example, when the word ***kecewa*** ('disappointed'), which has a negative sentiment, was replaced with ***keceaw***.

The results indicate that mBERT demonstrates greater robustness in certain domains than others. For example, it has a high accuracy for customer service complaints, such as the sentence *saya* ***mesn*** *tiket* ***tpai tiadk dikonformsi*** *dan proses* ***pengemblian unag*** *yang tidak jelas dan belum ada sampai skarang , pelayanan nya yang* ***sangt burk*** *dan* ***sran saay janagn*** *pesan di sini ,* ***sagat*** *mengecewakan .* ('i ordered a ticket but it was not confirmed and the refund process was unclear and has not happened until now , the service is very bad and my advice is do not order here , it is very disappointing.'). However, it is more prone to misclassification on political text, such as the sentence ***betl*** *. era sebelum perbedaan politik suatu hal ynag biasa . media sosial damai . sejak* ***jkoowi nypres*** *, pendukung-penudkung nya suka* ***cair*** *ribut ke yang berbeda . contoh , di* ***pligub*** *dki* ***kmai*** *ikut ulama , dibilang* ***radkial*** *intoleran .* ***jangn*** *bawa-bawa* ***agma*** *dan ulama .* ***sekaang*** *ya dengan jadi cawapres , ynag* ***tidk pliih*** *dibilang* ***anit ulaa*** *. hadeuh .* ***psing*** *.* ('right . the era before political differences were commonplace . social media was peaceful . since jokowi ran for president , his supporters like to pick fights with those who are different . for example , in the dki gubernatorial election we joined the ulama , called radical intolerant . don't bring religion and ulama into it . now with him as the vice presidential candidate , those who don't vote are called anti-ulama . oh my . what a headache .').

Additionally, mBERT struggled with Indonesian-specific abbreviations and slang, such as *gue* ('I'), likely due to its training being limited to formal sources like Wikipedia and lacking exposure to colloquial Indonesian.

Table 5.11 shows an example of the correctly labeled sentences, and Table 5.10 shows an example of the mislabeled sentences.

Original Sentence	apa **bagus** nya ya itu acara l - men , jijik gue sih **liat** nya . ('what's so good about it? it's an l - men show , i'm disgusted when i see it .')
Perturbed Sentence	apa **bags** nya ya itu acara l - men , jijik gue sih **lita** nya .
Gold Label	negative (2)
Predicted Label	negative (2)
Label Match	True

Table 5.11: Example of a correctly-labeled sentence from the orthographical error test on mBERT.

Original Sentence	harus belanja di shopee karena shopee itu memberikan bukti **pelayanan** yang **baik** , **bukan** sekadar **janji** doang , seperti mantan itu ('must shop at shopee because shopee provides proof of good service , not just promises , like that ex')
Perturbed Sentence	harus belanja di shopee karena shopee itu memberikan bukti **pelaanan** yang **bak** , **bukna** sekadar **jnji** doang , seperti mantan itu
Gold Label	positive (0)
Predicted Label	negative (2)
Label Match	False

Table 5.12: Example of an incorrectly-labeled sentence from the orthographical error test on IndoBERT.

This testing shows that mBERT is moderately robust to typos but still fails when errors disrupt key sentiment indicators. It still struggled with colloquial Indonesian, extreme word distortions, sentences laden with orthographical errors, and it lacks sensitivity to specific domains.

5.4 Formality Levels

The formality levels tests reveal that IndoBERT outperforms mBERT in all formality levels, especially low formality. This is likely because IndoBERT was pre-trained on a combination of formal and colloquial Indonesian datasets, whereas mBERT was trained solely on Wikipedia data.

5.4.1 IndoBERT

IndoBERT presents three different failure rates for each of the datasets: four failures for high politeness (13.33%), six failures for medium politeness (20.00%), and seven failures for low politeness (23.33%). Table 5.13 shows the general linguistic challenges and limitations that IndoBERT faced in handling linguistic variations through these tests.

Lexical Variations	**Example**
Slang	kureng ('too little') (informal form of *kurang*)
Abbreviations	yg ('which') (from *yang*)
Non-standard spellings	sbnrny (from *sebenarnya*)

Contextual Nuances	**Example**
Implied negativity	halah, katanya pemerintah mau prioritasin pendidikan (sarcasm, indicated from the word *halah*) ('oh, he said the government wants to prioritize education')

Negation Handling	**Example**
Inconsistencies with informal negation	budi bilang bu ika **gabisa** jelasin materi (classified correctly) ('budi said ma'am ika couldn't explain the material') aku gasuka donat (classified incorrectly) ('i don't like donuts')

Table 5.13: Linguistic challenges affecting IndoBERT performance.

In the high-formality dataset, the model struggled in reading sentiments implicitly stated in the sentence, such as *saya mendukung tindakan preventif dalam menjaga kelestarian lingkungan yang merupakan tanggung jawab bersama yang harus dilaksanakan dengan penuh kesadaran dan komitmen .* ('i support preventive measures in preserving the environment , which is a shared responsibility that must be carried out with full awareness and commitment .'). Although this statement is a factual neutral statement, the speaker conveys their positive support for the statement.

IndoBERT struggled to classify neutral sentences as negative in the medium-formality dataset. For example, *hpku ilang di stasiun .* ('my cellphone was lost at the station .'). IndoBERT classified this sentence as negative despite being factual and non-sentiment-laden.

In the low-formality dataset, IndoBERT misclassified sentences with slang or abbreviations, such as the sentence *njirr , mall di mana-mana tapi taman kota minim ! kureng bgt sih ini pas ngeremcanain bangun kotanya* ('njirr[1] , malls everywhere but minimal city parks ! this really sucks when planning to build the city'). Although this sentence is a complaint with negative sentiment, IndoBERT predicted it as neutral.

[1] An Indonesian curse word meaning 'dog' (Viklous, 2022)

5.4.2 mBERT

mBERT exhibits three different failure rates for each of the datasets: eleven failures for high politeness (36.37%), eight failures for medium politeness (26.27%), and twelve failures for low politeness (40.00%). The general linguistic challenges and limitations that mBERT faced in handling linguistic variations through these tests are shown in Table 5.14.

Lexical Variations	**Example**
Slang	gila ('crazy')
Abbreviations	bgs ('good') (from *bagus*)
Non-standard spellings	blkgn ('lately') (from *belakangan*)

Contextual Nuances	**Example**
Implied negativity	saya setuju bahwa kurangnya akses masyarakat terhadap ruang terbuka hijau adalah salah satu indikator kegagalan perencanaan tata ruang kota . ('i agree that the lack of public access to green open spaces is an indicator of the failure of urban spatial planning .')

Negation Handling	**Example**
Inconsistencies with negation	saya tidak suka makan donat (classified correctly) ('i don't like donuts') aku gasuka donat (classified incorrectly)

Table 5.14: Linguistic challenges affecting mBERT performance.

The model demonstrated high accuracy in the high-formality dataset but struggled with implicit sentiment and neutral-positive/negative boundaries. For example, in the sentence *saya setuju dengan pendapat gibran mengatakan bahwa qris telah memberikan solusi pembayaran praktis tanpa harus bergantung pada uang tunai .* ('i agree with gibran 's opinion that qris has provided a practical payment solution without having to rely on cash .'), the model incorrectly classified it as neutral. However, the speaker stated agreement with the statement, displaying positivity.

The model exhibited slightly lower accuracy in the medium-formality dataset, particularly with false negatives where neutral or positive statements are misclassified as negative. For instance, the sentence *anak itu jago banget main piano sampai menang lomba .* ('the child is so good at playing the piano that he won the competition .') was incorrectly labeled as negative despite its positive tone. Additionally, the model struggles with negation, as seen in *aku nggak suka makan donat .* ('i don't like eating donuts .'), which was misclassified as neutral instead of negative.

The model's performance in the low-formality dataset decreased significantly due to slang, abbreviations, and non-standard grammar. The sentence *lo cantik banget dah gila bisa ya secakep itu* ('you're so beautiful it's crazy how you can be that beautiful') was misclassified as negative instead of positive, likely due to the informal intensifiers *dah gila* ('crazy'), highlighting the model's inability to handle colloquial and explicit expressions.

5.5 Overall Discussion

This thesis evaluates the robustness of IndoBERT and mBERT in handling linguistic variations in sentiment analysis for Indonesian text and identifies the limitations of both models through behavioral testing. This section discusses the overall results of both models based on the testing and evaluation process.

We discovered that IndoBERT outperformed mBERT in several key areas. It showed a stronger capability in processing colloquial Indonesian, demonstrated greater robustness to negation substitutions, and handled orthographical errors more effectively. These advantages are likely due to its pre-training on various Indonesian datasets, including formal and informal Indonesian. However, IndoBERT still struggled with contextual understanding, especially in implicit and mixed sentiments cases.

On the other hand, mBERT performed fairly well when dealing with highly formal texts and shows stronger retention of explicit sentiment markers. Despite this, its performance dropped significantly when processing texts in informal Indonesian, and it is susceptible to negation variations in informal contexts.

The two models' most common shared limitation is their over-reliance on lexical cues rather than a deeper understanding of context. This is evident in their poor handling of implicit sentiments, such as sarcasm or irony, inconsistent predictions on sentences expressing mixed sentiments, vulnerability to appended phrases that disrupt the contextual flow, and limited robustness to extreme orthographical variations.

These findings carry important implications for real-world applications. IndoBERT is better suited for analyzing user-generated content and informal communication, while mBERT still remains effective for processing formal documents and standardized texts. Nevertheless, neither model currently demonstrates sufficient top-notch robustness for fully automated sentiment analysis across the wide range of linguistic variability found in Indonesian.

Future research should focus on designing more context-aware model architectures and expanding training corpora better to represent the full spectrum of Indonesian language use. This study offers both a methodological foundation and an empirical benchmark for further developing Indonesian language models.

6 Conclusion

6.1 Summary of Findings

In conventional metrics evaluation, IndoBERT outperformed mBERT by approximately 10% across accuracy, precision, recall, and F1-score. For behavioral testing, IndoBERT outperformed mBERT in almost all test cases except for one. IndoBERT showed better results for negative sentence insertions but struggled with positive insertions in sentence insertion tests. On the other hand, mBERT showed better results with positive sentence insertions but failed significantly with negative insertions. IndoBERT displayed robustness with colloquial negation handling. On the other hand, mBERT still struggled with informal negations. Both models showed equal results for orthographical error perturbation; however, errors in sentiment-bearing words decreased the performance of both models. IndoBERT exhibited a more consistent result in formality levels despite still struggling with slang or implied sentiments. Meanwhile, mBERT performed poorly on the informal text and implicit sentiments.

IndoBERT still struggled with handling slang, abbreviation, and implied sentiment. It is also overly sensitive to negative sentences and words laden with strong sentiments. IndoBERT also relied heavily on lexical cues over syntactic parsing. Meanwhile, mBERT is weak in informal language and negations and overly sensitive to strong sentiment words. Furthermore, it is overly sensitive to strong sentiment words.

Generally, IndoBERT is more robust than mBERT due to its language-specific pre-training, but both models still have context-specific weaknesses. This study reveals gaps in both models' handling of informal language, negation, and contextual nuance. These findings show that IndoBERT is more robust and more effective than mBERT in handling linguistic variations in sentiment analysis for Indonesian, despite the limitations that were uncovered in this study.

6.2 Future Work

The limited research on low-resource languages highlights the need for further exploration. In the context of Indonesian and behavioral testing, two experimental directions are particularly interesting:

1. **Expanding Behavioral Testing Coverage for Indonesian**
   This thesis only evaluated IndoBERT and mBERT using very little of the many linguistic variations in Indonesian. More linguistic variations, such as sarcasm, mixed dialects, language-switching, longer and more complex sentences, nested negations, abbreviations, etc., exist in Indonesian. Creating a behavioral testing framework such as CheckList, specially created for Indonesians, will help advance the research in Indonesian NLP even further.

2. **Improving Model Robustness on Indonesian Datasets**
   The results of behavioral testing in this thesis show the limitations and weaknesses of IndoBERT and mBERT, which enables further improvement in collecting annotated Indonesian datasets and fine-tuning the models. Experiments with adversarial training are also recommended to incorporate syntactic parsing to handle negation and context shifts better.

7 Appendix

The materials accompanying this bachelor's thesis have been submitted separately and include the following:

- The PDF version of this thesis;
- The original and constructed datasets used in the experiments;
- The Python scripts used to conduct the experiments;
- The experimental results in CSV format; and
- The referenced literature in PDF format.

References

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

Wietse de Vries, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim. 2019. Bertje: A dutch BERT model. *CoRR*, abs/1912.09582.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Dwi Noverini Djenar. 2008. Which self? pronominal choice, modernity, and self-categorizations. *International Journal of the Sociology of Language*, 2008(189):31–54.

Lenggo Geni, Evi Yulianti, and Dana Indra Sensuse. 2023. Sentiment analysis of tweets before the 2024 elections in indonesia using bert language models. *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, 9(3):746–757.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Artia Irianti, Halimah, Sutedi, and Melda Agariana. 2025. Integration of bert and svm in sentiment analysis of twitter/x regarding constitutional court decision no. 60/puu-xxii/2024. *Jurnal Teknik Informatika (JUTIF)*, 6(2):469–482.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Karthikeyan K, Shaily Bhatt, Pankaj Singh, Somak Aditya, Sandipan Dandapat, Sunayana Sitaram, and Monojit Choudhury. 2022. Multilingual checklist: Generation and evaluation. In *Findings of the Association for Computational Linguistics: AACL-IJCNLP 2022*, pages 282–295, Online only. Association for Computational Linguistics.

Fajri Koto, Jey Han Lau, and Timothy Baldwin. 2021. Indobertweet: A pretrained language model for indonesian twitter with effective domain-specific vocabulary initialization. *CoRR*, abs/2109.04607.

Bernadette Kushartanti, Nazarudin Nazarudin, and R. Niken Pramanik. 2019. Varieties of indonesian negation in indonesian children's speech. In *Proceedings of the 5th International Conference on Linguistics, Literature and Culture*, pages 77–82. Pusat Pengajian Ilmu Kemanusiaan, Universiti Sains Malaysia.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. ALBERT: A lite BERT for self-supervised learning of language representations. *CoRR*, abs/1909.11942.

Bing Liu. 2012. Sentiment analysis and opinion mining. volume 5.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. CamemBERT: a tasty French language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, Online. Association for Computational Linguistics.

Milad Moradi and Matthias Samwald. 2021. Evaluating the robustness of neural language models to input perturbations. *CoRR*, abs/2108.12237.

Jakub Náplava, Martin Popel, Milan Straka, and Jana Straková. 2021. Understanding model robustness to user-generated noisy texts. In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, pages 340–350, Online. Association for Computational Linguistics.

Dat Quoc Nguyen and Anh Tuan Nguyen. 2020. Phobert: Pre-trained language models for vietnamese. *CoRR*, abs/2003.00744.

Jude Obi. 2023. A comparative study of several classification metrics and their performances on data. *World Journal of Advanced Engineering Technology and Sciences*, 8:308–314.

Keiron O'Shea and Ryan Nash. 2015. An introduction to convolutional neural networks. *CoRR*, abs/1511.08458.

Kseniia Palin, Anna Maria Feit, Sunjun Kim, Per Ola Kristensson, and Antti Oulasvirta. 2019. How do people type on mobile devices? observations from a study with 37,000 volunteers. In *Proceedings of the 21st International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '19, New York, NY, USA. Association for Computing Machinery.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 79–86. Association for Computational Linguistics.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2020. Adapterfusion: Non-destructive task composition for transfer learning. *CoRR*, abs/2005.00247.

Dwi Ismiyana Putri, Ari Nurul Alfian, Mardi Yudhi Putra, and Putro Dwi Mulyo. 2024. Indobert model analysis: Twitter sentiments on indonesia's 2024 presidential election. *Journal of Applied Informatics and Computing*, 8(1).

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI*. Accessed: 2024-11-15.

Md. Mostafizer Rahman, Ariful Islam Shiplu, Yutaka Watanobe, and Md. Ashad Alam. 2025. Roberta-bilstm: A context-aware hybrid model for sentiment analysis.

Marco Túlio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with checklist. *CoRR*, abs/2005.04118.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. 1986. Learning representations by back-propagating errors. *nature*, 323(6088):533–536.

Robin M. Schmidt. 2019. Recurrent neural networks (rnns): A gentle introduction and overview. *CoRR*, abs/1912.05911.

Naoki Shibayama and Hiroyuki Shinnou. 2021. Construction and evaluation of Japanese sentence-BERT models. In *Proceedings of the 35th Pacific Asia Conference on Language, Information and Computation*, pages 731–738, Shanghai, China. Association for Computational Lingustics.

James N. Sneddon. 2003. *The Indonesian Language: Its History and Role in Modern Society*. A UNSW Press book. UNSW Press.

Maggie Stack. 2005. Word order and intonation in indonesian. In *University of Wisconsin-Madison LSO Working Papers in Linguistics 5: Proceedings of WIGL 2005*, pages 168–182. University of Wisconsin-Madison.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune BERT for text classification? *CoRR*, abs/1905.05583.

Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. 2011. Lexicon-based methods for sentiment analysis. *Computational Linguistics*, 37(2):267–307.

Uri Tadmor. 2009. Loanwords in indonesian. In Martin Haspelmath and Uri Tadmor, editors, *Loanwords in the World's Languages. A Comparative Handbook*, page 686–716. Mouton de Gruyter, Berlin.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Belinda Ekharisti Viklous. 2022. Perubahan bahasa dan makna kata “anjir” di social media: Kajian sosiolinguistik. *Multidisiplin West Science*, 1(2):213–225.

Mayur Wankhade, Annavarapu Chandra Sekhara Rao, and Chaitanya Kulkarni. 2022. A survey on sentiment analysis methods, applications, and challenges. *Artif. Intell. Rev.*, 55(7):5731–5780.

Bryan Wilie, Karissa Vincentio, Genta Indra Winata, Samuel Cahyawijaya, Xiaohong Li, Zhi Yuan Lim, Sidik Soleman, Rahmad Mahendra, Pascale Fung, Syafri Bahar, and Ayu Purwarianti. 2020. Indonlu: Benchmark and resources for evaluating indonesian natural language understanding. *CoRR*, abs/2009.05387.

Wilson Wongso, David Samuel Setiawan, Steven Limcorn, and Ananto Joyoadikusumo. 2024. Nusabert: Teaching indobert to be multilingual and multicultural.

Lei Zhang, Shuai Wang, and Bing Liu. 2018. Deep learning for sentiment analysis : A survey. *CoRR*, abs/1801.07883.

List of Figures

List of Listings

List of Tables