BERT for Sentiment Analysis of Japanese Twitter

Jordan W. Klein

A dissertation submitted in partial fulfilment of the requirements for the degree of

Master of Arts

Institute of Linguistics and Language Technology

University of Malta

July 31, 2024

L-Università ta' Malta

FACULTY/INSTITUTE/CENTRE/SCHOOL Institute of Linguistics and Language Technology

DECLARATIONS BY POSTGRADUATE STUDENTS

Student's Code [redacted]

Student's Name & Surname Jordan Klein

Course Master of Arts in Linguistics

Title of Dissertation
BERT for Sentiment Analysis of Japanese Twitter

(a) Authenticity of Dissertation

I hereby declare that I am the legitimate author of this Dissertation and that it is my original work.

No portion of this work has been submitted in support of an application for another degree or qualification of this or any other university or institution of higher education.

I hold the University of Malta harmless against any third party claims with regard to copyright violation, breach of confidentiality, defamation and any other third party right infringement.

(b) Research Code of Practice and Ethics Review Procedures

I declare that I have abided by the University's Research Ethics Review Procedures. Research Ethics & Data Protection form code MAKS-2024-00087.

As a Master's student, as per Regulation 58 of the General Regulations for University Postgraduate Awards, I accept that should my dissertation be awarded a Grade A, it will be made publicly available on the University of Malta Institutional Repository.

[redacted]
Signature of Student

JORDAN KLEIN
Name of Student (in Caps)

July 31, 2024
Date

30.01.2020

Acknowledgements

I would like to thank my primary supervisor, Dr. Tanti, who introduced me to Python programming and deep learning. Dr. Tanti was an ideal supervisor that was always available and responsive, but also allowed me the freedom to explore. I am also grateful to Dr. Assimakopoulos, my first contact at the University of Malta, who recommended that I join an intensive AI program hosted by SEA-EU, a formative experience.

My co-supervisor, Dr. Marty, encouraged me to take up R and, alongside Dr. Grech, ensured the project began with a robust methodology. I would also like to thank the German Research Center for Artificial Intelligence for their generous allocation of computing resources for the expensive process of pre-training.

Lastly, I appreciate the support from KSU, which allocated funds from the KSU Education Fund to pay crowdworkers for annotating the sentiment training set, JTS1k.

Abstract

This publication introduces novel, open-source resources for sentiment analysis on Japanese Twitter. **BERT for Japanese Twitter** is a pre-trained model that is highly competent in the target domain and adaptable to a variety of tasks. **Japanese Twitter Sentiment 1k (JTS1k)** is a compact sentiment analysis dataset optimized for balance and reliability. This combination of pre-trained model and dataset was used to fine-tune a sentiment analysis model that broadly applies to Japanese social networking services (SNS): **BERT for Japanese SNS Sentiment**. The primary focus of this project is domain adaptation. Using an established Japanese BERT model as a foundation, domain adaptation was achieved by optimizing the vocabulary and continuing pre-training on a large Twitter corpus. Similar methodology was used to develop Twitter Multilingual RoBERTa (XLM-T) (Barbieri et al., 2022), which is the state-of-the-art multilingual Twitter model. By using a monolingual approach, this study developed a more efficient model that outperformed XLM-T in the target language. This project explored fundamental elements of corpus construction, corpus refinement, dataset annotation, preprocessing, pre-training, fine-tuning, and benchmarking. It concludes with a demonstration that the sentiment model is valid, useful, and sensitive to changes in public sentiment that correlate with real-world events.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

This thesis presents the development of a large language model (LLM) that is specialized for sentiment analysis on Japanese Twitter. The topic was chosen for its mix of challenge and feasibility. The author of this thesis, having lived in Japan for three years, has knowledge of the language and culture, which is supplemented by a personal network of native Japanese speakers. Japan's role as a leader in natural language processing (NLP) and the availability of open-source resources further supported this project. Sentiment analysis is a classic benchmarking task for NLP, and Twitter specialized sentiment models have been developed in several languages. Although Japan is one of Twitter's largest user bases, no such model has been developed for Japanese. This research utilized a substantial corpus from the Twitter API, providing for rich corpus analysis alongside the overall aims of this project. This introductory chapter begins by defining sentiment analysis and outlines the approach used in this project. Ethical considerations follow, focusing on data handling and privacy. Various challenges specific to analyzing sentiment on Japanese Twitter are discussed. The introduction concludes with a summary of the thesis structure.

1.1: Sentiment Analysis on Social Media

Sentiment analysis, alternatively called opinion mining, is practiced by businesses, academia, and policymakers who aim to understand public sentiments as expressed through language. This field utilizes NLP to analyze large volumes of data from digital platforms where users frequently share their opinions (Liu, 2012). The rise of social media has significantly boosted the relevance of sentiment analysis by providing vast amounts of data on public opinion in real time (Dave et al., 2003). Organizations use sentiment analysis to evaluate consumer reactions and adjust their strategies accordingly (McGlohon et al., 2010). Sentiment analysis helps businesses understand customer satisfaction and guide product development (Hong & Skiena, 2010). In social media contexts, sentiment analysis helps identify trends and shifts in public mood, which is informative for marketing and public engagement (Mohammad, 2012). Therefore, sentiment analysis is a powerful tool for any entity interested in gauging and responding to public opinion online.

1.2: BERT for Japanese Twitter Sentiment

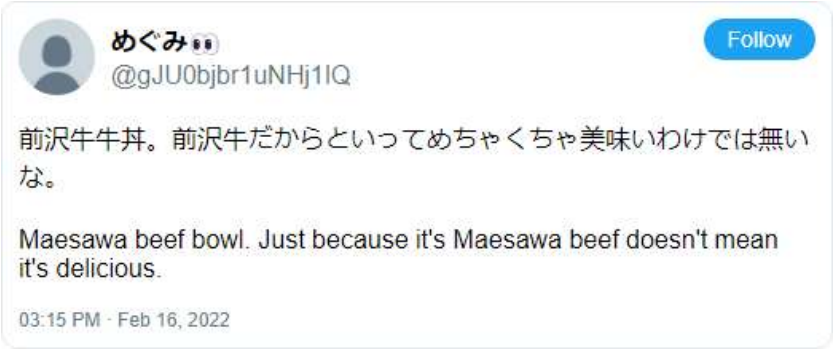
Sentiment analysis operates at three levels: document, sentence, and aspect (Liu, 2012). Document-level analysis assesses the overall sentiment of entire texts, such as reviews or articles. Sentence-level analysis classifies the sentiment of individual sentences, while aspect-level analysis

focuses on the specific subjects of opinions and their sentiments.[1] This project concentrates on document-level analysis, classifying tweets into one of four categories—positive, negative, neutral, or mixed. Classifications represent the feelings of the author of the tweet. The four categories are exemplified in Figures 1.1 to 1.4.

Figure 1.1 Example of a Positive Tweet

Positive tweets express emotions such as joy, satisfaction, optimism, or anticipation. Mixed emotions are considered positive if the positive emotions are dominant.

Figure 1.2 Example of a Negative Tweet



Negative tweets express emotions such as anger, sadness, disappointment, or fear. Mixed emotions are considered negative if the negative emotions are dominant.

Current state-of-the-art models utilize contextualized encodings derived from Transformers (Vaswani et al., 2017). The BERT architecture, which relies solely on Transformers, has proven to be highly effective across tasks (Devlin et al., 2019). This thesis introduces **BERT for Japanese Twitter**, which was adapted from Japanese BERT by continuing pre-training on a Twitter corpus. A lightweight training set, **Japanese Twitter Sentiment 1k (JTS1k)**, was developed for the target task. JTS1k, in conjunction with another dataset, were used to fine-tune **BERT for Japanese SNS Sentiment**, a

[1] "I'm feeling great about the election! Trump is going down 😊" The overall sentiment is positive. Aspect-level analysis elaborates that sentiment about Trump is negative.

sentiment classifier that generalizes across social network services (SNS). This approach leverages transfer learning, following a leading paradigm in NLP (Ruder et al., 2019).

1.3: Ethical Considerations

This project used the Twitter API to access tens of millions of tweets for training models. The data included identifiable information like usernames and profile details. Twitter's terms and conditions limit sharing to ID numbers only, although limited exchanges of text and user data among colleagues is allowed. In line with these rules, this project released its training dataset using only IDs and excluded any text data. This thesis occasionally references specific Twitter posts. While these posts are genuine, identifying details such as usernames and timestamps have been substituted with random generations. The ethical standards in research stress the importance of consent when using identifiable data (Association of Internet Researchers, 2019; Sloan et al., 2020). This thesis acknowledges the need for consent and complies with Twitter's guidelines.

Figure 1.3 Example of a Neutral Tweet

Neutral tweets express balanced or indifferent emotions, with no strong positive or negative sentiments. Examples include stating a fact, asking a question, or seeking a recommendation.

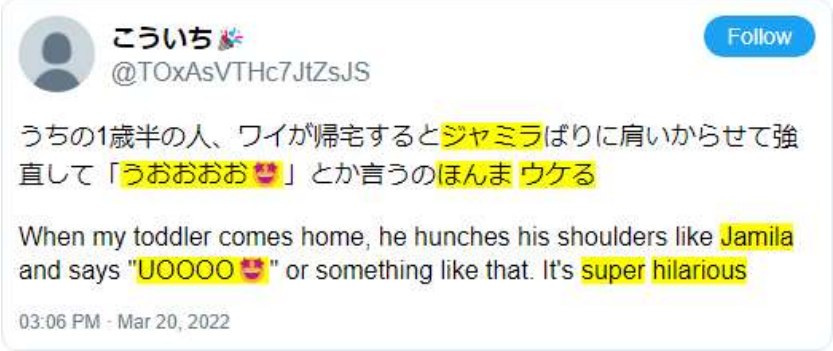
Figure 1.4 Example of a Mixed Tweet

Mixed tweets express both a clear positive and negative opinion, and it is unclear which one is dominant. It might be a balanced presentation of pros and cons, or the simultaneous expression of conflicting emotions.

1.4: Challenges of Japanese Twitter

Sentiment analysis on Twitter is challenging due to its diverse language usage, which prominently includes slang, abbreviations, misspellings, and emoticons. These elements can significantly vary in meaning depending on the context and the user (Poria et al., 2023). Additionally, Twitter users often mention named entities, such as brands, people, and locations, which may be central to the underlying sentiment. Moreover, traditional linguistic distinctions like dialectical variation and mimesis are richly represented on social media. This can pose a challenge for BERT models, which are trained and more formal and structured sources like Wikipedia, leaving them less accustomed to the distinctive and impactful language features found on platforms like Twitter.

Figure 1.5 Example of a Tweet with Marked Features



This brief tweet exemplifies several types of expressions that pose challenges for models. ジャミラ refers a giant monster from the popular series Ultraman. うおおおお represents an onomatopoeia for Jamila's roar in Japanese. ほんま is an emphatic term commonly used by speakers of the Kansai dialect, and ウケる, which translates to hilarious, is general slang.

Multilingualism adds another layer of difficulty. Japanese Twitter users frequently use English, Chinese, Korean, and other languages. In addition, Twitter users borrow characters from alternative scripts for producing kaomoji[2]. Characters from 55 unique languages were observed across the Twitter corpus. The complexity of model design increases with the need to accommodate multiple language scripts as well as non-standard characters like emojis. The model has a limited vocabulary that is constrained for efficiency. It should be comprehensive enough to cover fundamental Japanese as well as common colloquialisms, neologisms, and multilingual expressions from Twitter. At the same time, it must remain adaptable to new, unseen terms.

[2] (^ω^), Kaomoji translates to “face mark.” This original emoticon became popular with the rise of SMS (Bedrick et al., 2012)

One of the primary limitations of NLP approaches to sentiment analysis is the loss of context to preprocessing. Tweets are marked by a date, an author, and are bound in conversation by replies and hashtags. Furthermore, Twitter is a multimodal platform that allows users to enrich their posts with media content, including stylized links, photos, and video. In preparing model inputs, the text data is stripped from these critical contexts, shuffled, and transformed into machine-readable form. Sarcasm and irony are particularly challenging because they often rely on subtleties that the model is not exposed to. Current research directions, such as multimodal sentiment analysis, sarcasm awareness, and temporal alignment, are addressing these shortcomings (Poria et al., 2023; Lai et al., 2023; Loureiro et al., 2022). This project overlooks these complexities, solely focusing on the text-based representation of sentiment.

Figure 1.6 Example of an Ironic Tweet

This tweet was posted with an image of the bio page of a passport, which has been completely covered in scribblings and doodles in permanent marker. The incomplete narrative from the text hints at irony. Without the image, this tweet would be classified as either neutral or positive. With the image, a case could be made for any of the four sentiment classes.

The communication style in Japanese often uses indirect expressions, making it hard for text-based models to interpret emotions accurately. A fundamental concept in Japanese interaction is 空気を読む (kuuki wo yomu), meaning "read the air." This involves assessing and aligning with the group's mood to maintain harmony. For example, it is customary in Japanese etiquette to modestly deflect compliments to avoid appearing boastful. Emotions, particularly strong ones, are more safely conveyed through factual statements and implicatures rather than direct expressions. While platforms like Twitter provide some anonymity, which invites more direct emotional expression, the traditional indirect style heavily influences communication. This causes models to misclassify tweets, with consistent confusion between negative and neutral sentiments.

Figure 1.7 Example of Negative Tweet with Postive Language

A fan comments on their team's losing streak, and they use a structure intended for comparison of positive and negative points. By only mentioning a weak positive, overall negative sentiment is implied. This example could potentially fit any of the four sentiment categories.

A final consideration for working with the Twitter domain is the abundance of automatically generated content. Opinion spamming and content factories are recognized pollutants of web corpora. Additionally, public entities, marketers, and third-party developers have created various tools to interact with the public *en masse*. Although these functions are innovative and useful, they generate a considerable amount of repetitive text, which poses challenges for language models that favor diverse and meaningful data. Such repetitive texts not only bloat the training material but also risk promoting rote memorization of robotic phrases by language models. This situation can render language models less flexible and less capable of making human-like, context appropriate inferences. Therefore, it is essential to filter out this monotonous content to preserve the quality and effectiveness of the models (Lee, et al., 2022).

1.5: Thesis Structure

This thesis is structured into twelve chapters, with each chapter focusing on a narrow objective. **Chapter 2** presents the research aims and objectives, summarizing the research questions, hypotheses, and evaluation methods. **Chapter 3** is the literature review. **Chapter 4** details the methodology used for acquiring data via the Twitter API. **Chapter 5** addresses the issues of repetitive text and user imbalance in the training corpus. **Chapter 6** describes the construction of the **JTS1k** training set. **Chapter 7** focuses on the tokenizer design and explores the change in vocabulary across Twitter adaptation. **Chapter 8** describes the initial pre-training and evaluation of **BERT for Japanese Twitter. Chapter 9** searches for opportunities to utilize transfer learning to train a more useful sentiment classifier. **Chapter 10** fine-tunes **BERT for Japanese SNS Sentiment**, which is benchmarked on the target task against a series of top-tier models. **Chapter 11** demonstrates sentiment analysis using the target model. **Chapter 12** concludes.

Chapter 2: Aims and Objectives

The main goal of this project is to produce novel, open-source tools for sentiment analysis on Japanese Twitter. This thesis details the development of three components. **BERT for Japanese Twitter** is a pre-trained model that performs well across various tasks within the target domain. The **Japanese Twitter Sentiment 1k (JTS1k)** dataset trains models to analyze sentiment in tweets. **BERT for Japanese SNS Sentiment** was fine-tuned using a combination of datasets that broadly represents social networking services (SNS). All the deliverables were designed with the intent to optimize three qualities: usefulness, robustness, and accessibility.

BERT for Japanese Twitter is the first publicly available model tailored for this language and domain. It offers a flexible fine-tuning checkpoint that saves developers time and resources. It has demonstrated superior performance in a range of tasks beyond sentiment analysis and is accessible as an open-source tool in a public repository. BERT for Japanese Twitter was adapted from a base sized BERT model, which has a relatively small parameter footprint that lowers hardware requirements and broadens user accessibility.

JTS1k is a well-balanced and reliably annotated dataset designed for training and evaluating current generation models. Its IDs and labels are published in an open repository. To access text data, developers must use the Twitter API.

BERT for Japanese SNS Sentiment is efficient and cost-effective. It performs sentiment analysis accurately, with applications for tracking public sentiment on social media. Like the initial model, it has a minimal parameter footprint, reducing hardware demands. This model is also available in an open repository, facilitating widespread use and collaborative enhancements.

The central focus of this project is domain adaptation. It specifically applies the method advocated by Gururangan et al. (2020), which necessitates a large corpus within the target domain. Acquiring sufficient training material can be challenging, and even when available, constructing a balanced and representative sample is complex. Moreover, even with a suitable training corpus, continued pre-training demands significant time and resources. This project employs this resource-intensive approach for domain adaptation in a context where training data and resources are relatively unconstrained. It validates this method by yielding a fine-tuned model with superior performance in the target domain and task.

2.1: Research Questions

- What are the essential qualities of a training corpus for unsupervised pre-training? Which sampling and preprocessing methods produce the best corpus?
- What characteristics define an effective sentiment analysis training set? What are the best practices for directing crowdworkers to ensure high-quality annotations?
- Does domain adaptation result in a model that outperforms on Twitter-specific tasks? How does it compare with larger, general-purpose models? Does the Twitter-adapted model maintain general task proficiency? Was the investment in continued pre-training justified?
- Does the fine-tuned sentiment model perform as expected in real-world applications?

2.2: Hypotheses

- Using language, time, and place filters in the Twitter API to formulate queries will create a balanced and representative training corpus.
- Refining the training corpus through eliminating repetitive text and balancing user contributions will enhance training outcomes by removing spam, increasing linguistic diversity, and boosting efficiency.
- The relatively small JTS1k dataset will achieve comparable or better results to larger datasets because it is optimized for balance and reliability.
- An XLM-T model fine-tuned on JTS1k will be competent in multilingual Twitter sentiment analysis.
- The vocabulary yielded by the data-driven WordPiece algorithm will include colloquialisms, neologisms, and multilingual expressions, while excluding formal and domain-specific items such as literary, medical, and historical terms.
- BERT for Japanese Twitter will excel in Twitter tasks and surpass state-of-the-art models in Japanese Twitter sentiment. Compared to larger, more general models, it will be less costly and more efficient.
- BERT for Japanese Twitter will show reduced proficiency in general domain tasks, but the decline will not be severe.
- Combining the JTS1k and WRIME datasets will train a more robust sentiment classifier that will broadly apply to Japanese SNS.
- BERT for Japanese SNS Sentiment will make predictions that align with expectations. It will be sensitive to shifts in public sentiment that correlate with real-world events.

2.3: Evaluation Methods

- The **pre-training corpus** is assessed based on size, balance, and diversity. The initial raw Twitter corpus includes over 60 million tweets, though refinement processes reduce this number significantly. This project experiments with two training corpora, differing in size and refinement levels. Corpus content is compared using n-grams, mentions, and hashtags. The superior training corpus yields better performing models.
- **JTS1k** consists of a thousand examples that are optimized for balance and reliability. Each example is labelled by three annotators, with inter-annotator agreement measured by Krippendorff's alpha. The dataset's trainability is validated through benchmarking a series of generative AI, and its general representation of the Twitter domain is supported by exploring cross-lingual transfer.
- **BERT for Japanese Twitter** is fine-tuned on tasks related to social media to demonstrate enhanced performance within the target domain. Fine-tuning with JGLUE (Kurihara et al., 2022), which measures general language understanding, evaluates the retention of general knowledge. The embedding matrices of the original and Twitter adapted BERT are compared with t-SNE projections. Human annotators assess the acceptability of masked token predictions.
- **BERT for Japanese SNS Sentiment** is benchmarked against a series of state-of-the-art models on the targeted task of sentiment analysis. The thesis concludes with a demonstration that the model's predictions are both valid and practical, confirming its effectiveness in real-world applications.

2.4: Deliverables

- **BERT for Japanese Twitter** and **BERT for Japanese SNS** will be published on HuggingFace with documentation that supports their usability, validity, and reliability.
- **JTS1k** will also be published on HuggingFace. In compliance with Twitter policy, text data will not be shared openly.

Chapter 3: Literature Review

The literature review begins by introducing Bidirectional Encoder Representations from Transformers (BERT), exploring its architecture, training methodology, and the reasons behind its effectiveness. The discussion moves to tokenization, comparing the strategies used in contemporary Japanese models. The next section provides an overview of Japanese encoder and decoder models and justifies the choice of foundation model. The state-of-the-art Twitter models are introduced, and their capabilities are discussed. The methodology used for domain adaptation is based on the training procedure for Twitter Multilingual RoBERTa (XLM-T) (Barbieri et al., 2022). Adapting Japanese BERT will require a large training corpus. The composition and sampling methodology of comparable Twitter corpora are explored to ensure robust methodology for corpus construction. This study aims to train a specialized Japanese Twitter sentiment model that surpasses the state-of-the-art in performance and efficiency. The models and datasets will be useful contributions to the study of multilingual social media. The literature review concludes with an outline of the models and datasets used for building and benchmarking.

3.1: BERT Architecture and its Utility

BERT uses word embeddings to store word-level knowledge, which is an established strategy in NLP (Mikolov et al., 2013). Embeddings are vector representations that capture the semantic properties of words. They are learned from large text corpora by training models to understand contexts or predict words. Language models have a fixed vocabulary which is collectively represented by the embedding matrix. During training, the embedding matrix is continually updated so that words used in similar contexts will cluster in vector space. Similar words have similar embeddings, but over the multidimensional encoding process, words may be near in one context and distant in another. The size of the vocabulary and the dimensions of the embeddings are critical design considerations that affect the model's learning capacity and efficiency.

BERT is an encoder model that utilizes the Transformer architecture to process word sequences into contextualized encodings (Devlin et al., 2019; Vaswani et al., 2017). The Transformer takes an input that represents words and their order, which is then passed to the self-attention layer. Self-attention, which determines the relationship between words, is the essential feature of the Transformer. Each Transformer layer has several attention heads that consider word relationships from alternative perspectives. BERT processes encodings across multiple Transformer layers, enabling it to capture complex syntactic structures and long-range dependencies. Given an appropriately large

and diverse training corpus, Transformers have an immense capacity to take on representations of words in context. Furthermore, it processes sequences simultaneously, which is much faster than sequential processing techniques. Key elements of model deisgn, such as the number of Transformer layers, the number of self-attention heads per layer, and the size of hidden layers, are configured to optimize performance and efficiency.

The target model was adapted from the latest Japanese BERT model developed by the Tohoku NLP group[3]. Their training corpus combined Japanese Wikipedia with the Japanese portion of the CC-100 (Conneau et al., 2020), totaling 426 million sentences. Tohoku NLP uses the same pre-training objectives as the original BERT: masked language modelling (LM) and next sentence prediction (NSP) (Devlin et al., 2019). In masked LM, BERT randomly masks a percentage of the input tokens and learns to predict the original token based on the context provided by the non-masked words. For NSP, the model learns to predict whether two segments of text naturally follow one another, which aids in understanding sentence relationships and coherence. This combination of tasks is intended to help the model acquire a deep, bidirectional understanding of language structure and use. The current generation of encoder models continue to rely on masked LM for pre-training, but NSP has fallen out of favor (Zhuang et al., 2021) Japanese BERT was adapted for Twitter by continuing the masked LM objective with the Twitter corpus.

BERT models are designed to take advantage of transfer learning (Ruder et al., 2019). Pre-training is resource-intensive, demanding days or weeks in a high-performance computing environment. The pre-trained model provides a general checkpoint for fine-tuning on specific tasks. Fine-tuning is less demanding, usually completed in minutes or hours on a single GPU. BERT models excel in transfer learning because the pre-training objectives tackle the difficult task of acquiring language. The Transformer architecture is adaptable, accommodating different tasks through the addition of a single trainable layer. Fine-tuned BERT models achieve near state-of-the-art, even with smaller training sets (Devlin et al., 2019).

3.2: Challenges and Strategies in Tokenizing Japanese

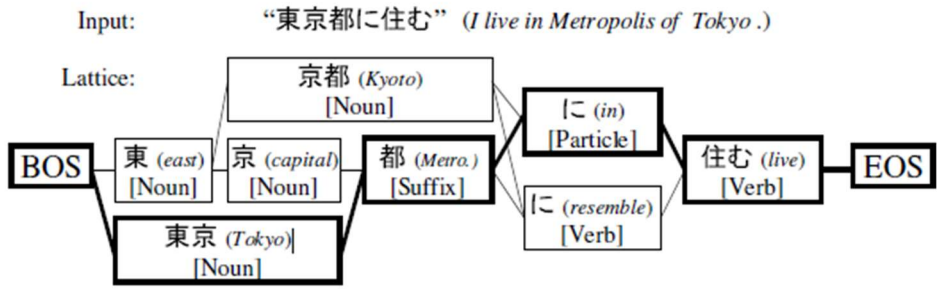
Tokenizing Japanese is a longstanding issue because the language does not use spaces to separate words. Context-aware language understanding is required to accurately identify token boundaries. The Japanese tokenizer, MeCab, addresses this by applying conditional random fields (Kudo et al., 2004). The process starts by decomposing the input into all possible token sequences

[3] Tohoku NLP Github: https://github.com/cl-tohoku/bert-japanese

with part of speech tags. MeCab then determines the most likely token boundaries by analyzing context both forward and backward across n-gram segments.

This method of combining contextual analysis with parts of speech results in a highly accurate tokenizer that is capable of handling new words. MeCab requires substantially more computational power than trivial whitespace tokenizers. Despite this, it excels in large-scale applications. MeCab remains one of the most effective tools for parsing Japanese, setting a high benchmark for accuracy and efficiency in NLP tasks (McCann, 2020).

Figure 3.1 Conditional Random Fields for Tokenizing Japanese



Lattice of all possible token boundaries for the input, 東京都に住む. For a Japanese speaker, the intended segmentation, 東京、都、に、住む, is clear. Otherwise, there are many acceptable possibilities.

To address the constraints imposed by a fixed vocabulary, BERT tokenizers incorporate subword tokens. With Japanese language models, three approaches to subword tokenization are prevalent: character, WordPiece, and SentencePiece. Both WordPiece and SentencePiece methods use a training corpus to develop a vocabulary that optimizes coverage and flexibility. The character tokenizer, on the other hand, divides text into individual characters, ensuring broad token coverage with a minimal vocabulary size. The main drawback of character tokenizers is their propensity for longer sequence lengths. The space complexity of a training step grows quadratically with length, resulting in slower and more resource-intensive training (Al-Rfou et al., 2018). Within the Tohoku NLP family of Japanese BERT models, character models consistently underperform compared to WordPiece models[4].

This study developed a modified WordPiece tokenizer based on the latest version of Japanese BERT (Devlin et al., 2019). The tokenizer begins by pre-tokenizing text, segmenting sequences into words using MeCab. Subsequently, out-of-vocabulary words are further segmented into subword

[4] Performance of Tohoku NLP Models on JGLUE: https://github.com/cl-tohoku/bert-japanese

tokens. This methodology enables the representation of problematic elements like rare words, misspellings, and stylistic variations.

Example 3.1: Segmentation of a Rare Word by a WordPiece Tokenizer

Input: "antidisestablishmentarianism is a long word"

Output: [anti, '##dis', '##esta', '##b', '##lish', '##ment', '##arian', '##ism', 'is', 'a', 'long', 'word']

WordPiece segments rare words into meaningful subword tokens. Subword tokens lead with double hashmarks: '##'.

SentencePiece, developed by Dr. Kudo of MeCab, is also highly effective (Kudo et al., 2004; Kudo et al., 2018). Unlike WordPiece, SentencePiece processes raw text directly. Whitespace is treated as a normal character, and it may be joined with other characters into a token unit. The algorithm prioritizes common words and subwords, but it can also form phrase-level tokens. Some tasks may favor one tokenizer over the other. WordPiece is a more straightforward choice for tasks where explicit word boundaries matter, such as dependency parsing and extractive question answering (Kurihara et al., 2022). On the other hand, WordPiece tends to split named entities inappropriately, and recomposing them can be problematic. SentencePiece better preserves the original format of text, spacing, and punctuation, which is advantageous for named entity recognition (Yamada et al., 2020). Furthermore, SentencePiece is language agnostic, making it an excellent choice for multilingual language models (Conneau et al., 2020). Most tasks, including sentiment analysis, work perfectly well on either tokenizer.

3.3: Overview of Japanese Encoder and Decoder Models

The original Transformer architecture, designed for sequence-to-sequence translation, features both an encoder and a decoder (Vaswani et al., 2017). BERT, as an encoder model, focuses on processing input data to create representations useful for various natural language processing tasks. Since the advent of BERT, significant advancements in training methods and architecture have emerged. RoBERTa improves upon BERT by significantly increasing the amount of training data, removing NSP, and optimizing the training process (Zhuang et al., 2021) LUKE builds on RoBERTa with a masked LM task focused on entities (Yamada et al., 2020). DeBERTa introduced a disentangled attention mechanism that separately considers word position and content (He et al., 2020). MegatronBERT specializes in parallel training across multiple GPUs, supporting the development of

much larger models (Shoeybi et al., 2019). DistilBERT maintains the efficacy of larger models while reducing the model size through a student-teacher training paradigm (Sanh et al., 2019). Big Bird adapted the Transformer architecture to handle longer documents a novel attention mechanism approach (Zaheer et al., 2020).

Table 3.1 Japanese Encoder Models Available on HuggingFace

Affiliation	Developer	Archetype	Parameters	Publication Date
Company	Colorful Scoop	BERT	111M	2021 Q3
	LINE	distilBERT	68.1M	2023 Q1
	rinna	RoBERTa	111M	2021 Q4
	Studio Ousia	LUKE	279.1M	2022 Q4
			559.9M	2022 Q4
		LUKE lite	133.1M	2022 Q4
			413.8M	2022 Q4
	レトリバ	MegatronBERT	1.3B	2024 Q2
University	Izumi Lab.	BERT	17.8M	2021 Q4
		DeBERTa	110.0M	2023 Q4
			17.8M	2023 Q4
	Kawahara Lab	BigBird	113.4M	2023 Q2
		RoBERTa	110.6M	2022 Q4
			336.7M	2022 Q4
	Koichi Yasuoka	DeBERTa	124.6M	2022 Q2
			386.5M	2022 Q2
		RoBERTa	278.3M	2021 Q4
			560.2M	2021 Q4
	KU NLP	DeBERTa	160.0M	2024 Q2
			338.8M	2023 Q1
		RoBERTa	100.2M	2022 Q3
			322.7M	2022 Q3
	Tohoku NLP	**BERT**	**111.2M**	**2023 Q2**
			337.4M	2023 Q2

The archetype provides information about the model architecture and training configuration. The number of parameters represents the size. Larger models offer the potential of enhanced performance, but they are costly to train and operate. The model selected for Twitter adaptation is highlighted in bold.

The Tohoku NLP BERT model was selected based on three criteria:

- **Established Performance**: Over three generations, developers have improved the model's performance, incorporating a larger training corpus, dynamic masking, and whole-word

masking. The Japanese BERT model is competitive, achieving near or state-of-the-art performance across tasks, as per JGLUE evaluations (Kurihara et al., 2022).

- **Ease of Use**: Initially, the Japanese BERT from Tohoku NLP was the most accessible. They developed the *BertJapaneseTokenizer* within the *Transformers* library, facilitating easier preprocessing. Moreover, their English documentation was superior. Other groups have recently improved tokenization and documentation.
- **Parameter Efficiency.** The base BERT model has one of the lowest parameter counts, which minimizes resource consumption and enhances accessibility for a wider audience. This project aimed to train a model that outperformed larger models on Twitter tasks.

The decoder architecture, which powers advanced generative AI (GenAI) models, has recently gained considerable attention. Table 3.2 compares the Japanese GenAI models available on HuggingFace, showing significant private sector investments. The industry values GenAI for their practical applications in areas such as chatbots, information retrieval, and content generation. One of their major advantages is adaptability. They can be configured for various tasks through prompt tuning with minimal examples required (Brown et al., 2020). However, GenAI models are large and costly to operate. Encoder models are less flexible and require more training examples, but they are faster and more efficient. This project aims to justify the trade off in flexibility by training a specialized classifier that surpasses state-of-the-art GenAI in performance. In Chapter 6, the study benchmarks a series of GenAI models on Japanese Twitter sentiment, including state-of-the-art multilingual models and a few Japanese Llamas.

3.4: State-of-the-Art Encoder Models for Twitter

Although this changed in 2023, Twitter has a long history of allowing researchers free access to its data. This abundance of data has attracted researchers aiming to develop techniques for domain-specialized natural language understanding (Nakov, et al., 2016). Transformer models have been extremely successful in working with Twitter data. Initially, the focus was on English Twitter, with BERTweet (Nguyen et al., 2020) emerging as the state-of-the-art model. More recently, there has been increased attention on languages other than English. Barbieri et al. (2022) attempted to dethrone BERTweet by leveraging cross-lingual transfer with Twitter Multilingual RoBERTa (XLM-T). Within English, BERTweet outperforms XLM-T in most English tasks, but XLM-T is state-of-the-art for multilingual Twitter understanding.

Table 3.2 Japanese Decoder Models Available on HuggingFace

Affiliation	Developer	Archetype	Parameters	Publication Date
Company	ABEJA	GPT2	750.1M	2022 Q3
		GPT-NeoX	2.7B	2022 Q3
		Mixtral	8x7B	2024 Q2
	ELYZA	Llama 2	13B	2023 Q4
			7B	2023 Q3
		Llama 3	8B	2024 Q2
	Lightblue KK.	**Llama 3**	**8B**	**2024 Q2**
	LINE	LINE	1.7B	2023 Q3
			3.6B	2023 Q3
	rinna	GPT	1B	2022 Q1
		GPT2	123M	2021 Q2
			361M	2021 Q2
			43.7M	2021 Q3
		GPT-NeoX	204M	2022 Q3
			3.6B	2023 Q2
		Llama 2	7B	2023 Q4
		Llama 3	8B	2024 Q2
		Qwen	14B	2023 Q4
			7B	2023 Q4
	stability.ai	Llama 2	7B	2023 Q4
			70B	2023 Q4
Independent	**@alfredplpl**	**Llama 3**	**8B**	**2024 Q2**
	@haqishen	**Llama 3**	**8B**	**2024 Q2**
University	Kawahara Lab	GPT2	110.4M	2022 Q1
			1.5B	2022 Q4
	KU NLP	GPT2	90M	2023 Q2
			310M	2023 Q4
			717M	2023 Q4
	TokyoTech-LLM	Llama 3	8B	2024 Q2

Compared to the encoder models, decoder models are much larger, which makes pre-training extremely expensive. Archetypes differ in architecture and training material. All models are strong in English. Mixtral is optimized for European languages, and Qwen specializes in Chinese. Each of these models have continued pre-training with Japanese.

3.5: Role Models in Corpus Construction

This study follows the methodology of XLM-T by starting with an established model and continuing pre-training on a Twitter approach requires an extremely large training corpus that generally represents the target language. In preparation for collecting data, a role model in corpus

construction was sought. Along with XLM-T, another noteworthy project was the construction of the Edinburgh Twitter Corpus (Petrović et al., 2010), which served as a valuable reference for developing a robust and representative training corpus.

Edinburgh Twitter Corpus (ETC)

The ETC was introduced as an open-source resource when Twitter was an emerging platform in social media. The corpus was not proposed for training, but it did aim to capture a general representation of language use on Twitter. Tweets were sampled by the streaming functionality of the Twitter API, meaning production and collection occurred simultaneously. Sampling was carried out over two months, yielding a corpus of nearly 100 million tweets contributed by approximately 10 million authors. Analysis of word frequency, hashtags, and user mentions suggest that Twitter in 2010 was predominantly English, and topics within Western pop culture are heavily represented. The ETC provided valuable insights into the functionalities of the Twitter API and modeled some useful approaches for corpus analysis. The research aims of XLM-T aligned much closer to this project than the ETC, and their procedure was a greater influence.

Twitter Multilingual RoBERTa (XLM-T)

To build XLM-T, Barbieri et al. (2022) began with Multilingual RoBERTa (XLM-R) and continued pre-training on a corpus of 198 million tweets. The corpus was retrieved from the Twitter archive dating from 2018 to 2020. Comparing their training corpus with the ETC shows how the user base had expanded since its early days. With tweets grouped by language, English remains dominant, constituting a fifth of the corpus body. Japanese accounts for 5% of the corpus with 10 million tweets, and fifteen languages contributed at least a million tweets. The outcome was a rich blend of high-resource languages with some respectable contributions from low-resource languages.

In addition to pre-trained models, their contributions included TweetEval, a multitask, multilingual package of datasets within the Twitter domain (Barbieri et al., 2020). They introduced the Twitter Sentiment Multilingual (TSML) dataset to explore cross-lingual transfer with XLM-T by training on one language and evaluating in another (Conneau et al., 2020).

3.6: BERT for Japanese Twitter

Barbieri et al. (2022) have invited other researchers to utilize their datasets and models to set new benchmarks against monolingual Twitter models. Japanese is one of the best represented languages in the XLM-T training corpus, but it is not explored in TweetEval. This project aimed to

deliver a model that can outperform XLM-T on the target language and a dataset that is compatible with TSML.

This will not be the first Japanese BERT model that is specialized for Twitter. Another model, hottoSNS-bert, was trained from scratch on a corpus of 85 million tweets. This model shows improved performance on Twitter tasks, but relatively poor performance on general domain tasks (Sakaki et al., 2019; Suzuki et al, 2022; Keshi et al., 2017). This model is not publicly available, and it was only discovered recently. This thesis recognizes the contributions by Sakaki et al. (2019), but hottoSNS-bert is not included in the benchmark studies.

The remainder of this literature review outlines the models and datasets used for building and benchmarking. This information will serve as a useful reference in Chapters 8-10, which focus on these elements.

3.7: Models for Pre-Training

Table 3.3 Pre-Trained Models for Building and Benchmarking

Model	Version	Parameters	Vocab	Tokenizer
BERT Japanese	large	337.4M	32.8K	WordPiece
	base	111.2M		
LUKE Japanese	large	559.9M	32.8K	SentencePiece
	base	279.1M		
LUKE-lite Japanese	large	413.8M		
	base	133.1M		
XLM-R	large	559.9M	250.0K	SentencePiece
XLM-T	large	559.9M		
	base	278.0M		

This table compares the architecture of BERT for Japanese Twitter with the other pre-trained models. The other models are based on RoBERTa, which is a bulkier configuration. The multilingual models feature a much larger vocabulary. The Twitter model, XLM-T, inherited its vocabulary from XLM-R.

Japanese BERT: The base version of Japanese BERT was adapted to Twitter, and the large version is used in benchmarking. Japanese BERT provides a lower bound baseline for Twitter tasks and an upper bound for general tasks.

LUKE Japanese: Japanese LUKE models generalize well across many tasks and are popular for fine-tuning. Beginning with the RoBERTa architecture and pre-training objective, LUKE incorporates an additional masked language modelling task that focuses on entities (Zhuang et al., 2021; Yamada et al., 2020). Entity representations are stored in a separate, lower-dimensional embedding matrix. Full

LUKE models include pre-trained entity embeddings from Wikipedia, while LUKE lite models do not. LUKE models provide an upper bound baseline for Japanese, and they generalize well to Twitter.

Multilingual RoBERTa (XLM-R): Developed by Facebook's AI research team, XLM-R extends the RoBERTa architecture to multiple languages (Conneau et al., 2020). Its robust pre-training on a diverse dataset enables strong performance across various languages. XLM-R is meant to be compared with XLM-T to demonstrate the benefits of domain adaptation.

Twitter Multilingual RoBERTa (XLM-T): XLM-T was adapted from XLM-R through continued pre-training on a large Twitter corpus (Barbieri et al., 2022). This specialization makes XLM-T highly effective for sentiment analysis, trend detection, and content moderation in a multilingual social media context. The XLM-T sentiment model was trained on the TSML dataset. This project aimed to develop a specialized Japanese sentiment model that exceeds XLM-T sentiment.

3.8: Datasets for Fine-Tuning

Table 3.4 Datasets for Fine-Tuning

Domain	Dataset	Size	Task	Description
Twitter	JTBR	44k	Text Classification	Sentiment Analysis
	JTDD	4k		Defamation Detection
	TSML	3k x 8		Sentiment Analysis
	SB10k	10k		Sentiment Analysis
Social Media	WRIME	30k	Multi-label Classification	Emotional Intensity & Sentiment Analysis
General	MARC-ja	193k	Text Classification	Sentiment Analysis
	JCoLA	7.8k		Grammatical Acceptability
	JNLI	23k	Sequence Pair Classification	Natural Language Inference
	JSTS	14k		Semantic Textual Similarity
	JSQuAD	67k	Question Answering	Extractive QA
	JCSQA	10k		Multiple Choice QA

Datasets within the social media domain have applications for sentiment analysis, marketing research, and content moderation. They showcase the enhanced capabilities of BERT for Japanese Twitter. The general datasets were sourced from the Japanese General Language Understanding Evaluation (JGLUE) (Kurihara et al., 2022). This diverse group of tasks provides a broad overview of natural language competencies. The Twitter sentiment datasets are compared in a study that investigates cross-lingual transfer.

Writer Reader Intensity Measurement of Emotions (WRIME): This dataset aimed to integrate emotional intensity perceptions of text by authors and readers (Kajiwara et al., 2021). It was developed by enlisting crowdworkers to rate their past posts on social networking services (SNS) for emotional intensity and sentiment polarity (Suzuki et al., 2022). Each post was additionally annotated by three readers. This process created a dataset of 30,000 SNS posts appraised on a four-point scale for each of Plutchik's eight emotions. The publication treats author and reader annotations distinctly,

but this study streamlines the approach by averaging annotations together. The dataset excludes non-standard characters like *emojis*, which enhances its accessibility for models less exposed to such elements.

Tweet Sentiment MultiLingual (TSML): This dataset includes tweets in eight languages: Arabic, English, French, German, Hindi, Italian, Portuguese, and Spanish (Barbieri et al., 2020). It categorizes tweets into three sentiments: negative, neutral, and positive. The dataset is perfectly balanced with 1,011 examples for each language and sentiment combination. This dataset was used to train the XLM-T sentiment classifier. The test splits of TSML were used to evaluate cross-lingual transfer

SpinningBytes 10k (SB10k): This dataset was designed for sentiment analysis on German Twitter and was a principal source of inspiration for this project (Cieliebak et al., 2017). Developed in the pre-transformer era, it was used to train convolutional neural networks and support vector machines that achieved contemporaneous state-of-the-art results. This dataset is compared with JTS1k in the cross-lingual transfer study.

Japanese Twitter Defamation Detection (JTDD): The JTDD trains models to identify defamatory language in tweets[5]. Initially, it included 5,000 examples, but now only 3,800 are available for download. Compared to sentiment analysis, defamation classification is a challenging task for annotators, reflected in a low *alpha* agreement score of 0.3. Despite these challenges, the dataset performs well across models.

Japanese Twitter Brand Reputation (JTBR): The dataset was designed to analyze the reputation of various products and brands using tweets from Twitter (Keshi, et al., 2017). Initially, it contained over 500,000 tweets primarily about mobile brands. Currently, only 80,000 remain available for download. Tweets are classified by the topic and sentiment. The sentiment labels represent the feelings of the user towards the topic. Therefore, this dataset aligns with aspect-based sentiment analysis.

Multilingual Amazon Reviews Corpus Japan (MARC-ja): MARC-ja is a simplified version of the Japanese segment of the Multilingual Amazon Reviews Corpus (MARC) (Keung et. al, 2020). This dataset has been adapted into a binary classification format. 1 and 2-star reviews are labelled as negative, 4 and 5-star reviews as positive, and neutral 3-star ratings are excluded. Reliability was enhanced using crowdworkers to validate the sentiment of each review, only including those with majority vote.

[5] JTDD is available at https://huggingface.co/datasets/kubota/defamation-japanese-twitter

Japanese Corpus of Linguistic Acceptability (JCoLA): Examples were extracted form linguistic journals, handbooks, and textbooks, and were judged for syntactic acceptability by experts. The published version contains helpful English translations for most examples. This task aligns with CoLA from the English GLUE (Warstadt et al., 2018). The original dataset used Matthew's correlation coefficient to measure performance, but this project used accuracy to compare with other JCoLA benchmarks.

Japanese Natural Language Inference (JNLI): JNLI is the Japanese adaptation of the Natural Language Inference (NLI) dataset, focusing on determining the inference relationship between a premise and a hypothesis sentence (Bowman et al., 2015). The possible relationships are entailment, contradiction, and neutral. The dataset employs sentences extracted from the Japanese MS COCO Caption Dataset and the YJ Captions Dataset, enabling the evaluation of models' abilities to discern subtle differences in meaning between closely related sentences.

Japanese Semantic Textual Similarity (JSTS): This is the Japanese adaptation of the Semantic Textual Similarity (STS) dataset, focusing on assessing the semantic similarity between pairs of sentences. The sentences in JSTS are derived from the Japanese MS COCO Caption Dataset and the YJ Captions Dataset (Miyazaki & Shimizu, 2016). Each sentence pair in the dataset is rated on a scale from 0 (completely different meanings) to 5 (equivalent meanings). The dataset aggregates scores from multiple annotators into a final continuous value. This classification task is regressive, meaning the model outputs a continuous value that is scaled during post-processing.

JCommonsenseQA (JCSQA): This dataset is the Japanese adaptation of the CommonsenseQA dataset (Talmor et al., 2019), which is designed to assess the commonsense reasoning capabilities of AI systems through multiple-choice questions. It is developed using crowdsourcing and utilizes seeds from the knowledge base ConceptNet to generate questions that require an understanding of everyday logic and common sense.

Japanese SQuAD (JSQuAD): This dataset is based on the Stanford Question Answering Dataset (SQuAD), utilizes the Japanese Wikipedia for reading comprehension exercises (Rajpurkar et al., 2016). Each entry presents a question related to a Wikipedia article snippet, requiring models to extract the answer.

Chapter 4: Building the Pre-Training Corpus

This chapter describes how Twitter data was acquired using the Twitter API. Different types of data were collected for various stages of development, but the majority was for building a corpus for unsupervised pre-training. Using the Twitter API to select a balanced and representative sample of language is not straightforward. This chapter explores the functionalities of the Twitter API, leading to the querying strategy that was used to build the training corpus. The corpus is evaluated by the diversity of time, place, and user data.

4.1: Limitations of the Twitter API

Twitter's Academic Research tier offered a monthly limit of 10 million tweets, modest when compared to the platform's daily output of 500 million posts. The corpus created by Barbieri et al. (2022) included 5% Japanese Tweets. If representative, this suggests that Japanese users generate approximately 25 million tweets daily. Beyond the Twitter Sample stream, which distributes 1% of all tweets, the API lacks additional sampling tools. API users can refine searches using parameters targeting specific keywords, accounts, times, among other objective features. To achieve a current and comprehensive view over time, a corpus covering at least a year was desirable. The Full Archive search, permitting access to tweets from any period, was the optimal choice. Two query parameters were used: a language filter selected the target data, and place filter extracted a manageable sample.

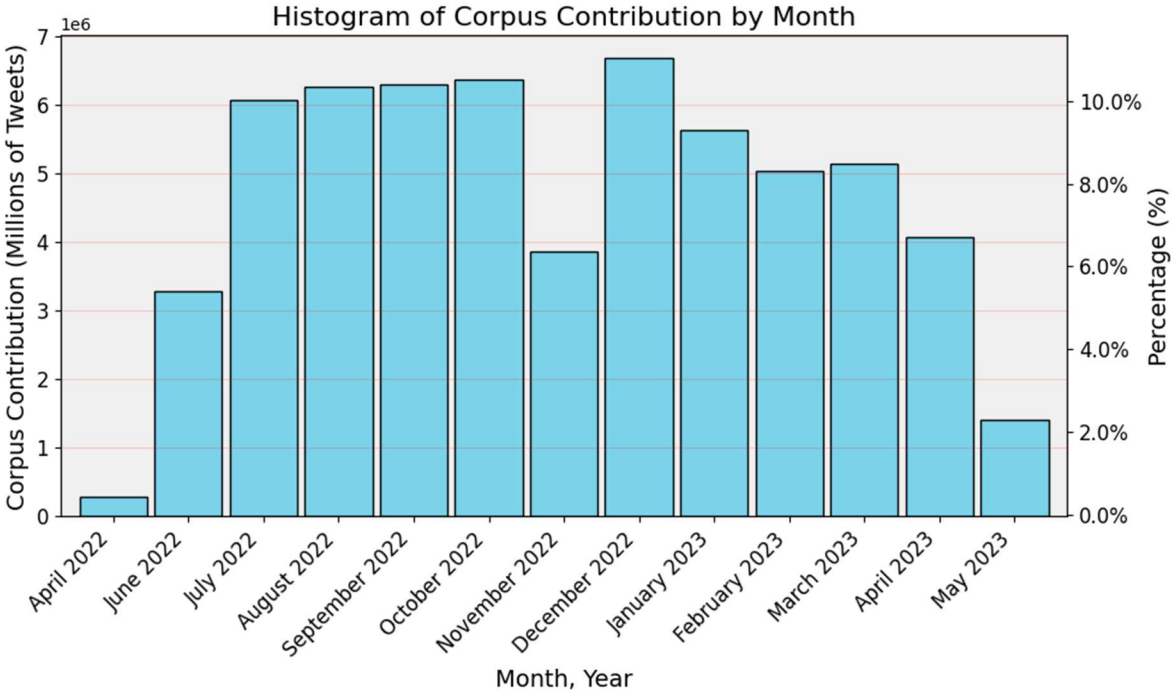
4.2: Filters for Targeting and Sampling Tweets

The *lang* operator filters tweets by the language set in user profiles, targeting those in a specified language. Some tweets retrieved with this filter, however, may contain little or no Japanese, highlighting the difficulty in accurately capturing language use. Despite its limitations, the *lang* operator is a straightforward and effective way to collect Japanese data. Ideally, the model trained on this data should handle multilingual expressions effectively, despite potential noise.

To distribute the quota over the desired timespan, a more specific operator was needed. The *place_country* operator, which searches for geotagged tweets matching a specified country code, proved useful. Twitter does not automatically geotag posts. Users must opt to share their location, and few do. Combining the *lang* and *place_country* operators, the query typically returned an average of 200,000 tweets over 24 hours. The archive was queried using these two operators on consecutive days until the quota was exhausted. This approach continued until access to the Academic Research tier was revoked in May 2023, by which time over 60 million tweets had been collected for the training corpus.

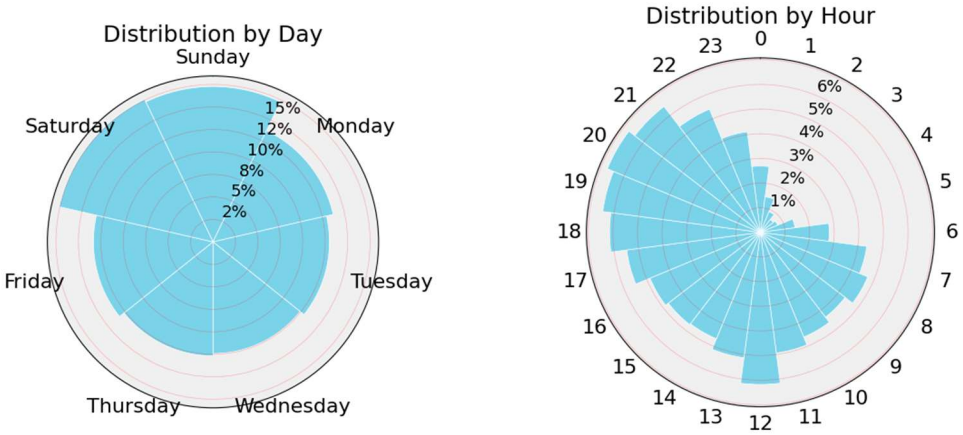
4.3: Corpus Analysis of Time and Place

Figure 4.1 Distribution of the Twitter Corpus by Month



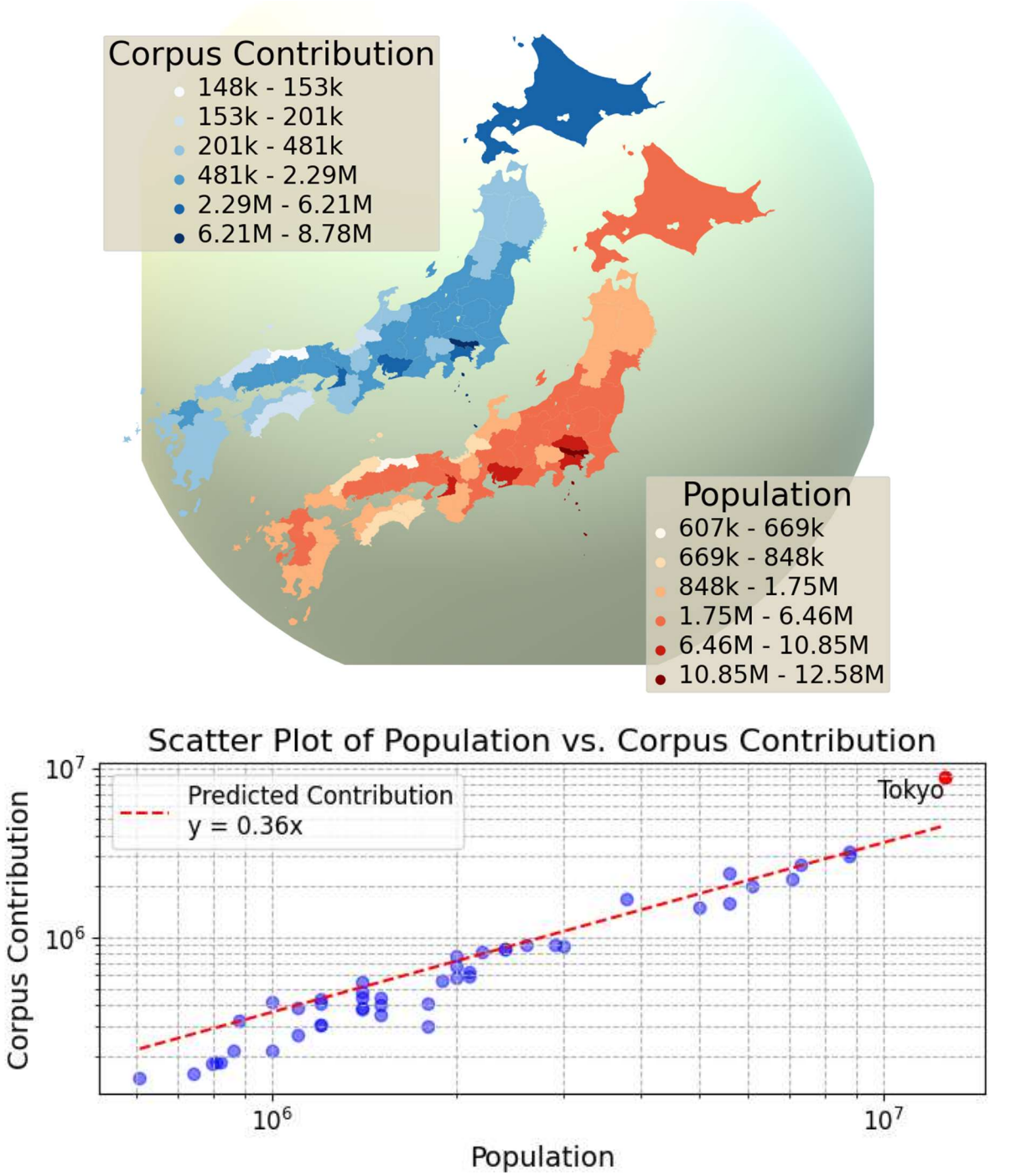
Sampling proceeded from June 2022 to May 2023 and included significant contributions from almost every day of this window. The low volume of tweets collected in November 2022 reflects a learning curve in using the API.

Figure 4.2 Distribution of the Twitter Corpus by Day and Hour



The hourly plot (left) shows consistent activity from 7:00-21:00 JCT that wanes and waxes across nighttime hours. The daily plot (right) shows activity over weekdays with a small boost on the weekend. These observations align well with the behavior expected from users living in Japan, validating the temporal representation of the corpus procured.

Figure 4.3 Distribution of the Twitter Corpus by Prefecture



The sampling method provided detailed geo data. The map (top) categorizes prefectures by percentile rank based on their population and corpus contribution. The scatter plot (bottom) illustrates the strength of the relationship between the two. The predicted contribution is given by dividing the size of the corpus by the population of Japan. The relationship is strong, with a Pearson correlation of 0.89. The largest population, Tokyo, is heavily represented. Higher population prefectures in general are better represented, which is further explored in Table 4.1.

Table 4.1 Distribution of the Twitter Corpus by Prefecture

	Population		Corpus Contribution		
Prefecture	Rank	Freq	Rank	Freq	Contribution Ratio
Tokyo	1	12.6M	1	8.8M	0.7
Osaka	2	8.8M	3	3.0M	0.34
Kanagawa	3	8.8M	2	3.2M	0.36
Aichi	4	7.3M	4	2.7M	0.38
Saitama	5	7.1M	6	2.2M	0.31
Chiba	6	6.1M	7	2.0M	0.33
Hokkai	7	5.6M	5	2.4M	0.43
Hyogo	8	5.6M	9	1.6M	0.29
Fukuoka	9	5.0M	10	1.5M	0.3
Shizuoka	10	3.8M	8	1.7M	0.45
Ibaraki	11	3.0M	13	880K	0.3
Hiroshima	12	2.9M	11	914K	0.32
Kyoto	13	2.6M	12	904K	0.34
Niigata	14	2.4M	14	859K	0.35
Miyagi	15	2.4M	15	857K	0.36
.....					
Okinawa	32	1.4M	23	548K	0.4
Yamagata	33	1.2M	38	303K	0.25
Oita	34	1.2M	37	306K	0.25
Ishikawa	35	1.2M	27	431K	0.37
Miyazaki	36	1.2M	30	409K	0.35
Akita	37	1.1M	40	267K	0.23
Toyama	38	1.1M	32	385K	0.35
Wakayama	39	1.0M	42	213K	0.21
Kagawa	40	1.0M	28	417K	0.41
Yamanashi	41	885K	36	320K	0.36
Saga	42	866K	41	215K	0.25
Fukui	43	822K	44	182K	0.22
Tokushima	44	810K	43	183K	0.23
Kochi	45	796K	45	179K	0.23
Shimane	46	742K	46	158K	0.21
Tottori	47	607K	47	148K	0.24

The table displays population and corpus contribution for the top and bottom fifteen prefectures, ranked by population. It calculates a contribution ratio based on these values. Prefectures with larger populations tend to have higher contribution ratios, as evidenced by a Spearman correlation of 0.55. This indicates that Twitter user activity is not only proportionally distributed across different regions but also somewhat concentrated in major urban areas.

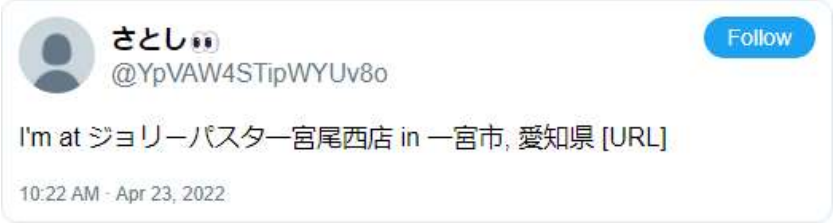
4.4: Shortcomings of the Sampling Procedure

The analysis of time and place data, shown in Figures 4.1 – 4.3 and Table 4.1, supports the representativeness of the Twitter corpus. The sampling method produced a geographically and temporally balanced dataset that covered nearly an entire year, suggesting a comprehensive and current language representation. However, further analysis revealed significant issues. The use of the place filter introduced two biased features into the dataset: auto-generated text and a notable lack of user diversity.

Auto-Generated Text

Slot filling, a method that uses templates filled with variable data, made up a significant portion of the corpus. This type of generated text is problematic for training because of its uniformity. Repetitive sequences inflate the corpus, which uses more hardware resources and slows down training processes (Lee et al., 2022), while also introducing biases through unnatural token associations. The app Swarm, which uses gamification to prompt user reviews, was a major source of this issue.

Figure 4.4 Example of Template Generated Text



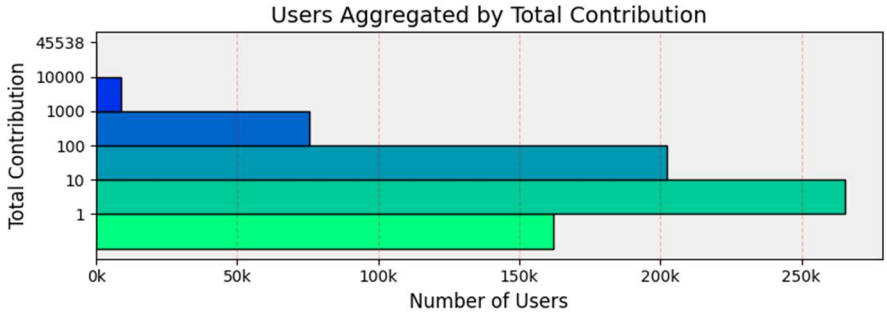
An example of a tweet generated by Swarm follows the template: "I'm at {BUSINESS} in {PLACE}".

The app encourages users to check in at locations, which are automatically geotagged. Therefore, the place filter skews sampling towards these entries. To counter this, regular expressions were used to identify and filter out these templated entries. Tweets from Swarm alone accounted for an alarming 10% of the corpus. The issue of template text and corpus refinement is addressed in Chapter 5.

Low User Diversity

For a balanced corpus, the ideal is to sample a few tweets from as many users as possible. The Twitter corpus averaged 85 tweets per user, which is far from ideal. Figures 4.4 and 4.5 visualize user balance by grouping users by the number of tweets that they contributed.

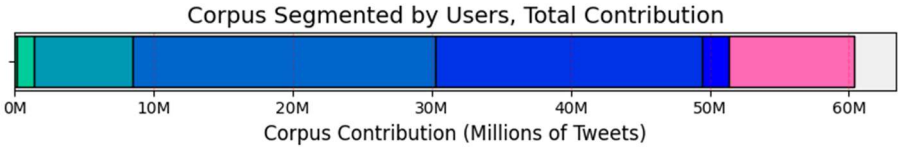
Figure 4.5 Distribution of Users by Contribution Volume



Range		Absolute		Relative	
Min - Max		Sum	Cum. Sum	Sum	Cum. Sum
1 - 1		162k	162k	22.70%	22.70%
2 - 10		265k	428k	37.15%	59.85%
11 - 100		202k	630k	28.33%	88.18%
101 - 1000		76k	706k	10.58%	98.76%
1001 - 9962		9k	714k	1.23%	99.98%
10085 - 45538		113	714k	0.02%	100.00%

The corpus includes tweets from approximately 714,000 unique users. Over half of these users contributed ten tweets or less, and almost 90% contributed fewer than 100 tweets. A few outlier users posted more than 10,000 tweets, with the highest number of tweets from a single user being 45,538.

Figure 4.6 Corpus Segmented by Users of Varying Contribution Volume



Range		Absolute		Relative	
Min - Max		Sum	Cum. Sum	Sum	Cum. Sum
1 - 1		0.2M	0.2M	0.27%	0.27%
2 - 10		1.2M	1.3M	1.97%	2.23%
11 - 100		7.2M	8.5M	11.86%	14.10%
101 - 1000		21.7M	30.2M	36.01%	50.11%
1001 - 9962		19.2M	49.5M	31.87%	81.97%
10085 - 45538		1.9M	51.4M	3.10%	85.08%
Unknown		9.0M	60.4M	14.92%	100.00%

The corpus is predominantly made up of contributions from the heaviest users. Those who contributed ten tweets or less accounted for only 2.2% of the corpus. The 10% of users who contributed between 100 and 1,000 tweets represented over a third of the total contributions. Their input is almost matched by the 1.2% of users who contributed between 1,000 and 10,000 tweets. This skewed distribution indicates a significant lack of diversity in user contributions, affecting the representativeness and potential bias of the dataset. 'Unknown' tweets were collected with the Twitter Download, which does not provide user data.

4.5: Interim Conclusion

This procedure evaluated whether a query formulation using time, place, and language filters could create a balanced and representative corpus. The sampling achieved a balanced, current, and comprehensive timespan. Place filtering allowed for detailed geographic data analysis, supporting the geographic representation of the Twitter corpus. However, user representation fell short of expectations.

Table 4.2 Comparison of User Diversity Between Corpora

Corpus	Total Tweets	Unique Users	Tweets per User
BERT for Japnese Twitter	60,365,838	714,479	84.49
hottoSNS-bert	85,925,384	1,872,623	45.89
Edinburgh Twitter Corpus	96,369,326	9,140,015	10.54

Both the Edinburgh Twitter Corpus (ETC) and the training corpus for hottoSNS-bert were sampled using the Twitter Stream (Petrović et al., 2010; Sakaki et al., 2019). The ETC did not use language filters, therefore accessing the largest user base. The training corpus for BERT for Japnese Twitter is less diverse than the other two. This suggests that the place filter, which excludes non geotagged tweets, is biased towards relatively few users that use geotagging habitually.

The place filter provided valuable insights for corpus analysis, though it compromised balance. Filtering by geotagging is appropriate when the specific place matters, such as in event tracking and natural disaster monitoring. For a general language sample, using a keyword-based approach might have produced a higher quality corpus. A recommended strategy to obtain a semi-random sample is to use stop words. Keyword based sampling could help address auto-generated text. By analysing common templates, distinct terms frequently found in these templates can be identified. These terms can then be used to create a list of negative keywords aimed at filtering out bot-generated tweets, thereby improving the representation of human-generated content in the corpus. Managing bloated web corpora is a well-known challenge in NLP (Lee, et al., 2022). Strategies like deduplication, which will be discussed in Chapter 5, help address this issue.

Chapter 5: Refining the Pre-Training Corpus

The evaluation of the 60 million tweet corpus from Chapter 4 revealed two major issues: repetitive text and user imbalance. This chapter outlines the steps taken to refine the raw corpus into one that is leaner and more linguistically diverse. It begins by discussing how duplicate text undermines training. Next, it describes two methods that were employed to refine the corpus. First, an algorithm involving n-grams was used to remove near-duplicates, a process that was both time-consuming and computationally intensive. This section defines the algorithm, details parameter fine-tuning, and outlines the deduplication procedure. The impact of deduplication is discussed, illustrated by text samples from varyingly affected users. Further refinement is achieved by balancing authors' contributions with a simple heuristic. The raw and refined corpora are evaluated by size, user balance, and frequency distribution of n-grams, hashtags, and mentions.

5.1: Pitfalls of Duplicate Text

On of the challenges of working with web corpora is the ubiquity of duplicate text. Duplicates are problematic for training. First, they can cause data leakage, where the same examples are used in both the training and the test set. Excessive data leakage produces models that memorize rather than generalize (Elangovan et al., 2021). Such models tend to perform well on their test set but struggle outside of the training context. The other consideration is efficiency. Repeated observation of similar token sequences is wasted training time. Deduplicating corpora speeds up convergence by reducing the training load while preserving language variety. If removing duplicates resolves some data quality issues, there is potential to improve performance (Lee et al., 2022).

5.2: Addressing Template Generated Text

Chapter 4 raises concerns about the 5 million tweets that were generated by *Swarm*. These tweets use an easily identifiable template: "I'm at {PLACE} in {PLACE}". Template-generated text creates strong, unnatural associations between the tokens used in the template, necessitating their removal. The simplest deduplication method is to remove identical examples, but this is insufficient for addressing template-generated text. For example, with exact duplicates removed, 1.6 million of the 5 million Swarm generated tweets remain. Once the template has been identified, regular expressions prove highly effective at targeting such text. However, the template problem extends beyond Swarm. Identification of all templates scattered throughout the corpus is beyond the resources of this project. Therefore, a more sophisticated, data-driven approach was required.

5.3: Efficient Near-Duplicate Deduplication Using Min-Hashes

Near-deduplication involves reducing texts that exceed a certain similarity threshold. A common approach segments text into n-gram shingles to compute the Jaccard similarity coefficient, as demonstrated in Appendix 1. This method is highly effective for identifying template generated texts but poses scalability challenges. For example, the 60 million tweet corpus entails over 100 trillion comparisons. To address resource constraints, shingled texts are further processed into Min-Hashes. Min-Hashes are preferred because they quickly and accurately estimate similarity. Furthermore, they cluster well, which substantially reduces the complexity of the problem (Broder, 1997; Lee et al., 2022). The specially designed Python library, *NLPDedup*[6], was used to carry out the algorithm. The deduplication procedures were performed on a server with an 8-core CPU and 62.6Gb of RAM.

5.4: Fine-Tuning the Deduplication Parameters

The deduplication algorithm is controlled by several parameters that impact its sensitivity, accuracy, and complexity. Two parameters that control sensitivity—Jaccard similarity and n-gram size—were chosen for fine-tuning to optimize the process: **Jaccard similarity** is a ratio that indicates the percentage of shared n-gram sequences required for two tweets to be considered nearly identical. Higher thresholds result in fewer duplicates being recognized, thus reducing sensitivity. **N-gram size** represents the number of tokens in each n-gram shingle. Larger n-grams decrease the likelihood of shared sequences, further reducing sensitivity. To determine the optimal configuration for deduplication, two sample corpora were prepared:

- **Target Corpus**: Consists of 5 million tweets generated by Swarm. This set acts as a benchmark for measuring recall. The Swarm tweets are simple. To be robust against more complex templates, the configuration must be aggressive enough to eliminate the target.
- **Control Corpus**: Comprises 5 million tweets from the raw corpus, excluding Swarm tweets. This set helps estimate precision. Although the known templates are removed, a significant portion is still duplicate. The ideal trial will flag close to 100% of the target while showing restraint with the control.

Deduplication was applied to both corpora under various configurations, with the results displayed in Table 5.1 and Figure 5.1. This approach aims to fine-tune the process, ensuring high

[6] Python Library *NLPDedup:* https://saattrupdan.github.io/NLPDedup/nlp_dedup.html

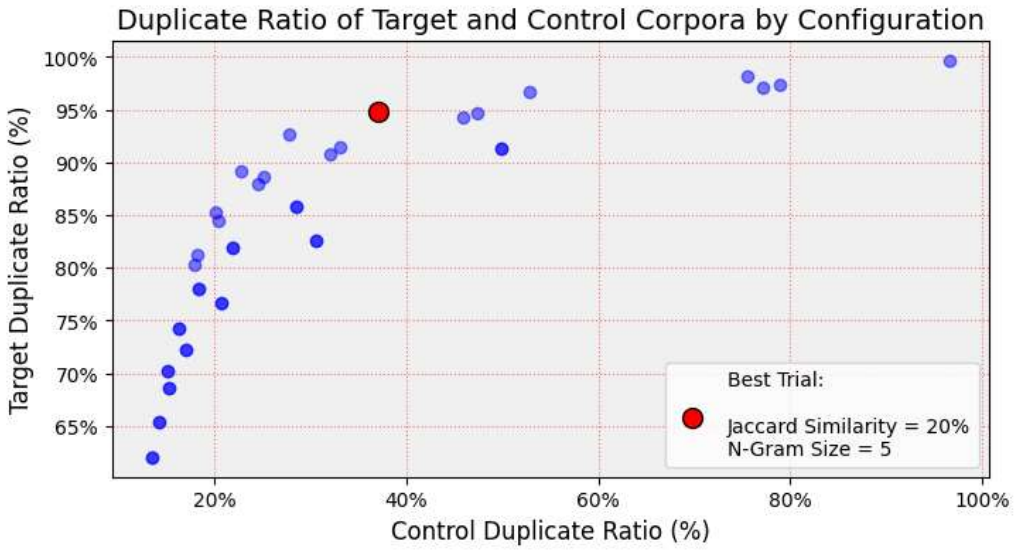
efficiency in recognizing and removing duplicates without excessively impacting the integrity of the control corpus.

Table 5.1 Results from Deduplicator Tuning Procedure

Jaccard Similarity	N-Gram Size 2	3	4	5	6	7	Legend
50%	82.6%	76.7%	72.3%	68.6%	65.4%	62.1%	Target
	30.6%	20.7%	17.1%	15.3%	14.3%	13.5%	Control
45%	82.6%	76.7%	72.3%	68.6%	65.4%	62.1%	
	30.6%	20.7%	17.1%	15.3%	14.3%	13.5%	
40%	91.4%	85.8%	81.9%	78.0%	74.3%	70.2%	
	50.0%	28.5%	21.9%	18.4%	16.4%	15.1%	
35%	91.4%	85.8%	81.9%	78.0%	74.3%	70.2%	
	50.0%	28.5%	21.9%	18.4%	16.4%	15.1%	
30%	97.1%	94.3%	90.7%	87.9%	84.5%	80.3%	
	77.2%	45.9%	32.1%	24.6%	20.5%	17.9%	
25%	97.4%	94.6%	91.5%	88.7%	85.3%	81.2%	
	78.9%	47.3%	33.1%	25.2%	20.2%	18.2%	
20%	99.7%	98.3%	96.7%	**94.9%**	92.7%	89.2%	
	96.5%	75.5%	52.8%	**37.1%**	27.9%	22.8%	

The percentages reflect the proportion of the corpora that were flagged as duplicate. The optimal configuration, with an n-gram size of 5 and 20% Jaccard similarity threshold, flagged nearly 100% of instances in the target corpus while showing more restraint with the control group.

Figure 5.1 Scatterplot of the Deduplicator Tuning Procedure



This configuration struck a favorable balance between thorough deduplication of the target corpus and minimal impact on the control group. It performed better than the next three configurations, which resulted in a higher loss of control data.

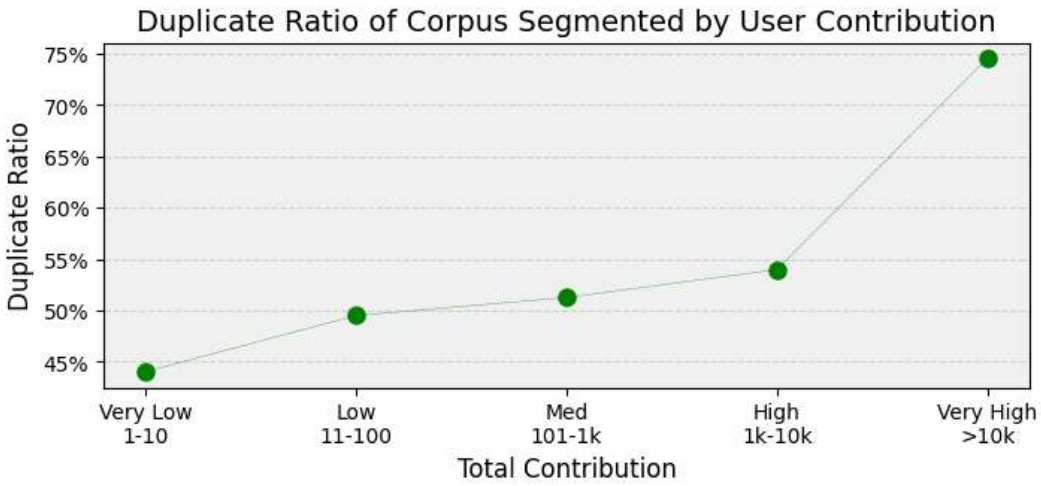
5.5: Execution of the Deduplication Procedure

While Min-Hashes significantly reduce overhead, the procedure remained extremely expensive, particularly the memory cost. Deduplicating the entire corpus at once would require hundreds of gigabytes of RAM. The fit within hardware constraints, the corpus was randomly batched into eight sub-corpora. To provide the opportunity for a comprehensive pair-wise comparison, that procedure was repeated equal times to the number of batches. Therefore, batching increases the time complexity linearly. Sixty-four iterations of batch deduplication, each taking about two hours, amounted to approximately 128 hours of dedicated computing time. The procedure concluded with 32 million tweets identified as duplicate, representing 53% of the corpus

5.6: Impact of Deduplication on User Balance

It was assumed that high-activity users that contributed hundreds or thousands of tweets to the Twitter corpus would be more likely to produce duplicate text. If this were true, then deduplication should have a natural balancing effect on user contributions. Following deduplication, the relationship between the size of user contributions and the likelihood of duplicate text was analyzed. This involved segmenting the corpus by user contributions and calculating the duplicate ratio. The corpus was aggregated as shown in Figure 4.5.

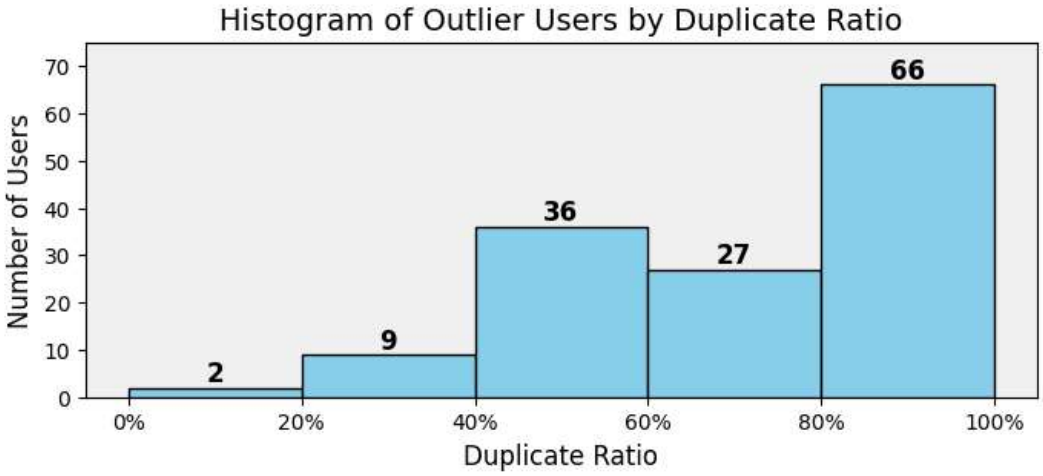
Figure 5.2 Duplicate Ratios of Corpus Segmented by User Contribution Level



Within the normal range, there was no relationship between contribution size and duplicate ratio. Deduplication reduced the corpus volume by approximately half across contribution levels. However, the 3% of the corpus contributed by 'Very High' activity users was reduced by

75%. Therefore, those users with outlying zone of 'Very High' activity do seem to be more likely to produce repetitive text. These users are examined more closely in Figure 5.3.

Figure 5.3 Histogram of 'Very High' Contributors Grouped by Duplicate Ratio



Most of the 'Very High' activity users produced more than 80% duplicate tweets. Therefore, the deduplication procedure did have a limited balancing effect on user contributions. However, a few of these outlier uses returned duplicate ratios that were well below average.

Deduplication partly balanced the corpus by reducing the influence of outlier users, yet it remains biased towards heavy contributors. Tables 5.2 to 5.4 analyze text samples from three 'Very High' contributors, illustrating varying impacts of deduplication and supporting its effectiveness in targeting redundant texts.

Table 5.2 Text Sample from User with 99% Duplicate Ratio

Japanese	English
[USER] おとぎの国姫さん、こんにちは😊💮✌✨ 本日も宜しくお願いします。💠🎬	[USER] Hello fairy-tale princess 😊💮✌✨ Thank you for your continued support today. 💠🎬
[USER] いそ丸水産さん、こんにちは😊💮✌✨本日も宜しくお願いします。🍳🍙🍺	[USER] Hello, Isomaru Suisan 😊💮✌✨ Thank you for your continued support today. 🍳🍙🍺
[USER] Hiroさん、おはようございます😊💮✌✨ 本日も宜しくお願いします。🍳	[USER] Hiro-san, good morning 😊💮✌✨ Thank you for your continued support today. 🍳
[USER] うーちゃん、おはようございます😊💮✌✨本日も宜しくお願いします。🌸🍳	[USER] Good morning, U-chan 😊💮✌✨ Thank you for your continued support today. 🌸🍳
[USER] ぼんゆうさん、おはようございます😊💮✌✨今週も宜しくお願いします。🍳🌸	[USER] Good morning, Bonyu-san 😊💮✌✨ Thank you for your continued support this week. 🍳🌸

This user generated 31k tweets at a rate of 112 tweets per day, showing minimal language variation. The content predominantly consists of template text using a slot fill mechanism to incorporate names.

Table 5.3 Text Sample from User with 57% Duplicate Ratio

Japanese	English
日本維新の会が「暴力団(統一教会)のイベントに参加(爆笑)」してた議員に対してどんな処分するのか？注目でしょう❤(笑)	What kind of punishment will the Japan Restoration Party take against a lawmaker who "participated in an organized crime group (Unification Church) event (lol)"? It will be worth attention 💙 (lol)
あんまり言うと怒られる(大爆笑)から程々にしとくけどさ❤(笑)	If I say it too much, I'll get angry (lol), so I'll keep it in moderation 💙 (lol)
しかし、誰がマインツに付いたのか？❤(笑)	But who joined Mainz? 💙 (lol)
教えた細かい技術は今後の課題、コーチ共々、やっとけよ❤(笑)	The detailed techniques I taught will be my future work, so let's do it together with my coach 💙 (lol)
そんなに際どくねぇ～し❤(笑)	It's not that risque 💙 (lol)

As the corpus's largest contributor, this user posted 45k tweets at a rate of 173 tweets per day, usually ending posts with " ❤(笑)". They employ unconventional posting styles, such as 'daisy-chaining' to bypass character limits. Chains are disrupted during preprocessing. Within a model's capabilities, these tweets are difficult to interpret because they lack context. Nevertheless, this user makes a valuable contribution by discussing a variety of topics.

Table 5.4 Text Sample from User with 17% Duplicate Ratio

Japanese	English
ディアボロとキンクリの目が同じ目してるのすこすこ	Diavolo and Kinkuri have the same eyes.
え?これゴローニャはやっぱ通信交換なの?	Huh? Is this Gologna a communications exchange after all?
おやすみぃぃ(スライディング就寝)	Good night (sliding bedtime)
[USER] るゆむつぬんんるゆむゆる？んゆむつつつぬぬむゆ！へぬゆ	[USER] Ruyumutsununnruyumyuru? Nyumutsutsutsununumuyu! Henuyu
ドククラゲで80本ならウツロイドもそんくらいありそう	If there are 80 doku jellyfish, there are likely to be about that many Uturoids as well.

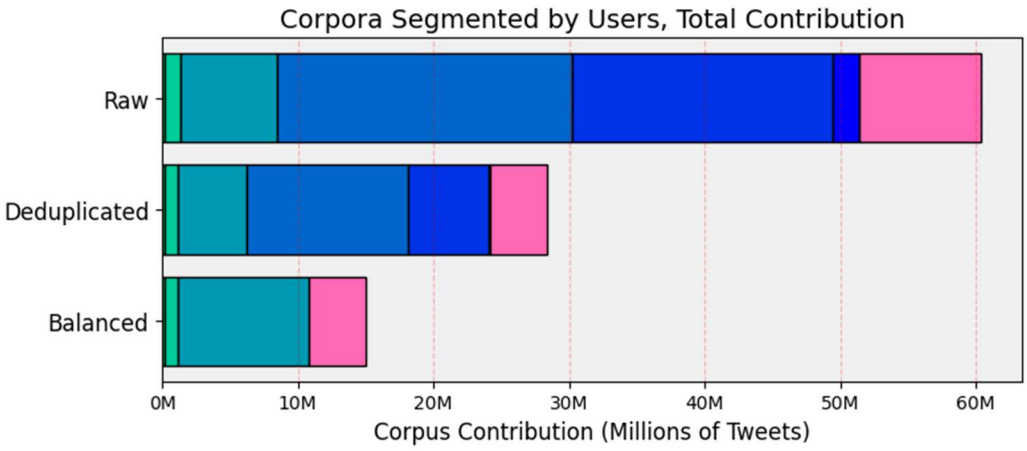
This user, posting 12k tweets at 45 tweets per day, shows no signs of automated text generation. Their contributions are linguistically varied, enhancing the corpus diversity.

5.7: Corpus Refinement by Capping User Contributions

The review of user samples indicated that deduplication achieved its intended purpose, but a comprehensive examination was not feasible. This leaves some uncertainty about potential biases introduced by user imbalance. To advance the refinement process, a simple heuristic was

implemented: contributions from individual users were capped at one hundred tweets. The impact of this measure on user contributions is depicted in Figure 5.4.

Figure 5.4 Balance by User Contribution of Raw and Refined Corpora



Total Contribution	Raw		Deduplicated		Balanced	
1	**162k**	0.27%	**158k**	0.56%	**158k**	1.06%
2 - 10	**1.2M**	1.97%	**1.0M**	3.69%	**1.0M**	6.97%
11 - 100	**7.2M**	11.86%	**5.0M**	17.81%	**9.6M**	64.22%
101 - 1k	**21.7M**	36.01%	**11.8M**	41.77%	---	---
1k - 10k	**19.2M**	31.87%	**6.0M**	21.02%	---	---
10k+	**1.9M**	3.10%	**106k**	0.37%	---	---
Unknown	**9.0M**	14.92%	**4.2M**	14.79%	**4.2M**	27.75%

This analysis illustrates the change in user balance over stages of refinement. The relative contributions from users of ten or less tweets significantly increase. From users of more than 1,000, contribution decreases. Almost two thirds of the balanced corpus was from the highest contributors. Therefore, the balanced corpus remains skewed, but it has better representation of lower level contributors.

Deduplication cut the corpus size from 60 million to 27 million tweets. User balancing reduced it to 15 million tweets, much smaller than sizes reported in other studies (Barbieri et al., 2022; Cieliebak et al., 2017; Nguyen et al., 2020; Sakaki et al., 2019). Despite its smaller size, the refined corpus allows for more efficient training and is expected to provide a more diverse representation of language. This setup should help ensure that the model can learn effectively from a broad spectrum of content without being dominated by prolific users. To further investigate the impact of refinement on diversity, the corpora were compared by n-grams, mentions, and hashtags.

5.8: Content Analysis of Raw and Refined Corpora

Comparison of the Raw and Deduplicated corpora indicates that deduplication maintained a high level of linguistic diversity. Despite being less than half the size, the Deduplicated corpus retains approximately 75% of the unique n-grams from the Raw corpus. Conversely, the unique n-grams drop significantly from the Deduplicated to the Balanced corpus.

Table 5.5 Analysis of N-Grams across Raw and Refined Corpora

	Raw		Deduplicated		Balanced	
	Total	Unique	Total	Unique	Total	Unique
Unigrams	1.5B	2.7M	808.5M	2.0M	447.6M	1.5M
Bigrams	1.5B	56.7M	808.5M	44.4M	447.6M	29.7M
Trigrams	1.5B	270.1M	808.5M	204.9M	447.6M	125.4M
4-grams	1.5B	558.4M	808.5M	408.0M	447.6M	235.8M

MeCab was used to tokenize the corpora (Kudo et al., 2004). Analysis of n-grams was facilitated by the Python library, NLTK[7].

The ratio of unique to total n-grams is the highest in the Balanced corpus. However, this is not a fair comparison, as the ratio of unique to total n-grams naturally decreases as corpora increase in size. To estimate content diversity as a function of text quality rather than corpus size, a million tweets were sampled from each corpus. Figures 5.5 and 5.6 analyze the diversity of n-grams, hashtags, and mentioned users. These analyses calculate diversity using the Shannon Diversity Index. If the refinement processes enhanced the diversity, then these three elements should increase across stages of refinement.

Equation 5.1: Shannon Diversity Index

$$p_i = \frac{n_i}{N} \qquad Diversity = -\sum_{i=1}^{N}\left(p_i * ln(p_i)\right)$$

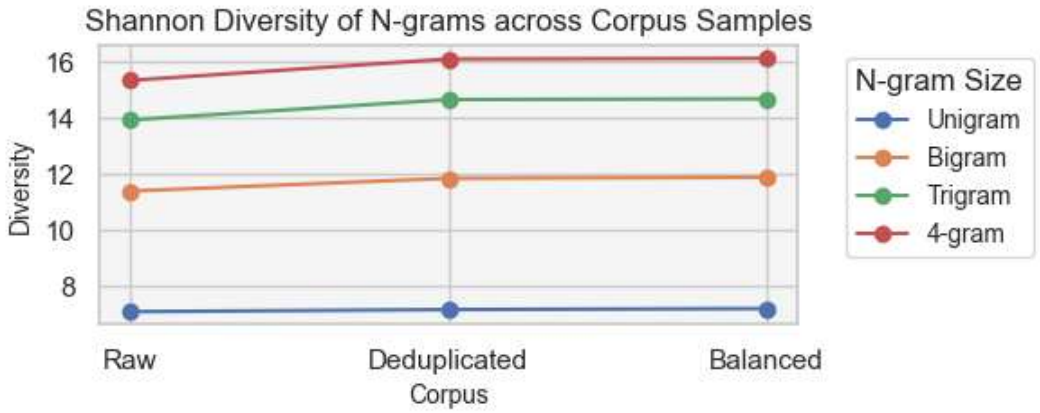
The proportion of item i, p_i, is given by the number of i, n_i, divided by the total number of items, N. Diversity is given by the negative summation of each item proportion multiplied by its natural log.

Based on the analysis shown in Figures 5.5 and 5.6, the assumption that linguistic diversity would with refinement was partially supported. The transition from the Raw to the Deduplicated corpus showed a slight increase in n-gram diversity, which supports that the examples that remain are less repetitive. Further refinement through balancing did not enhance n-gram diversity. However, it did improve hashtag diversity, suggesting more rounded topical coverage within the Balanced

[7] https://www.nltk.org/

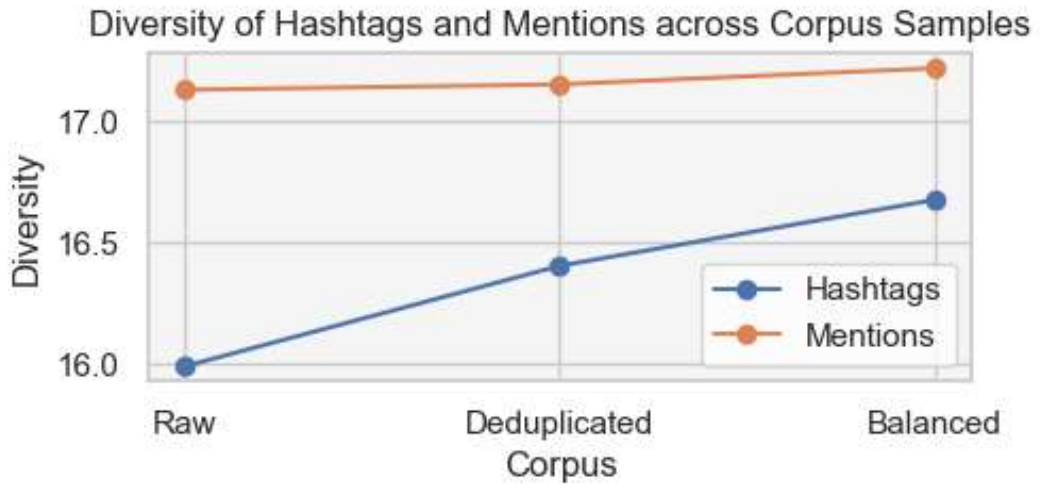
corpus. These findings are promising. However, the difference in diversity measurements is small, and the magnitude is not well understood. This study would be better informed by a more comprehensive analysis. It should take multiple samples to confirm the reliability and significance of measurements. Comparing these findings with another Twitter corpus would help frame the conclusions within the bigger picture of Twitter space.

Figure 5.5 Diversity of N-Grams across Raw and Refined Corpora



This analysis shows that diversity of n-grams increases slightly from the Raw to the Deduplicated corpus. However, there is no change between the Deduplicated and Balanced corpora.

Figure 5.6 Diversity of Hashtags and Mentioned Users across Raw and Refined Corpora



The mentioned users remain consistent, but hashtag diversity increases at every level of refinement. Hashtags are related to the topical content of corpora (Petrović et al., 2010). The increase in hashtag diversity may be a positive signal for the balance of content within the Balanced corpus.

The final phase of content analysis focuses on the most common n-grams, hashtags, and mentioned users. The objective is to identify problematic terms from the Raw corpus. This includes n-grams typically found in template-generated texts, promotional or spam-related hashtags, and users often mentioned in automated posts. If corpus refinement was successful, then these elements should be significantly reduced in the refined corpora.

Table 5.6 Top Frequency 4-Grams across Raw and Refined Corpora

Rank	Raw	Deduplicated	Balanced
1	I ' m at	！ ！ ！ ！	！ ！ ！ ！
2	お 願い し ます please	お 願い し ます please	お 願い し ます please
3	区 , 東京 都 Tokyo Metropolitan Ward	たの で 、 because	ありがとう ございました thank you very much
4	投稿 しました posted	ありがとう ございました thank you very much	て きました have come
5	を 投稿 しまし posted	て きました have come	たの で 、 because
6	写真 を 投稿 し posted a photo	て います 。 Is doing	て います 。 Is doing
7	に 写真 を 投稿 posted a photo to	なの で 、 therefore	しました 。 did
8	よろしく お 願い し thank you in advance	しました 。 did	して おり ます doing
9	ありがとう ございました thank you very much	に なり ました became	なの で 、 therefore
10	お めでとう ござい ます congratulations	お 疲れ 様 でし good work	して い ます is doing
11	お 疲れ 様 でし good work	疲れ 様 でした thank you for your hard work	に なり ました became
12	疲れ 様 でした thank you for your hard work	して い ます is doing	お 待ち して waiting
13	して おり ます doing	と 思い ます 。 I think	よろしく お 願い し thank you in advance
14	, 東京 都) Tokyo Metropolis	して おり ます doing	ましたが 、 however
15	お 待ち して waiting	ましたが 、 however	お 疲れ 様 でし good work

Because of the prevalence of Swarm tweets, 'I'm at' is the top ranking 4-grams. The problematic entries highlighted in red dropout and are appropriately replaced with phatic language, such as "お願い し ます", and grammatical phrases, such as "し ました 。".

Table 5.7 Top Frequency Hashtags across Raw and Refined Corpora

Rank	Raw	Deduplicated	Balanced
1	#Yahooニュース #Yahoo News	#ラーメン #ramen	#ラーメン #ramen
2	#アメブロ #Ameblo	#Yahooニュース #Yahoo News	#イマソラ #Sky Now
3	#イマソラ #Sky Now	#イマソラ #Sky Now	#Yahooニュース #Yahoo News
4	#ラーメン #ramen	#ランチ #lunch	#ランチ #lunch
5	#TikTok #TikTok	#阪神タイガース #Hanshin Tigers	#アメブロ #Ameblo
6	#shindanmaker #shindanmaker	#shindanmaker #shindanmaker	#京都 #Kyoto
7	#静岡県 #Shizuoka Prefecture	#アメブロ #Ameblo	#沖縄 #Okinawa
8	#静岡県東部 #Eastern Shizuoka Prefecture	#京都 #Kyoto	#北海 #Hokkaido
9	#ランチ #lunch	#沖縄 #Okinawa	#サウナ #sauna
10	#17LIVE #17LIVE	#北海 #Hokkaido	#大阪 #Osaka
11	#企業公式が毎朝地元の天気を言い合う #morning weather report	#TikTok #TikTok	#TikTok #TikTok
12	#横浜 #Yokohama	#舞いあがれ #Maiagare!	#shindanmaker #shindanmaker
13	#阪神タイガース #Hanshin Tigers	#chibalotte #chibalotte	#note #note
14	#ライブ配信中 #live streaming now	#スマートニュース #smartnews	#カレー #curry
15	#東京 #Tokyo	#WBC #WBC	#日本酒 #sake

The top frequency hashtags do not change as dramatically as the other categories. "#17LIVE" is a streaming website that advertises on Twitter aggressively. The other two hashtags are engagement seeking. The most frequent type of tag are place names, which likely resulted from the sampling method.

Table 5.8 Top Frequency Mentioned Users across Raw and Refined Corpora

Rank	Raw	Deduplicated	Balanced
1	@YouTubeより	@YouTubeより	@YouTubeより
2	@jreast_official	@YouTube	@YouTube
3	@711sej	@sharenewsjapan1	@sharenewsjapan1
4	@YouTube	@T_IPPONGP	@bozu_108
5	@famima_now	@bozu_108	@T_IPPONGP
6	@akiko_lawson	@tweetsoku1	@souhakurumi
7	@sharenewsjapan1	@oogiri_zamurai	@truckmeimei
8	@haneda_official	@souhakurumi	@tweetsoku1
9	@aeon_japan	@Sankei_news	@Sankei_news
10	@T_IPPONGP	@truckmeimei	@oogiri_zamurai
11	@driveplaza	@nonbeiyasu	@YahooNewsTopics
12	@c_nexco_sapa	@tsuisoku777	@syouwa_otome
13	@mcdonaldsjapan	@YahooNewsTopics	@tsuisoku777
14	@keikyu_official	@syouwa_otome	@kishida230
15	@bozu_108	@pyonkichitweets	@livedoornews

Japanese transportation entities, such as JR East, utilize Twitter for public outreach. Three of these entities rank among the most mentioned users, and their tweets tend to be highly repetitive. The other problematic accounts are tied to businesses. After refinement, the most frequently mentioned users shifted to include news sources like "@YahooNewsTopics," politicians such as "@Kishida230," and lesser-known individuals like "@souhakurumi" who are primarily recognized through their Twitter presence.

The final analysis of content showed that the refinement process impacted the representation of the corpus. Repetitive n-grams, spammy hashtags, and corporate users are signals of bloated content. Following deduplication, these features dropped in salience. It was expected that the Balanced and Deduplicated corpora would yield different results, and their similarity raises doubts about the effectiveness of the balancing step. While there is hope that the Balanced corpus will enhance training efficiency, the analysis indicates that balancing users did not improve the diversity of the corpus.

5.8: Interim Conclusion

In retrospect, features like n-grams, mentions, and hashtags could have been more effectively utilized. N-grams, for example, are useful for identifying pervasive templates which can then be extracted using regular expressions. Hashtags and mentions, on the other hand, could help pinpoint spam accounts. Leveraging these features could have resulted in a more diverse target corpus and a cleaner control group. Moreover, a more thorough screening would have eased the workload of deduplication and helped manage resource constraints. Despite these areas for improvement, the overall analysis confirms that deduplication effectively achieved its intended effect.

This exploration was aimed at testing the hypothesis that refining the training corpus can improve its quality. Indeed, deduplication enhanced the corpus's diversity across several metrics. However, the impact of capping user contributions on the quality of the corpus remains uncertain. The Balanced corpus is only half the size of the Deduplicated corpus. While larger corpora traditionally yield better results, the field of domain adaptation suggests a trade-off between general competence and domain-specific accuracy (Gururangan et al., 2020). It is possible that a smaller training corpus could provide an optimal balance between general and Twitter-specific domains. Additionally, the reduced size means that training epochs complete more quickly, allowing for more frequent iterations and opportunities for optimization.

The project involved experimenting with both corpora, leading to slightly different vocabularies and tokenizer configurations, as discussed in Chapter 7. The choice of training corpus was a key variable hyperparameter, further explored in Chapter 8, highlighting its impact on model performance and adaptability.

Chapter 6: Building the Sentiment Dataset

This chapter describes the process of building the Japanese Twitter Sentiment 1k (JTS1k) training set. It begins by defining key measures of dataset quality: size, balance, representativeness, and reliability. Preparing for the annotation phase was a team effort involving two native Japanese speakers. A balanced and representative dataset was selected from the Twitter corpus for annotation. The annotation procedure was rehearsed, and clear task instructions were developed. As annotations proceeded, reliability was validated by measuring annotator agreement. Once annotation was complete, JTS1k was used to benchmark a series of generative AI models, demonstrating its compatibility with a variety of models.

6.1: Considerations for Size

Effective model training requires a diverse and representative language sample. Traditionally, representation has been achieved through large volumes of data, as larger datasets tend to yield better results. However, any dataset will fail if annotation quality is poor. Building high-quality datasets is costly and requires a budget that supports validating reliability proportional to the dataset size. Cieliebak et al. (2017) optimized size and reliability in constructing a sentiment analysis dataset of 10,000 German Twitter examples (SB10k). With stricter annotator agreement thresholds, it outperformed a less reliable dataset ten times its size. In the pre-transformer era, they posited that 10,000 examples were the minimum for high-quality outcomes. Contemporary encoder models achieve state-of-the-art results with significantly smaller datasets (Devlin et al., 2019). Advanced generative AI models also perform well in some tasks with minimal examples (Brown et al., 2020). Given these advancements and budget constraints, a dataset of a thousand examples was deemed sufficient, expected to perform adequately on its own and be useful in future, larger-scale projects.

6.2: Considerations for Balance and Representativeness

The optimal composition of a dataset depends on the task requirements. For detecting sentiments in general contexts where clear opinions are rare, a higher proportion of neutral examples may be preferable. However, the target sentiment analysis model is intended for an opinion-rich data stream, often containing subtle expressions of sentiment. Therefore, an evenly balanced dataset is more appropriate.

The methodology required an equal and representative sample from each of the four sentiment categories: positive, negative, neutral, and mixed. Given the smaller size of the dataset, achieving representativeness required deliberate sampling techniques. For instance, Cielback et al.

(2017) enhanced token representation by applying k-means clustering to bag-of-words representations of their corpus. For clustering the Japanese Twitter corpus, a more contemporary approach was used. K-means clustering was applied to sentence embeddings encoded by a prototype BERT for Japanese Twitter[8]. While the bag-of-words methodology promotes unigram diversity, this method is more semantically driven. Both strategies aim to capture the full diversity of language usage, ensuring that the datasets are representative and comprehensive in their linguistic features.

Following the methodology of Cielback et al. (2017), the Twitter corpus was divided into 250 clusters, with the intention of sampling one example from each sentiment class per cluster. Their publication emphasizes efforts to maximize the inclusion of mixed sentiment candidates. Despite these efforts, their dataset predominantly consisted of neutral tweets, making up over half of the total, while mixed sentiment tweets constituted less than 5%.

6.3: Considerations for Reliability

Reliability in a dataset indicates the probability that annotations accurately reflect true values. Typically, reliability is assessed by labelling examples multiple times and evaluating the consistency among labels. The choice of metric for measuring this agreement varies, depending on how the workload is distributed among annotators. For annotating JTS1k, subsets of data were assigned unevenly among numerous crowdworkers. Therefore, Krippendorff's alpha was the most appropriate measure. Alpha values range from 0 to 1, where 1 represents perfect agreement. Generally, agreement scores are considered valid above a certain threshold, which varies by task (Hayes & Krippendorff, 2007). In a review of multilingual Twitter sentiment datasets, Mozetič et al. (2016) found that the most reliable datasets achieved inter-annotator agreement scores between 0.6 and 0.7.

The complexity of JTS1k is slightly higher than the datasets reviewed by Mozetič et al. (2016) because it additionally includes the mixed sentiment label. More labels increase the opportunity for disagreement, which threatens the alpha score. Cieliebak et al. (2017) used an even more complex labelling scheme with the addition of the unknown label. Although they targeted high agreement levels, they reported an alpha of 0.39. In this study, an alpha of 0.5 was considered the minimum threshold, with 0.6 indicating good agreement.

[8]Hyperparameter tuning involved pre-training many different models with varying configurations. The prototype model used for sentence embeddings performed well compared to other candidates, but it was not the fully optimized BERT for Japanese Twitter.

6.4: Participatory Design for Dataset Annotation

The annotation phase was a critical and high-risk stage of the experiment, requiring a reliable pool of crowdworkers and an efficient interface for presenting stimuli and recording responses. These requirements were met by CrowdWorks[9], a Japanese freelance recruitment service. A key advantage of this platform is its interface, which is designed entirely in Japanese, enhancing usability for native speakers. Additionally, *CrowdWorks* mandates that workers verify their identity using documents issued in Japan, ensuring the recruitment of individuals who are not only fluent in Japanese but also culturally knowledgeable. The web application supports survey creation, worker recruitment, response verification, and the compilation of labelled data into CSV format, streamlining the entire data collection process.

The approach for gathering data from crowdworkers followed the methodology of participatory design (PD) (Sanders & Stappers, 2008). One of the core elements of PD is to identify the primary stakeholders and to clarify the stakes. For building JTS1k, there are two groups:

- **Developers:** This includes any public or private entity that may use the dataset for analyzing sentiment. Training and a model, even one optimized for efficiency, is costly. The dataset is only valuable if its labels are valid, and it is potentially harmful if not.
- **Social Media Users:** These individuals provide the data for analysis, with primary concerns cantered around privacy, consent, and ethical data use. For users whose tweets were selected for JTS1k, the stakes are higher due to privacy concerns. Identifiable information was properly anonymized throughout the process.

Two native Japanese speakers were recruited to help prepare the dataset for annotation. In the context of a group design process, PD advises defining roles and clarifying how team members align with the primary stakeholders. The team members are stakeholders too, and it is important to establish how they benefit from the completion of the project. The roles were defined as follows:

- **Author**: The project leader who provides direction and incorporates feedback from the team members.
- **Quality Control (QC) Specialist:** Ensures a balanced selection of clear examples for crowdworkers. Tweets were pre-labelled by both the Author and the QC Specialist, and only those with mutual agreement moved to the annotation phase. The QC Specialist, who is close

[9] https://crowdworks.jp/

to the author, knowledgeable of the research aims, and generally trustworthy, is motivated to see the project succeed.

- **Expert Annotator:** Reviews prototype datasets and provides feedback to refine the procedure. This process was repeated until both the Author and the Expert Annotator were satisfied that the task and sentiment labels were clearly defined. During the annotation phase, the Expert Annotator also participated as a crowdworker. The Expert Annotator is a web developer that meets with the Author regularly for language exchange, and they are motivated to see the project succeed.
- **Crowdworkers:** Paid 300 円 to label batches of 29 tweets. This group is particularly vulnerable because, at the Author's discretion, responses may be rejected. In this situation, renumeration is not issued, and their rating suffers, which may impact their access to future work.

Every member of the dataset annotation team represented the interests of the developers by ensuring the creation of a useful dataset. The QC Specialist and the Expert Annotator protected the interests of the crowdworkers by participating in selecting examples and writing instructions. The crowdworkers are an important group of stakeholders because they have no vested interest in the creation of JTS1k. Therefore, they are the least biased and the most representative of social media users. Incorporating the feedback of the native Japanese team members ensured the dataset annotation was a mutually beneficial experience.

6.5: Workload Distribution and Monitoring of Annotation

The dataset was divided into forty batches. Each workday, three or four batches were published, and crowdworkers usually completed several batches. Batches published in the same workday included four repeated examples for measuring self-agreement. These examples were changed each workday to ensure a varied set of responses for measuring self-agreement. Inter-annotator agreement was monitored throughout the process and consistently maintained a comfortable margin above the set acceptability threshold. Fewer than 10% of the responses were rejected, and annotations were always returned on the day of submission. The JTS1k training set was complete within a budget of €220. The PD approach guided the development of a balanced and reliable dataset, which is supported by the analysis that follows.

6.6: Evaluating Size, Balance, and Reliability

The annotation of JTS1k mostly met the original criteria of size, balance, and reliability. A small portion of data was lost due to a failure to reach consensus, resulting in a 7.5% shortfall of the size

target. While the dataset was balanced for most labels, the mixed category was underrepresented, consistent with difficulties noted by Cieliebak et al. (2017). The inter-annotator agreement measured 0.56, increasing to 0.63 when tie votes were excluded, which is well above the acceptability threshold. Additionally, the self agreement was nearly perfect, alleviating concerns about crowdworker burnout. Inter-annotator and self agreement scores were more than satisfactory, supporting that the annotations were completed diligently and in good faith.

Table 6.1 Size and Balance of JTS1k

Negative	Neutral	Positive	Mixed
274	261	253	137

The annotations were consolidated by majority vote. 75 examples were excluded because they did not reach a majority. The dataset is mostly well-balanced, with an approximate label ratio of 1: 1: 1: 0.5.

Table 6.2 Reliability of JTS1k

Inter-Annotator Agreement	
Label	Alpha
Negative	0.59
Neutral	0.5
Positive	0.67
Mixed	0.43
All	**0.56**

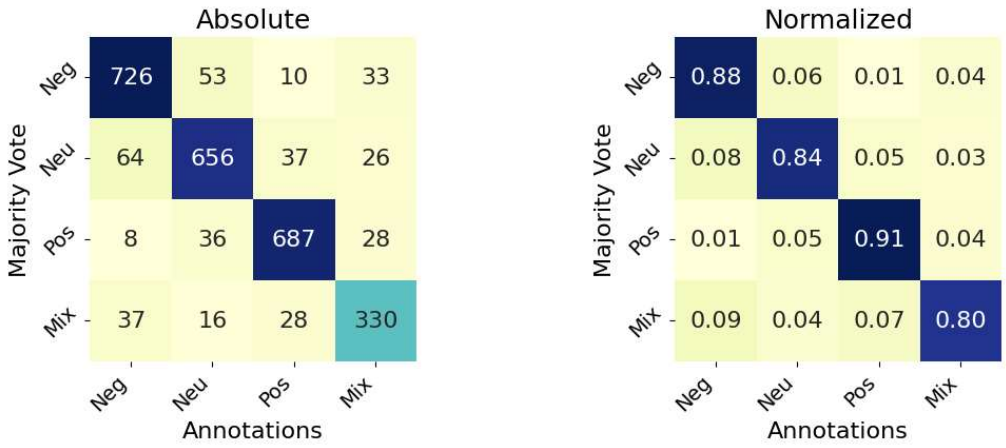
	Self-Agreement		
Worker ID	Total Responses	Repeated Responses	Self Agreement
Expert	348	48	94%
worker_001	696	111	99%
worker_002	564	65	100%
worker_003	489	60	97%
worker_004	315	35	86%
worker_005	145	20	95%
worker_006	116	16	100%
worker_007	87	12	100%
worker_008	87	8	100%
worker_009	58	4	75%
worker_010	54	8	75%
worker_011	58	0	--
.....			
worker_025	25	0	--
TOTAL	3432	387	96%

The inter-annotator agreement (left) scored an alpha of 0.57. The agreement of individual labels was calculated by treating the problem as binary. The self-agreement (right) was nearly perfect at 96%, given by repeating over 10% of the total annotations.

The alpha scores for individual labels showed varying degrees of disagreement. The positive label was the most straightforward, with an alpha of 0.67. In contrast, the mixed class produced the most disagreement, resulting in a low score of 0.43. The high level of disagreement around the mixed

class, along with difficulties in achieving balanced representation, marked the mixed category as the most challenging label. Figure 6.1 explores inter-label disagreement further with confusion matrices that compare annotations with majority vote labels.

Figure 6.1 Confusion Matrices of Annotations and Majority Vote Labels of JTS1k



Comparing the annotations with their majority vote labels provides a clearer picture of the patterns of confusion exhibited by the annotators.

The most confusion arises between the negative and neutral sentiments. This may be connected to the Japanese convention of expressing negative opinions discreetly. The concept of **“腹芸” (hara gei),** which translates to “belly art,” refers to the practice of communicating from one’s gut.

Figure 6.2 Example of a Negative Tweet with Neutral Language

This tweet, critiquing Noda’s party, conveys negative sentiment subtly rather than explicitly.

The confusion between negative and neutral sentiments is not restricted to Japanese. The tendency for people to politely express negative opinions through factual statements is a recognized challenge in sentiment analysis (Liu, 2012). In contrast, negative and positive labels show the least amount of pairwise disagreement, and mixed was confused with all other categories.

The next round of analysis uses the complete JTS1k to benchmark a series of generative AI (GenAI) models. State-of-the-art GenAI models are highly effective at zero-shot sentiment analysis

across datasets (Krugmann & Hartmann, 2024). If they perform well with JTS1k, that will support the notion that JTS1k aligns well with other sentiment analysis datasets. This investigation will also examine the response patterns of GenAI to see if they exhibit the same patterns of confusion as human annotators or if they show their own unique biases.

6.7: Benchmarking Generative AI Models for Sentiment Classification

Table 6.3 Generative AI Models Evaluated with JTS1k

Developer	Model	Parameters
Meta	Llama 2	7B 13B 70B
	Llama 3	8B 70B
@haqishen @alfredplpl Lightblue KK.	Japanese Llama 3	8B
MistralAI	Mistral	7B
	Mixtral MoE	8x7B
Google	Gemini 1.0 Pro	--
OpenAI	ChatGPT 3.5-Turbo	--
	ChatGPT 4	--

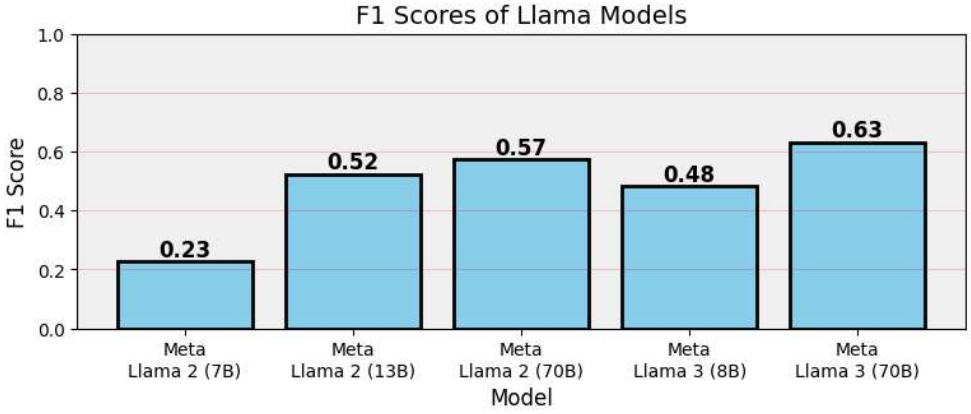
Three groups of GenAI models were evaluated. The Meta Llama models offer insights into how performance is influenced by the number of parameters and enhancements in training data and architecture. The Japanese Llama modes evaluate the benefits of language adaptation. The models by MistralAI, Google, and OpenAI represent state-of-the-art.

GenAI models are advantageous over encoder models due to their adaptability to various tasks through prompt engineering. They require minimal training examples, with the most advanced models achieving high performance in zero-shot testing (Krugmann & Hartmann, 2024; Radford et al., 2019). Prompt engineering is like giving instructions to crowdworkers: tasks must be clear, examples provided, and expected responses modeled (Giray, 2023). Some models, especially during training, struggled to produce understandable responses. The top-performing models handled natural language instructions well. For example, when ChatGPT was directed to output JSON while using CrowdWorks instructions, it generated precise and well-structured responses. In contrast, smaller models often returned responses that were fragmented, overly detailed, or formed incorrectly. These models showed improvement when given detailed examples of the input-output cycle, as seen in Appendix 3. Using explicit examples to impose structure was effective across all models but it used

more tokens than necessary. For advanced models that do not require detailed instructions, this approach is inefficient. All models were tested with the same prompt setup in this study, which worked well here but is generally costly and not advisable for more sophisticated models.

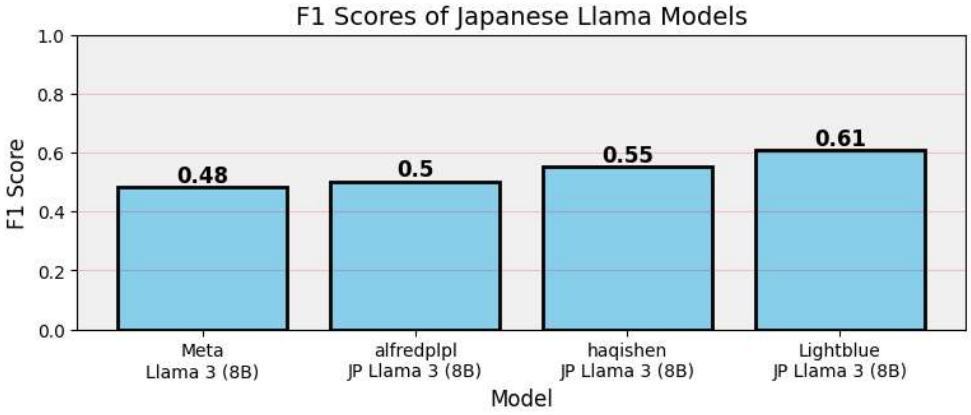
The GenAI models' performance was assessed based on their ability to classify the JTS1k dataset with few-shot learning. Newer, larger models were expected to perform better. The Japanese Llamas were anticipated to surpass the original. Among top-tier models, OpenAI was expected to lead.

Figure 6.3 Performance of Llama Models on JTS1k



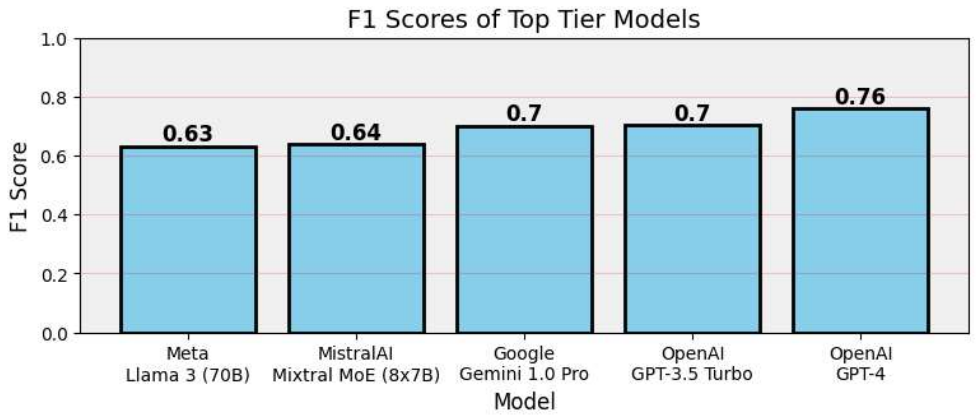
Adding more parameters generally improves model performance, albeit with diminishing returns. The Llama 3 models marked a significant upgrade over their predecessors. The smallest Llama 3 nearly matched the performance of the mid-sized Llama 2, whereas the smallest Llama 2 performed only slightly better than random guessing. The largest Llama 3 surpassed its counterpart by 10%

Figure 6.4 Performance of Japanese Adapted Llama Models on JTS1k



Independent developers, noting suboptimal performance on Japanese tasks, fine-tuned the smallest Llama 3 using a Japanese training corpus. Their efforts led to performance enhancements, with the top Japanese-tuned Llama nearly reaching the F1 score of the standard Llama 3 (70B).

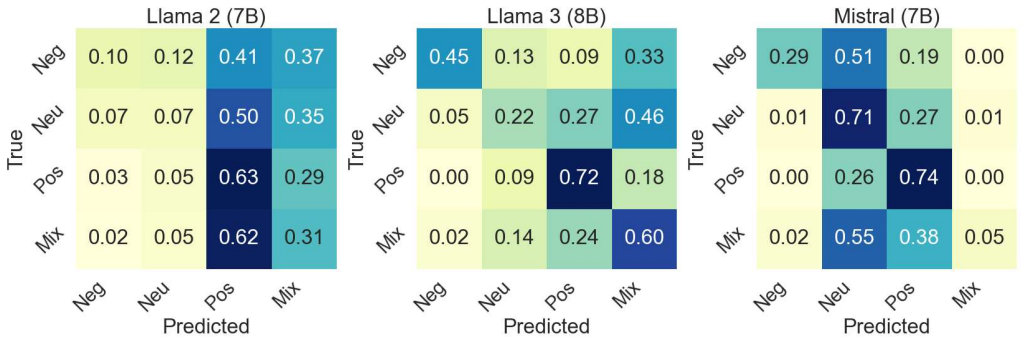
Figure 6.5 Performance of Top-Tier Models on JTS1k



MistralAI demonstrates promise in the Mixture of Experts (MoE) architecture with Mixtral. Despite having fewer parameters, it outperforms Llama 3. GPT 3.5 matched the performance of Google Gemini. GPT 4 set a high benchmark that would be difficult to compete with.

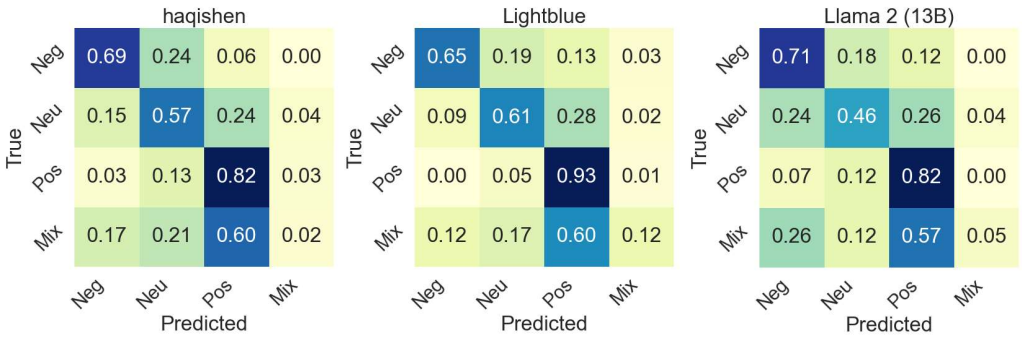
Further analysis focuses on the specific responses provided by GenAI models. The crowdworkers that annotated JTS1k exhibited three response patterns: confusion between negative and neutral classes, clear distinction between negative and positive classes, and challenges classifying mixed classes. Examination of GenAI model responses show whether they exhibit similar patterns. Two alternative patterns of bias emerged: the 'Optimistic' models and the 'Polarized' models.

Figure 6.6 Confusion Matrices of Responses by the 'Optimistic' Models



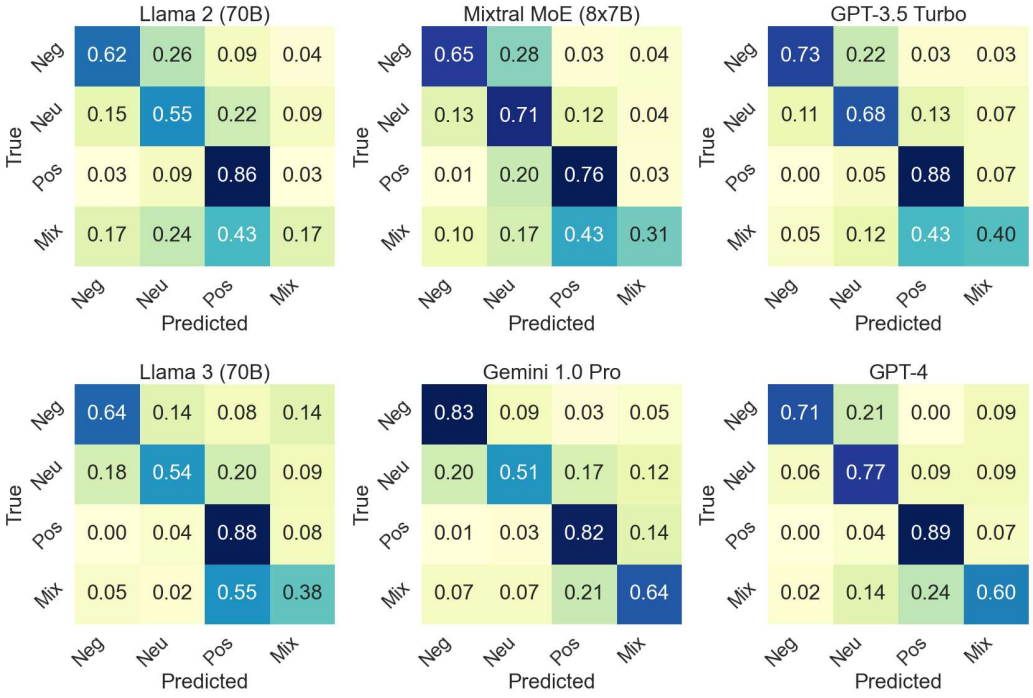
Likely influenced by a directive to please users, the 'Optimistic' models are hesitant to select the negative category and readily choose positive (Buscemi & Proverbio, 2024). This pattern is most prevalent in the smallest Llama 2, which also had the lowest F1 score. The smallest Llama 3 shows significant improvement. Its negative classifications are highly precise, but low in recall. The Mistral model avoided both the negative and the mixed categories.

Figure 6.7 Confusion Matrices of Responses by the 'Polarized' Models



The 'Polarized' models made accurate predictions about the three main sentiment categories but rejected the mixed class. These included all the Japanese Llamas, which is an interesting development given that the original Llama 3 was overzealous with the mixed category.

Figure 6.8 Confusion Matrices of Responses by Top-Tier Models



These models are the most consistent with human responses. There is confusion between neutral and negative, distinction between negative and positive, and mixed is a consistent challenge. Showcasing its superiority, GPT-4 classifies all categories with human-like performance.

Most GenAI models showed biases that did not match human behaviour, pointing to the complex challenge of designing AI to be safe, compliant, and accurate. The models often struggled with the mixed category, a challenge that human annotators managed more effectively. Notably,

GPT4 classified mixed examples with high precision and recall. This suggests that as models grow and sophistication, they can make deeper inferences that better align with human judgement.

6.9: Cross-Lingual Transfer

The final evaluation of JTS1k focuses on cross-lingual transfer, which tests a model's ability to generalize across languages (Conneau et al., 2020). The study uses the Twitter Sentiment Multilingual (TSML) dataset, containing 24,000 evenly balanced examples across eight languages (Barbieri et al., 2020). The labels are negative*,* neutral*, and* positive. XLM-T is considered highly capable of cross-lingual transfer and has shown competence with various cross-lingual training and testing configurations using TSML. This study aims to demonstrate JTS1k's robustness as a training set by examining its compatibility with XLM-T and TSML. The experiment compares three datasets:

- **JTS1k** is the target dataset, reduced to just under 800 examples to match TSML's labelling scheme of negative, neutral, and positive.
- **WRIME is** Japanese emotion and sentiment analysis dataset representing social media (Suzuki et al., 2022). It uses continuous polarity scores converted to categorical labels as described in Chapter 9 and contains over 30,000 examples.
- **SB10K** is a German Twitter sentiment analysis dataset from the pre-transformer era (Cieliebak et al., 2017). It includes 10,000 examples but is less balanced and has inconsistent annotations.

This experiment fine-tunes the base and large versions of XLM-T on each dataset. The models are compared with the state-of-the-art XLM-T sentiment model that was fine-tuned on all languages from the TSML dataset. JTS1k will be considered successful if it demonstrates compatibility with the other datasets. Table 6.4 documents the F1 scores of models on the test splits of the three experimental datasets and the language splits of TSML.

Within the test splits of the three datasets, the top scores were achieved by the large models with matching training data. All models evaluated extremely well with JTS1k, with WRIME showing the strongest transfer. The XLM-T sentiment model performed significantly better on JTS1k than the other two datasets. The models fine-tuned on JTS1k transferred well to WRIME but were the lowest performers on SB10k. WRIME models tested better than JTS1k models, but XLM-T sentiment showed the best transfer. The SB10k models performed decently on the two Japanese sets but were the lowest performers overall.

Within the language splits of TSML, XLM-T sentiment is the best performer in almost every language. The SB10k models are dominant in their native language, German. Interestingly, the large JTS1k model achieved the top score in Arabic, an unexpected and difficult-to-interpret result. When comparing the large models, the JTS1k model earned the best results, scoring closely to the XLM-T sentiment on nearly every language split. Although it struggled with French, it scored higher than the others. Comparing the base models, WRIME appears to be a better overall fit. Despite underperforming in a few languages, it tops JTS1k in most languages. Outside of German, SB10k underperformed in all categories.

Table 6.4 Cross-Lingual Transfer with XLM-T

			F1 Score on Test		
Model	**Train**	**Size**	**JTS1k**	**WRIME**	**SB10k**
XLM-T	**TSML**	base	0.75	0.67	0.61
	JTS1k	large	**0.82**	0.69	0.52
		base	0.78	0.68	0.48
	WRIME	large	0.76	**0.78**	0.52
		base	0.77	0.74	0.54
	SB10k	large	0.71	0.65	**0.67**
		base	0.64	0.65	0.65

			F1-Score on TSML Language Splits								
Model	**Train**	**Size**	**All**	**AR**	**EN**	**FR**	**DE**	**HI**	**IT**	**PT**	**ES**
XLM-T	**TSML**	base	**0.70**	0.67	**0.73**	**0.74**	0.75	**0.57**	**0.69**	**0.76**	**0.69**
	JTS1k	large	0.66	**0.68**	0.69	0.52	0.70	0.55	0.68	0.73	0.69
		base	0.60	0.62	0.61	0.53	0.61	0.53	0.62	0.56	0.62
	WRIME	large	0.62	0.56	0.64	0.44	0.67	0.54	0.60	0.71	0.69
		base	0.61	0.64	0.69	0.40	0.70	0.50	0.54	0.66	0.65
	SB10k	large	0.59	0.47	0.56	0.25	**0.81**	0.51	0.61	0.69	0.63
		base	0.58	0.46	0.59	0.32	0.76	0.50	0.54	0.66	0.64

Compares the performance of fine-tuned XLM-T on the test splits of experimental datasets and the language splits of TSML. Peak scores are highlighted in bold.

6.8: Interim Conclusion

This chapter addressed two hypotheses. First, it demonstrated that JTS1k, optimized for balance and reliability, is a high-quality dataset yields comparable results to larger datasets. It elicited a range of performance from various decoder models and evaluated well with top-tier models. In the cross-lingual transfer experiment, the models fine-tuned on JTS1k performed closest to the upper bound baseline. This does not imply that JTS1k is superior to the other two datasets, but it suggests

that it is best aligned with the multilingual Twitter domain. These unexpected results strongly indicate the quality of JTS1k.

Participatory design proved effective for the annotation phase, emphasizing iterative feedback and continuous improvement. By involving native Japanese speakers who understood the stakeholders' needs, the team effectively refined the instructions and procedures before investing in crowdworker resources. This collaborative approach resolved issues early in the process, ensuring the creation of a reliable and well-structured dataset that met quality standards and remained within budget.

Chapter 7: Adapting the Vocabulary for Twitter

This chapter explores the design of the tokenizer for the BERT models tailored for Japanese Twitter. The tokenizer is based on the configuration from Japanese BERT and incorporates a specially adapted vocabulary. The chapter begins with an overview of the Japanese writing system, focusing on the various character families used. It then discusses various web elements and how they affect preprocessing and tokenizer design. WordPiece tokenizers are trained using a specially prepared corpus. This study compares two corpora representing different levels of refinement, and both were used to train a unique vocabulary. The refined corpora are compared with an analysis of token frequency. The vocabulary of the original and Twitter tokenizers is compared, considering both character families and parts of speech. The Twitter tokenizers have adopted colloquialisms, neologisms, and multilingual expressions, better aligning them with the Twitter domain.

7.1: Overview of Japanese Writing

Japanese uses four character sets: hiragana, katakana, kanji, and romaji. Hiragana, the indigenous Japanese syllabary, consists of 46 symbols. It is the primary script for grammatical functions, such as verb inflections and conjunctive particles. This flexible character family is also used for names, phatic expressions, and native Japanese words that lack kanji representations. Katakana, a parallel syllabary, primarily represents foreign words. Along with hiragana, it is commonly used for interjections and onomatopoeia. Kanji, the largest character set, consists of logographic characters mainly derived from Chinese. There are tens of thousands of kanji, but only a fraction is routinely used. Romaji, Japanese use of Roman letters, is most frequently used with named entities, like place names and international brands. It is also commonly used when culturally appropriate, such as in expressing basic English words and phrases or in reference to pop culture. These character sets are somewhat interchangeable, and flouting convention might communicate some pragmatic intent. Nevertheless, appropriate character use is critical to the coherence of Japanese writing.

7.2: Considerations for Preprocessing

Converting raw text into machine-readable inputs is a longstanding challenge in NLP. Modern algorithms, such as WordPiece and SentencePiece, are highly effective, but the field continues to evolve (Devlin et al., 2019; Godey, et al., 2022; Kudo et al., 2018). Twitter data presents specific challenges for preprocessing. For example, tweets often contain web elements, like URLs, which should not be tokenized like regular text because they introduce noise that models struggle to handle. Additionally, preserving the text's structure is important, and the informal and fragmented nature of

tweets can complicate tokenization. Designing a vocabulary for Twitter data must address these issues to ensure accurate and meaningful text analysis.

Special Tokens for Mentioned Users and URLs

Twitter commonly features three web elements: URLs, mentioned users, and hashtags. The State-of-the-art models, XLM-T and BERTweet, employ different strategies for handling web elements (Barbieri et al., 2022; Nguyen et al., 2020). To accommodate mentioned users, both models mitigate noise through substitution with a normalized token. XLM-T uses the same strategy for URLs. On the other hand, BERTweet filtered tweets containing URLs from their training corpus as a precaution against spam, and their documentation recommends removing URLs during preprocessing. Hashtags, which often carry meaningful context, are left unmodified during preprocessing by both models.

The vocabulary design follows the approach of Barbieri et al. (2022) by substituting both URLs and mentioned users with normalized tokens. As an additional precaution, they were assigned special token status. Special tokens are normally reserved for specific functions, but they have the advantage of never being combined with other tokens. This strategy helps preserve the structural integrity of the tweets while minimizing the noise introduced by URLs and mentioned users.

Special Tokens for Newlines

Models also differ in their handling of structural tokens, like newlines. In the context of a long document, newline characters are cleaned out during preprocessing because they are not meaningful. However, in the context of short tweets, newlines can be important structural elements. XLM-T, which inherited its tokenizer from XLM-R (Conneau et al., 2020), treats newlines as general whitespace tokens. In contrast, BERTweet assigns newlines their own token status. WordPiece tokenizers are generally configured to remove newlines during preprocessing, but the HuggingFace tokenizer can be easily adjusted to treat newlines as normal tokens. This study followed Nguyen et al. (2020) by assigning newlines a unique token. This approach helps preserve the structural integrity of tweets, ensuring that important formatting cues are not lost during preprocessing.

7.3: Preparing the Tokenizer Training Corpora

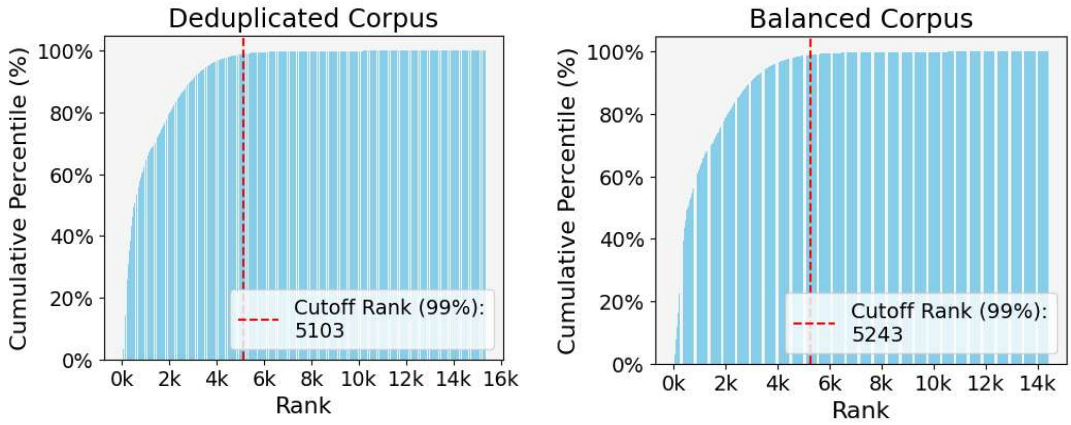
The WordPieceTrainer[10] in the Transformers library generates a vocabulary from a training corpus, aiming to minimize sequence lengths and optimize coverage (Wolf et al., 2020). This tool is designed for use with space-segmented languages. For Japanese, which does not segment words with spaces, preparing the training corpus involved some additional considerations. The

[10] HuggingFace Tokenizer Trainers: https://huggingface.co/docs/tokenizers/api/trainers

BertJapaneseTokenizer streamlines preprocessing by normalizing raw text and pre-tokenizing it with MeCab.[11] To prepare the training corpora for the WordPieceTrainer, the refined Twitter corpora were pre-tokenized by the same mechanism.

The WordPieceTrainer prioritizes individual characters in its vocabulary. This approach poses a problem with Twitter data, which contains a wide variety of characters, many of which appear only once or a few times. For instance, the Deduplicated Twitter corpus contains over 16,000 unique characters. To optimize character selection, corpus analysis is necessary. In this study, characters were ranked by the number of tweets in which they appeared. Histograms were generated to identify the minimum number of characters required to cover 99% of the corpus. The results from both corpora are plotted in Figure 7.1, providing a clear visualization of the cutoff point for effective vocabulary coverage.

Figure 7.1 Histograms for Selecting Character Rank Cutoff



Different training corpora yield different vocabularies. The larger Deduplicated corpus is covered with 5,103 unique characters, compared to 5,233 for the Balanced corpus.

Another important consideration is the inclusion of essential characters, especially in Japanese, which uses abundant kanji. The original Japanese BERT includes over 6,000 kanji, many of which may not be frequently used in Twitter. Japanese Twitter tends to use more hiragana and katakana compared to more formal texts. However, it is important to include certain kanji that are essential for general understanding, even if they are underrepresented in the Twitter corpus. To ensure these essential kanji are covered, the WordPieceTrainer was initialized with the jōyō kanji[12], a standardized set of commonly used characters in Japanese.

[11] The BertJapaneseTokenizer is explained in more detail in Appendix 4.
[12] Accessible at https://www.kanjidatabase.com/.

The WordPieceTrainer, when provided with a training corpus, a vocabulary size, a character limit, and a list of essential characters, generates an optimized vocabulary. To evaluate the vocabulary, the Twitter corpora were tokenized using the original Japanese BERT tokenizer and the two tokenizers yielded by the refined corpora. The analysis searches for signs of improvement, including shorter sequence lengths, a reduction in the frequency of unknown tokens, and a better representation of low-frequency tokens.

7.4: Analysis of Token Frequency

Figure 7.2 Rank-Frequency Distribution of Tokens by Tokenizer and Corpus

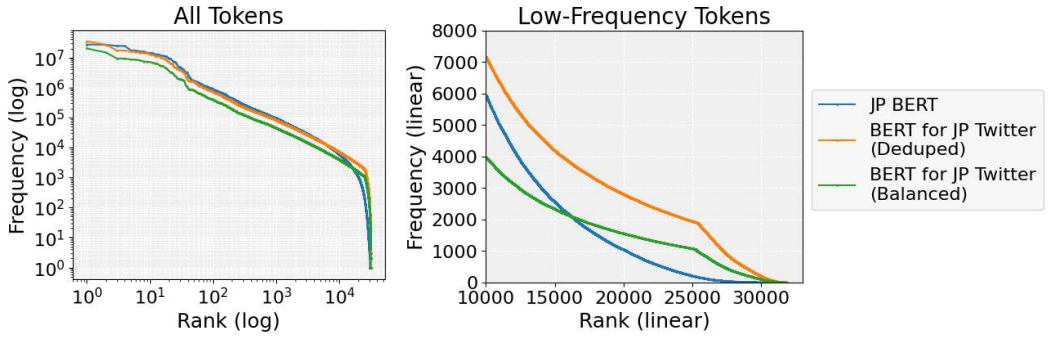


Table 7.1 Token Distribution of the Training Corpora

	Japanese BERT	BERT for Japanese Twitter	
Corpus	Deduplicated	Deduplicated	Balanced
Total Tweets	27.9M	27.9M	14.5M
Total Tokens	928M	838M	458M
Avg Tokens per Tweet	33	30	31
Vocab Size	32768	32000	32000
Observed Vocabulary(%)	95.04%	99.65%	99.58%
[UNK] Token (%)	2.89%	0.05%	0.05%
Top 100 Tokens (%)	53.37%	50.41%	49.85%
Bottom 16K Tokens (%)	1.23%	3.62%	3.64%

The original Japanese BERT vocabulary was well optimized for general text. When tokenizing the Twitter corpus, it utilized over 95% of its vocabulary, demonstrating its efficiency in covering standard Japanese language use. However, it struggled with the abundance of non-standard characters found on Twitter, resulting in nearly 3% of tokens being returned as unknown. In contrast, the Twitter-specific tokenizers returned only 0.05% unknown tokens, showing a significant improvement in handling the diverse and informal nature of Twitter text.

The average sequence lengths dropped by almost 10% when using the Twitter tokenizers compared to the original Japanese BERT. Shorter sequences are more efficient to train, and even small differences can have a significant impact over the long pre-training process. Furthermore, the original

BERT had less frequent low-frequency tokens. Around the midpoint of the vocabulary by rank frequency, the number of tokens in the original BERT drops below that of the Balanced corpus, despite the count being from the Deduplicated corpus, which is twice the size. Comparing the two Twitter tokenizers, the token distribution is quite similar. The Balanced tokenizer counts about half as many tokens as the Deduplicated tokenizer, reflecting the differences in corpus size and content.

7.5 Change in Vocabulary after Twitter Adaptation

This section examines the differences in vocabulary between the original Japanese BERT tokenizer and the two tokenizers derived from the refined Twitter corpora. The expectation is that the new vocabulary will include terms characteristic of Twitter usage, such as colloquialisms, neologisms, and multilingual expressions. Conversely, the vocabulary is expected to exclude domain-specific terms like archaic words, historical figures, and mathematical and scientific terms. This analysis begins by exploring the changes in character representation.

Table 7.2 Change in Vocabulary by Character Family

Domain	Character Family	Japanese BERT	BERT for Japanese Twitter	
			Deduplicated	Balanced
Japanese	Hiragana	1495	3232	3147
	Kanji	21574	16334	16059
	Katakana	6321	5994	6009
General	Romaji	2193	2245	2496
	Digit	595	313	313
	Punctuation	186	332	326
	Special	5	8	8
Twitter	Symbol	246	630	648
	Script	129	610	709
	Pictograph	14	2302	2285
Total		**32758**	**32000**	**32000**

The vocabularies of the three tokenizers were classified by their character family. The Special tokens have additional functions. The Twitter model added three Special tokens: [URL], [USER], and '\n' (newline). The Script characters are from languages outside of the target. They usually appear in the context of kaomoji (Bedrick et al., 2012). The Pictographs mostly consists of emoji, but also encompasses decorative elements like dingbats.

The change in vocabulary represents a significant shift in training material. The number of kanji terms was reduced by about 25%, while hiragana terms more than doubled. This is consistent with the Twitter domain, where hiragana is used more frequently than kanji. Surprisingly, the number of katakana terms remained equivalent. Although katakana is nearly as productive as hiragana, this

suggests that many katakana terms were lost during the Twitter adaptation. Pictographs, which were scarcely represented in the original vocabulary, have now become the fifth largest category. The two Twitter tokenizers differ by only 1,200 terms. There are some significant differences in the composition of the vocabulary, particularly in the kanji, romaji, and script. However, these differences are difficult to interpret. At this level of analysis, the two Twitter tokenizers can be considered roughly equivalent and distinct from the original.

The analysis continued by classifying the vocabulary based on parts of speech. The method used for classification considers how the part of speech for certain terms changes in different contexts. MeCab, which combines tokenization with part of speech tagging, was used to tokenize and tag the Deduplicated corpus. Every token in the vocabulary was classified with a frequency distribution across the different parts of speech, reflecting their observed usage in the Twitter corpus. This methodology provided a context-aware analysis that accurately represented Twitter.

Table 7.3 Change in Vocabulary by Part of Speech

Class	POS	POS (Eng)	Japanese BERT	BERT for Japanese Twitter	
				Deduplicated	Balanced
Noun	名詞	Noun	20241	16358	16370
Verb	動詞	Verb	1881	2866	2756
Descriptive	形容詞	Adjective	318	666	652
	形状詞	Adnominal	305	425	407
	副詞	Adverb	303	743	721
	接頭辞	Prefix	206	211	211
	接尾辞	Suffix	671	705	707
Funtional	助動詞	Aux Verb	111	217	211
	接続詞	Conjunction	35	36	33
	連体詞	Determiner	30	48	44
	助詞	Particle	112	182	178
	感動詞	Interjection	76	226	212
	代名詞	Pronoun	73	128	121
Peripheral	補助記号	Punctuation	306	1686	1726
	記号	Symbol	516	922	927

The POS column represents the original tag as given by MeCab. This analysis did not consider subword tokens. The frequencies represent the number of terms that were tagged in at least 10% of contexts. Consequently, many terms were counted across several parts of speech.

These assumptions are tested through a structured exploration of the change in vocabulary. The analysis focuses on parts of speech grouped by class. From the vocabulary gained, the analysis searches for examples of neologisms, colloquialisms, and multilingual expressions. It also looks for

common themes among the items lost. Examples of terms with loaded sentiment are provided. For simplicity, this analysis groups the vocabularies from the Twitter tokenizers together.

Nouns

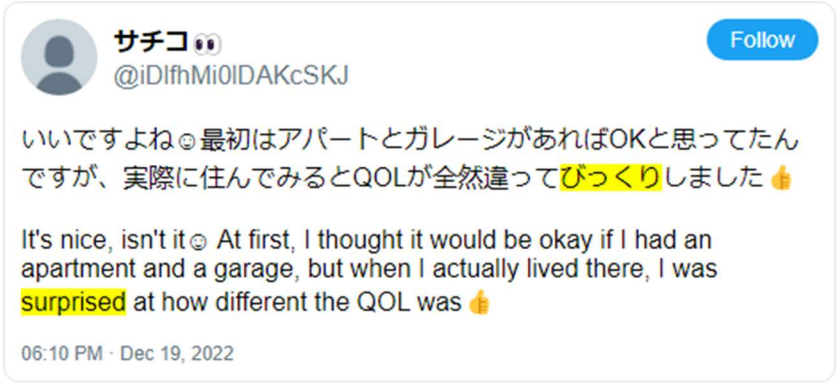
Table 7.4 Change in Noun Vocabulary

	Japanese BERT	BERT for Japanese Twitter		
		Count	Gained	Lost
Kanji	17490	**13895**	2571	6166
Katakana	5323	**5603**	1743	1463
Romaji	1924	**2034**	658	548
Hiragana	930	**1910**	1026	46
Digit	544	**300**	32	276
Total	**26211**	**23742**	**6030**	**8499**

The largest change in vocabulary was observed among nouns, making this category challenging to characterize comprehensively.

Many terms were gained across different character families. A significant number of terms were lost in kanji, katakana, and romaji, reflecting their use in more formal domains. However, hiragana terms were maintained, supporting their representation in casual contexts. New colloquialisms include expressions like すまほ (sumaho) for smartphone and パソコン (pasokon) for personal computer. Notable neologisms such as リモートワーク (remote work) and ワンオペ (one operation) reflect changing social and work dynamics. Katakana is the main character family for multilingual expressions, like カフェ (cafe) and アルバイト (part-time job) from the German "Arbeit."

Figure 7.3 Example of a Noun with Loaded Sentiment



The term びっくり (surprised) expresses emotional intensity and can convey either positive or negative sentiment. In this context, the user is expressing satisfaction with their living situation.

From the vocabulary lost, many terms are related to historical events or figures, such as レオナルド (Leonardo) and スターリン (Stalin). Others are historical terms like 治世 (reign) and 封建 (feudal). Domain-specific terms used in science, mathematics, or specialized industries, such as アルゴリズム (algorithm), 分子 (molecule), 製鋼 (steelmaking), and 技師 (engineer), were also removed. Additionally, rare or less commonly used terms, including obscure kanji like 罕 and 靨, geographic names such as ルイジアナ (Louisiana), and obsolete terms like 秬 and 侖, were excluded.

Verbs

Table 7.5 Change in Verb Vocabulary

	Japanese BERT	BERT for Japanese Twitter		
		Count	Gained	Lost
Kanji	4502	5600	1480	382
Hiragana	678	1283	631	26
Katakana	321	579	259	1
TOTAL	5501	7462	2370	409

This category loses a significant number of kanji. For hiragana and katakana, terms double without significant loss from the original vocabularies.

New terms include colloquialisms like やめる (to stop/quit) and つぶやく (to mutter/tweet) are frequently used in casual conversation. Examples of dialectal variation are おる (to be) and いける (to be good at), which are typical expressions of a Kansai speaker. New neologisms include ググる (to Google), which has achieved verb status like English. An interesting multilingual expression is サボる (to slack off), derived from the French "sabotage." Others are ダンスする (to dance) and メールする (to email), which combine katakana phonetizations of the English pronunciation with the auxiliary verb する. Verbs loaded with sentiment include たのむ (to request) and がんばる (to do one's best), which carry positive connotations of trust and determination, while あきらめる (to give up) and さけぶ (to shout) convey negative feelings of defeat and distress.

Many verbs lost are more formal or literary, such as 遂げ (accomplish) and 治め (govern). Some refer to very specific actions, like 乗り出し (embark) and 催さ (host), which are typically used in formal invitations and public addresses. Specialized and technical terms, like 採る (adopt) and 施す (implement), are more suited to professional contexts. Additionally, some verbs are archaic or less commonly used, like 經 (pass through) and 患う (suffer).

Figure 7.4 Example of a Verb with Loaded Sentiment

The term つまら is a stative verb that best translates to the adjective, boring.

Descriptive Terms

Table 7.6 Change in Descriptive Vocabulary

	Japanese BERT	BERT for Japanese Twitter		
		Count	Gained	Lost
Kanji	2102	2518	552	136
Hiragana	574	1366	810	18
Katakana	367	714	357	10
Total	3043	4598	1719	164

This descriptive category GenAIns a significant number of kanji, hiragana, and katakana. Notably, Hiragana terms more than double. Fewer terms, mostly kanji, were lost.

Japanese is recognized for its rich vocabulary of onomatopoeia. New vocabulary items include literal examples like ゴロゴロ (goro-goro) and ザーザー (zaa-zaa), which express the sounds of rolling thunder and heavy rain. Figurative examples such as ほっ (hotto) and ふわっ (fuwa) convey feelings of relief and softness. Colloquialisms like ヤバい (yabai), which can denote something extremely good or bad, and めちゃ (mecha), an informal intensifier like "very," are commonly used. Neologisms like ウケる (ukeru), meaning "that's funny," and リア充 (ria-juu), referring to someone content with their offline life, have also become part of the modern lexicon.

Many of the terms lost are formal or literary, such as 顕著 (remarkable), 厳格 (strict), and 迅速 (rapid). Some words are rare or complex, like 茫 (vague), 黯 (dark), and 纔 (just barely). Compound or derived forms like 細長い (slender) and 目立つ (conspicuous) are often more context specific. Some words are archaic or less commonly used, like 乗 (ride) and 猝 (sudden). Certain words denote highly specific descriptions, such as 凶暴 (ferocious) and 甚大 (immense). Specialized and technical

terms, such as メタリック (metallic) and コミカル (comical), are more suited to professional contexts.

Figure 7.5 Example of a Descriptive Term with Loaded Sentiment

The figurative onomatopoeia うるうる describes the feeling of tearing up. Normally used to express sadness, this example conveys a positive and moving experience.

Functional Terms

For the functional items, many new terms were gained and very few lost. The analysis will now only discuss new terms. The interjections and pronouns are particularly interesting and are discussed separately.

Table 7.7 Change in Interjection Vocabulary

	Japanese BERT	BERT for Japanese Twitter		
		Count	Gained	Lost
Hiragana	50	**197**	147	0
Katakana	46	**60**	17	3
Kanji	7	**13**	7	1
Total	**103**	**270**	**171**	**4**

The number of interjections grows significantly, with hiragana terms more than triple.

Most interjections are composed with hiragana. Many convey emotional intensity where the intended sentiment depends on context, such as あれー, expressing shock. Others communicate a lack of intensity, such as うーむ, which conveys deep thought. Some of the new interjections are dialectally marked. おおきに (ookini) is a thankful remark from the Kansai dialect. わっしょい (wasshoi), chanted by festival-goers carrying heavy a heavy shrine, encapsulates the communal joy and excitement during traditional celebrations. On Twitter, it may be used to show resilience in a difficult situation. くっそ (kusso) is a rude expression that conveys frustration or annoyance. なんちゃって (nanchatte) is a light-hearted expression that means "just kidding."

Figure 7.6 Example of an Interjection with Loaded Sentiment

こら is diminutive expression used to demand the attention of the addressee. It is often used by parents when scolding their children.

Table 7.8 Change in Pronoun Vocabulary

		BERT for JP Twitter		
	Japanese BERT	Count	Gained	Lost
Hiragana	45	101	56	0
Kanji	39	46	11	4
Katakana	15	30	16	1
TOTAL	99	177	83	5

Most of the original vocabulary of pronouns is retained. The hiragana terms more than double.

Figure 7.7 Example of a Pronoun with Loaded Sentiment

The second person singular pronoun, テメエ is usually rude and derogatory, although it may be used affectionately between those that are close, in a typical context, it is comparing the addressee to an animal.

Japanese personal pronouns are pragmatically loaded, capable of conveying subtle aspects of the speaker's attitude and social standing. Pronouns like おれ (ore), わし (washi), and ワタシ (watashi) vary significantly in terms of formality and gender implications, with おれ being casually

male-oriented, わし an older, somewhat outdated male pronoun, and ワタシ the neutral standard. Meanwhile, forms like おまえ (omae) reflect either familiarity or disrespect, depending on the context. わたくし (watakushi) as a very formal "I" carries a positive, respectful sentiment, whereas 貴様 (kisama) nowadays has a clearly negative connotation. The new vocabulary also includes basic demonstrative pronouns such as ソコ (soko, there), ソレ (sore, that), and コチラ (kochira, this).

Table 7.9 Change in Functional Vocabulary

	Japanese BERT	BERT for Japanese Twitter		
		Count	Gained	Lost
Hiragana	304	**570**	270	4
Kanji	88	**102**	26	12
Katakana	37	**41**	5	1
Total	**429**	**713**	**301**	**17**

Most of the original vocabulary of functional items is retained. Hiragana terms grow significantly.

Functional vocabulary is traditionally a closed class, and the new terms are generally more flexible variations of the standard vocabulary. Core auxiliary verbs, such as でしょう, ましょう, だろう, たろう, and やろう, were substituted with shortened forms dropping the terminal う. Commonly used elongated forms are represented as well, like なー, ねー, ねえ, and よー. Dialectal variations include さかい (because) from Kansai, じゃろ (probably) from Hiroshima, and ばってん (but) from Kyushu. Colloquialisms such as っけ (was it?), っす (yup), なんか (something like), and ちゃう (different) are common. Neologisms include でしゅ and でちゅ (baby-talk versions of です) and ちゃえ (a casual form of "to finish doing something"). Shortened forms like カノ (his/her) and コノ (this) reflect the trend towards conciseness.

7.6: Comparison of Tokenizers by Unique Vocabularies

The final analysis examines the vocabularies unique to each tokenizer. Table 7.10 segments these unique vocabularies by part of speech. It shows that the main distinction of the Balanced vocabulary is a larger selection of nouns. The Balanced vocabulary contains more technical terms and specialized words (釣行, 馬肉), while the Deduplicated vocabulary is more casual, incorporating modern slang and informal language (ウンコ, ちんこ). The Balanced vocabulary also includes a higher frequency of proper nouns and brand names, indicating a focus on specific entities and products (BAND, Beer, Fuji). This is significant because named entities are considered a sign of

business influence. An increased number of named entities implies that balancing users may have increased the representation of business accounts rather than reduced it. On the other hand, the Deduplicated vocabulary emphasizes common, everyday words and colloquial expressions (あの世, こたん, ひじき). Additionally, the Deduplicated corpus features significantly more terms from every other part of speech class. These results suggest that the Deduplicated vocabulary may have more emotive capacity, making it a better fit for sentiment analysis.

Table 7.10 Comparison of Unique Vocabularies from Each Tokenizer

		BERT for Japanese Twitter	
POS Class	Japnese BERT	Deduplicated	Balanced
Noun	8504	315	**447**
Verb	409	**107**	11
Descriptive	164	**75**	24
Pronoun	5	**7**	0
Interjection	4	**14**	0
Functional	17	**17**	0
Peripheral	235	69	**110**

Compares unique items between tokenizer by part of speech. The original Japanese BERT tokenizer is markedly different in most categories. The difference between the two Twitter tokenizers is more subtle.

7.7: Interim Conclusion

Using the WordPieceTrainer, an optimized vocabulary was produced for Japanese Twitter that minimized sequence lengths and unknown tokens. This chapter tested the hypothesis that a data-driven methodology would naturally lead to the inclusion of colloquialisms, neologisms, and multilingual terms while excluding formal, domain-specific terms. An exploration of the vocabulary supported this hypothesis. It was expected that the Deduplicated corpus, which includes heavy contributions from relatively few prolific authors, would skew towards business interests. However, the comparison of unique vocabularies showed that the Balanced corpus includes more named entities, indicating a greater business influence.

A final analysis of token distribution compares the vocabularies of BERT for Japanese Twitter with LUKE and XLM-T (Barbieri et al., 2022; Yamada et al., 2020). The training corpora for LUKE included a selection of web text, and its vocabulary encompasses basic emoji. LUKE uses a SentencePiece tokenizer (Kudo et al., 2018), which prioritizes shorter token units but can also take up phrase-level tokens. Phrase tokens commonly occur with phatic language in a monolingual SentencePiece vocabulary.

Example 7.1: SentencePiece vs. WordPiece Tokenization

Input: “ありがとうございます” (Thank you very much)

LUKE (SentencePiece): ['▁ありがとうございます']

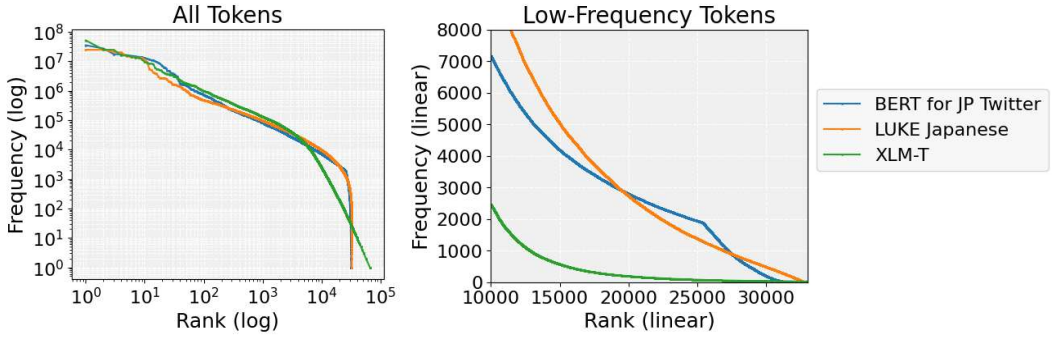
BERT (WordPiece): ['ありがとう', 'ござい', 'ます']

The LUKE tokenizer combines this common utterance into a single token, whereas the BERT tokenizer separates the interjection, the verb, and the auxiliary verb.

Phatic language is strongly represented on Twitter, contributing to the shorter tokenized lengths observed with the LUKE tokenizer. Despite having lower token counts, the percentage of unknown tokens remains lower than with the original Japanese BERT, making the LUKE tokenizer more suited for Twitter data. Pre-training LUKE on the Twitter corpus without changing the vocabulary might be worth exploring.

XLM-T has a much larger vocabulary that spans 100 languages, providing an initial advantage for handling various scripts and a large selection of non-standard characters. However, tokenized sequences are more atomic with XLM-T, resulting in longer sequence lengths and less efficient training. BERT for Japanese Twitter is more optimized for the target domain, returning fewer unknown tokens and utilizing more of its vocabulary.

Figure 7.8 Token Distribution of Twitter Corpus by Alternative Tokenizers



	BERT for JP Twitter	LUKE	XLM-T
Total Tokens	838M	766M	926M
Avg Tokens per Tweet	30	27	33
Vocab Size	32000	32772	250002
Observed Vocabulary(%)	99.65%	99.91%	28.90%
[UNK] Token (%)	0.05%	2.57%	0.40%

Chapter 8: Pre-Training BERT for Japanese Twitter

This chapter describes the hyperparameter tuning, pre-training, and evaluation of **BERT for Japanese Twitter**. It begins by detailing the setup for adapting BERT to a new domain, including the instantiation of two base models for pre-training, one for each refined corpus. It then discusses the resource-intensive nature of pre-training and the strategies used to manage budget constraints. The training setup was optimized for the available hardware, and a systematic exploration of tunable hyperparameters yielded a targeted search window. Dozens of candidate models were pre-trained within this search window and evaluated by fine-tuning, effectively exploiting both the knowledge gained from exploration and the available computational resources. The best model, continuing as **BERT for Japanese Twitter**, performed well on both sentiment analysis and defamation detection tasks. This model is evaluated with a series of techniques to assess the semantic networking of the embedding matrix, the quality of masked token predictions, and its competency across Twitter and general domain tasks.

8.1: Considerations for Data Leakage

Data leakage refers to an undesirable training scenario where models are trained and tested on the same examples. Consistent and well-documented dataset splits are important for preventing data leakage and for comparing different models trained on the same dataset. Strategies for dataset splitting vary at the discretion of the researcher. This study utilized four levels of dataset splits:

- **Train:** This split should be as large as possible while ensuring that the remaining data is sufficient for reliable validation.
- **Validation:** This dataset is used to intermittently provide an unbiased measure of model performance during training. In some setups, it safeguards against overfitting through early stopping—a mechanism that halts training when performance on the validation set declines.
- **Development:** This dataset is used to evaluate models after training in a hyperparameter tuning study. It is larger and less exposed than the validation set, thus providing a more accurate, unbiased measure of performance.
- **Test:** After hyperparameter tuning is complete, models are finally compared using the test set. While the *development* set informed choices during hyperparameter tuning, the *test* set remained hidden, making it the most unbiased measure of performance.

Table 8.1 Split Ratios for Pre-Training and Fine-Tuning

Split	Masked LM Pre-Training	Task Fine-Tuning
Train	75%	65%
Validation	5%	5%
Development	10%	15%
Test	10%	15%

Pre-training utilized a much larger data pool, and a higher percentage was dedicated to training. When fine-tuning, the published splits were used whenever possible. Otherwise, they were split by ratios above.

8.2: Initiating Models for Domain Adaptation

Pre-training BERT for Japanese Twitter followed the methodology of Barbieri et al. (2022), which continued pre-training XLM-R on a Twitter corpus. However, unlike XLM-T, which inherited the complete vocabulary from XLM-R, the experimental models in this study share only 60% of their vocabulary with Japanese BERT. This distinction is important when initializing the base model. During initialization, common embeddings were migrated, and the remaining ones were randomly initialized. Otherwise, the experimental models began pre-training with the same parameters as the original. Two experimental models were instantiated, each with a unique vocabulary tailored for training on either the deduplicated or balanced corpus.

8.3: Preparing a Training Budget

Pre-training a language model is resource intensive. The subject of this project is a base sized BERT model with 110 million parameters. Pre-training a model of this size calls for days of dedicated processing from a high-end GPU. Compounding the challenge, the success of training depends on the configuration of hyperparameters, referring to the variable elements of preprocessing, model architecture, and learning that influence the training outcome. The relationship between hyperparameters and performance is complex, often requiring multiple full training runs with varying configurations to achieve optimal results. Each additional hyperparameter exponentially increases the problem's complexity. This critical procedure must be carefully planned to stay within budget.

Experienced researchers from Google recently published their perspective on best practices for hyperparameter tuning. They advise starting by defining the compute budget. This project benefited from the high-performance computing cluster (HPCC) operated by the German Research Center for Artificial Intelligence (DFKI). Their support allowed for numerous full training cycles to be run simultaneously without significant queue times. When compute resources are effectively

unlimited, Godbole et al. suggest allocating a generous exploratory budget. This exploration phase focuses on understanding the problem through systematic testing of hyperparameter groups, ultimately leading to a narrow search field of variable parameters. To take full advantage of exploration, pipelines were set up for the storage, visualization, and evaluation of data.

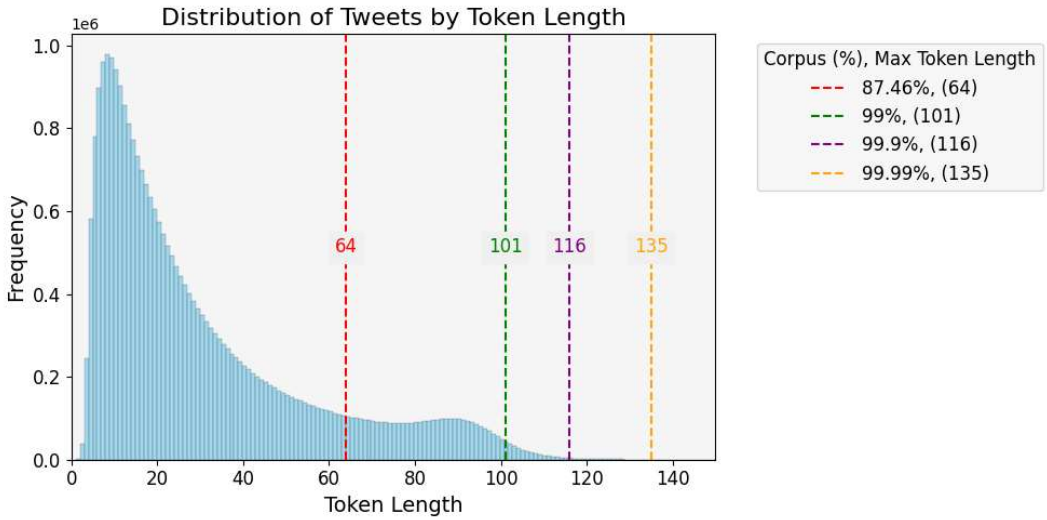
8.4: Exploration of Fixed and Variable Parameters

Exploration began by tailoring the training configuration to the hardware constraints. Two hyperparameters impacting the space complexity of GPU training are the maximum sequence length and the batch size. These represent the maximum tokens per sequence and sequences per training step, respectively. During training, steps are processed simultaneously, requiring the VRAM to accommodate a full batch of maximum-length sequences. Larger batch sizes enhance training speed without affecting the model's potential performance (You et al., 2020). However, batch size strongly interacts with other hyperparameters, particularly the learning rate. It is advised to optimize the batch size early and treat it as fixed (Godbole et al., 2023). Standard practice involves selecting a sequence length that represents the data and paring it with a batch size that fully utilizes the available VRAM. For instance, in training BERTweet, Nguyen et al. (2020) maximized their batch size by aggressively restricting the sequence length to 64. Given the significant data loss from refinement, this study opted for a more inclusive maximum. Figure 8.1 shows a histogram of the Twitter corpus comparing cumulative percentages of varying tokenized lengths. A maximum sequence length of 116 tokens was chosen, covering 99.9% of the corpus, and the minority of overlength sequences were truncated during preprocessing. Afterwards, the batch sized was calibrated to 55, optimally utilizing the 12GB of VRAM of the lowest spec GPU of the HPCC.

The hyperparameter problem is simplified by aligning fundamental design elements of the training setup with established research. Those hyperparameters that define model architecture were fixed to retain knowledge from the original Japanese BERT. This includes dropout, a layer active only during training that prevents overfitting by interfering with overly complex layers (Srivastava et al., 2014). Although dropout is a common variable parameter, all Japanese encoder models on HuggingFace used the same value (Wolf et al., 2020). Pre-training used a learning rate schedule with a warm-up and linear decay, which is a common choice for long training runs. Likewise, it also uses an *AdamW* optimizer. *Adam* is a standard choice for NLP that facilitates quick convergence and superior generalization through adaptive parameter updates (Kingma & Ba, 2017). *AdamW*, incorporates

weight decay, which further supports generalization by penalizing larger weights (Loshchilov & Hutter, 2017).

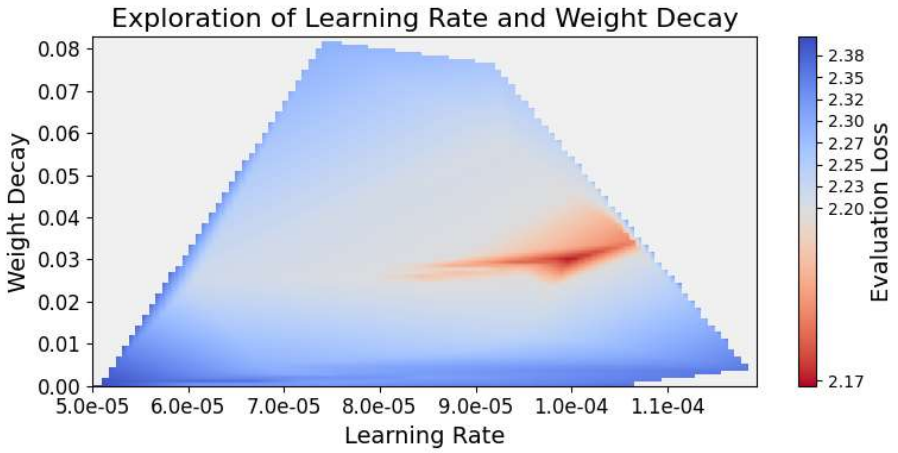
Figure 8.1 Analysis of Token Length Distribution to Determine Max Sequence Length



*A histogram of tokenized tweet lengths was used to select an optimal **max sequence length** of **116.***

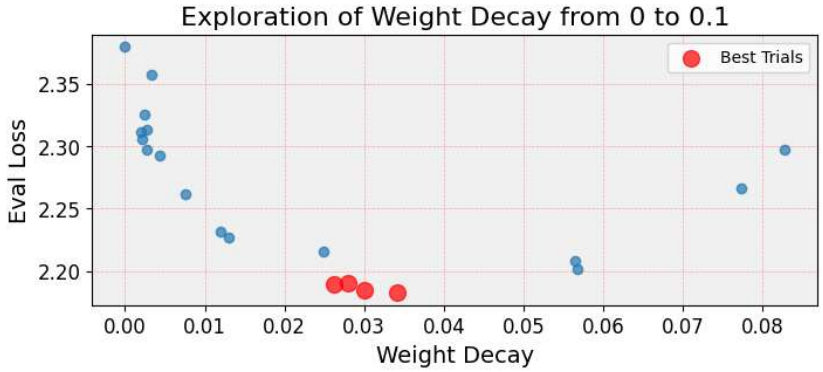
At this stage of the exploration, seven variable hyperparameters were identified. First among them is the choice between the deduplicated and balanced corpora. The learning rate schedule adds two more variables: the learning rate itself and the number of training epochs. Additionally, the optimizer's behavior is influenced by four hyperparameters: beta1, beta2, epsilon, and weight decay. The process of exploitation involves conducting a series of trials with hyperparameter configurations that uniformly cover a variable search space. Covering seven hyperparameters would require hundreds or even thousands of trials, surpassing the available time budget. Consequently, further exploration was directed at narrowing down the list of variable parameters, prioritizing those with the most significant impact (Godbole et al., 2023).

Figure 8.2 Exploration with Variable Learning Rate and Weight Decay



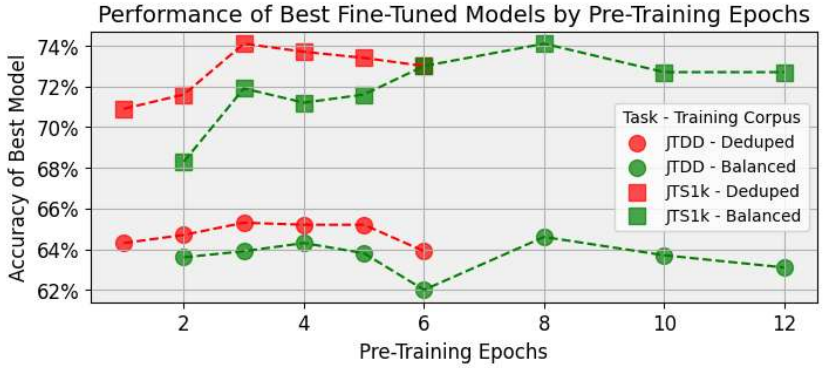
*Initial studies identified the learning rate and weight decay as high impact. This heatmap suggests an optimal **learning rate** of **1e-04** with moderate weight decay.*

Figure 8.3 Exploration with Fixed Learning Rate and Variable Weight Decay



*With the learning rate fixed, the best results are achieved with a **weight decay** of **0.3.***

Figure 8.4 Performance of Models with Varying Pre-Training Epochs



The number of train epochs was also significant. Increasing training time invariably yielded better loss, but evaluation by fine-tuning revealed a more complex relationship. This analysis shows that trainability peaks before declining with more train epochs.

For the remaining hyperparameters, exploration pointed to a narrow range that tended to produce better loss values, but these effects were less consistent and more subtle. Table 8.1 defines the hyperparameter search field.

Table 8.2 Search Field for Hyperparameter Sweep

Model Architecture		Preprocessing	
Model Type	BERT	Batch Size	55
Model Size	base (110M)	Max Sequence Length	116
Activation Function	gelu	Training Corpus	**{28M, 15M}**
Attention Dropout	0.1	**Learning Rate Scheduler**	
Hidden Dropout	0.1	Learning Rate	1e-04
Hidden Size	768	Decay Schedule	Linear
Initializer Range	0.02	Warmup Steps	10k
Intermediate Size	3072	Num Train Epochs	**{1, 6}; {2, 12}**
Layer Norm Epsilon	1e-12	**Optimizer**	
Max Pos Embeddings	512	Type	AdamW
Pos Embedding Type	Absolute	Beta1	**{0.8, 0.9}**
Num Attention Heads	12	Beta2	**{0.999, 0.9999}**
Num Hidden Layers	12	Epsilon	**{1e-08, 1e-05}**
Vocab Size	32k	Weight Decay	0.03

The search space for the hyperparameter sweep was encapsulated by the values highlighted in bold. The training corpora, Deduplicated (28M) and Balanced (15M), are defined by their size. The number of training epochs was adjusted according to the corpus size. The exploration examined every nth training epoch within the specified range. For beta2 and epsilon, only the default and optimized values were explored. The search for beta1 was more thorough, with greater focus on areas with optimized training epochs.

8.5: Evaluation of Candidate Models by Fine-Tuning

Godbole et al. emphasize that downstream task training is the most reliable evaluation method. This technique measures the quality of pre-trained parent model by the performance of its fine-tuned children. This study evaluates with two datasets:

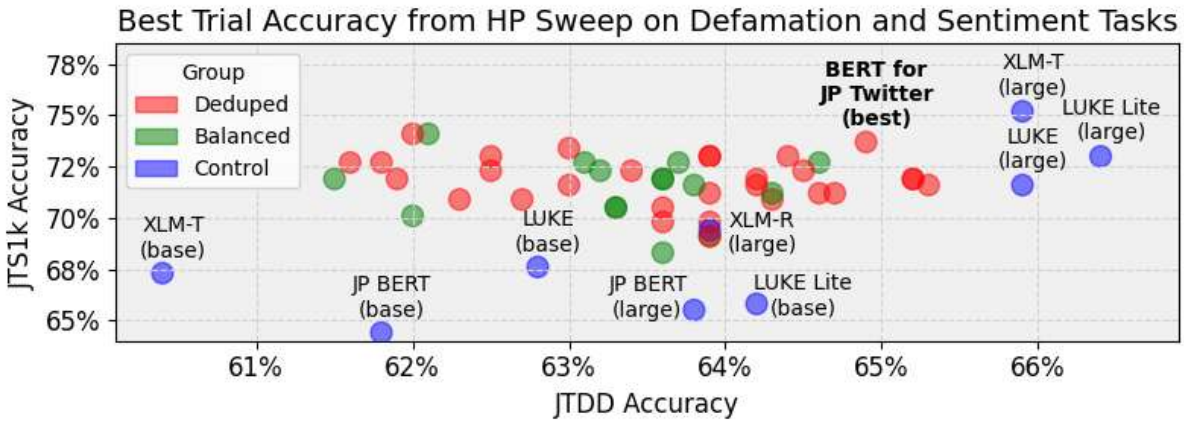
- **JTS1k:** The target sentiment analysis dataset.
- **JTDD:** For defamation detection on Japanese Twitter. This is a larger dataset and a more challenging classification task.

Fine-tuning entails another layer of hyperparameter optimization. Each evaluation involved a hundred trial hyperparameter sweep guided by a Bayesian search algorithm. Much of this process was automated through some useful python libraries, which is explained in more detail in Appendix

5. This evaluation compares candidate models against the encoder models that are listed in the literature review.

- **Japanese BERT:** Provide the lower bound baseline. The candidate models must outperform the baes model on both tasks. Ideally, they will outperform the large version too.
- **LUKE:** Provides an upper bound baseline for Japanese models. In addition, comparing the full and lite versions will provide some insight on the effectiveness of its entity embeddings.
- **XLM-T:** Provides an upper bound baseline for Twitter tasks. The evaluation also includes the large version of XLM-R. XLM-T should outperform.

Figure 8.5 Comparison of Candidate Models on JTS1k and JTDD



Every dot represents a pre-trained model fine-tuned on JTS1k and JTDD. Accuracy reflects the maximum accuracy across a hundred trial hyperparameter sweep. These fifty trials minimally outperformed the original Japanese BERT on both tasks.

Table 8.3 Comparison of Candidate Models on JTS1k and JTDD

Model	Version	JTDD	JTS1k
BERT for Japanese Twitter	**Deduped**	64.9%	73.7%
	Balanced	62.1%	74.1%
Japnese BERT	large	63.8%	65.5%
	base	61.8%	64.4%
LUKE	large	65.9%	71.6%
	base	62.8%	67.6%
LUKE Lite	large	**66.4%**	73.0%
	base	64.2%	65.8%
XLM-R	large	63.9%	69.4%
XLM-T	large	65.9%	**75.2%**
	base	60.4%	67.3%

Shows the best values from both corpora against the control.

Compared to the control models, the Twitter models are the smallest when measured by the number of parameters. The nearest neighbor is the base LUKE lite, which has 20% more parameters. The large BERT is triple the size, and the large LUKE and XLM models are five times the size. Despite the parameter disparity, the best Twitter model outperformed most of the controls on both tasks. All the experimental models exceeded even the large BERT on sentiment analysis. Defamation detection was more challenging, eliciting the best performance from the Japanese specialized LUKE models. Therefore, JTDD served as an ideal countermeasure against overfitting to Twitter. The large XLM-T model outperforms on both tasks, but the Twitter model exceeds the base version of XLM-T. Overall, this study validates the success of Twitter adaptation and the effectiveness of the Twitter model. The best model has been published on HuggingFace as BERT for Japanese Twitter. Going forward, this is the model used for exploration and fine-tuning.

8.6: Evaluation of BERT for Japanese Twitter

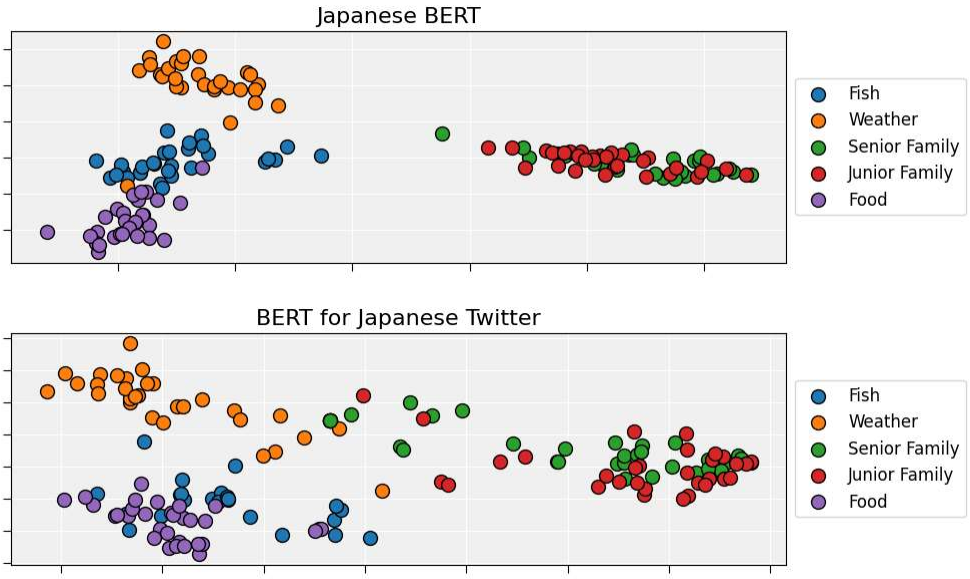
Evaluation of masked language models is not straightforward. During the hyperparameter sweep, candidate models were fine-tuned on a pair of tasks within the Japanese Twitter domain that varied subtly in their domain alignment. The next analysis thoroughly explores the chosen model using a series of techniques. The embedding matrices of the original and Twitter-adapted BERT are compared to probe the extent to which semantic networking is maintained. A t-SNE analysis visualizes the embedding clusters, revealing the underlying structure and semantic relationships between different words and characters, including non-standard characters like emojis. Next, a pair of tasks is used to evaluate the quality of masked token predictions. Accuracy is measured with a large-scale unsupervised task, and acceptability is measured with a task involving human annotations. Finally, the model is fine-tuned on a series of tasks that align either with the Twitter domain or the general domain. The model is expected to show enhanced performance on Twitter tasks and reduced but acceptable performance on general tasks.

t-Distributed Stochastic Neighbor Embedding (t-SNE)

t-SNE is a dimensionality reduction technique particularly well-suited for the visualization of high-dimensional data (van der Maaten & Hinton, 2008). This technique is useful for visualizing the embedding matrix in an understandable way because it can reveal the underlying structure of the data, showing how different words and characters are grouped based on their semantic relationships. If pre-training was effective, the semantic networks from the original BERT should be maintained. In addition, new elements, like emoji, should cluster meaningfully in embedding space. t-SNE was used

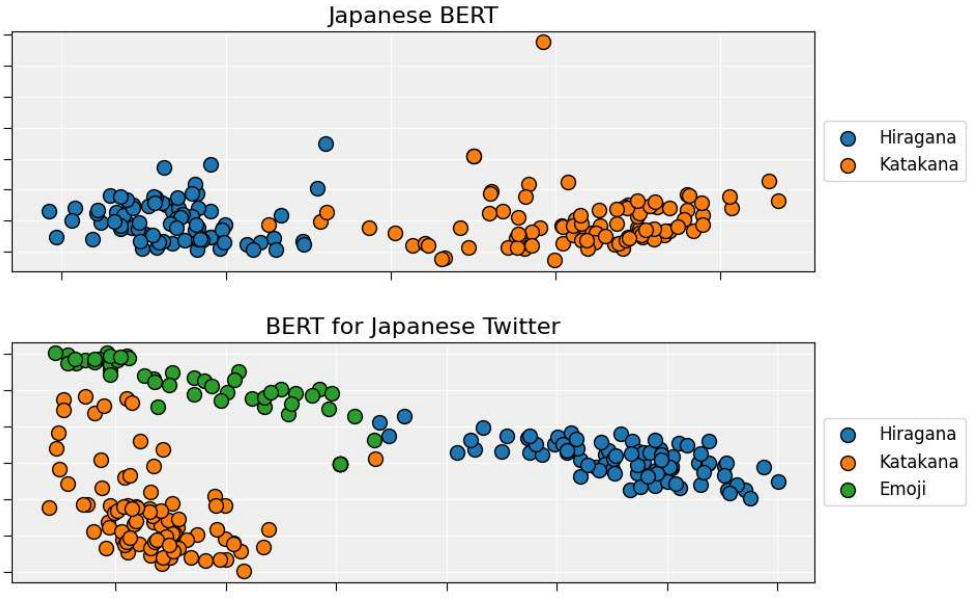
to plot CLS embeddings of selected words and characters, and the results are shown in Figures 8.7 – 8.9.

Figure 8.6 t-SNE Analysis of Common Japanese Nouns



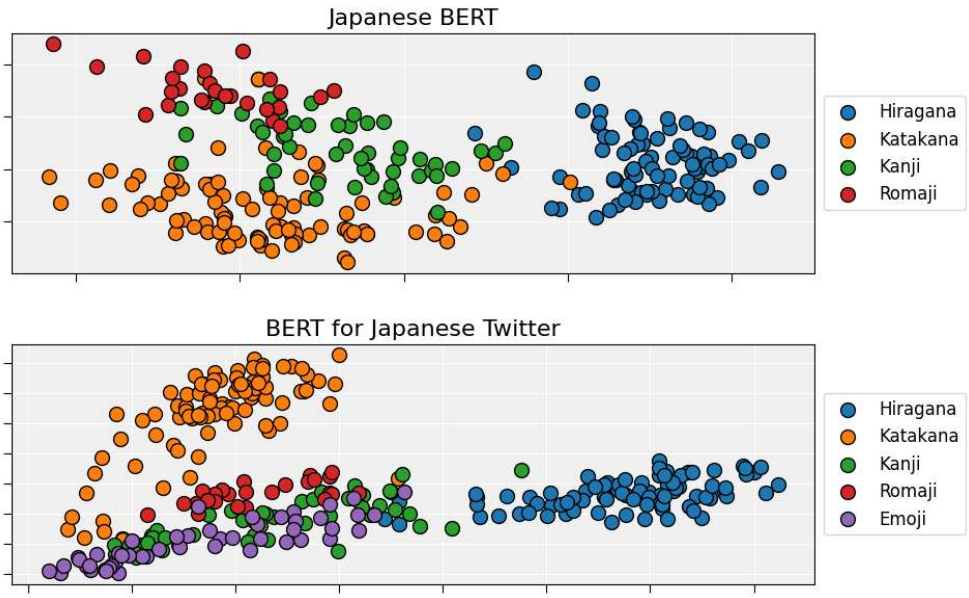
Plots 20 examples from five classes of common nouns. In the original, distinct classes cluster tightly. There seems to be no difference in junior or senior family terms, and fish clusters close to food, which is appropriate. In the Twitter model, clusters are maintained, but they are less distinct. This suggests that adaptation introduced some noise, but the overall structure is intact.

Figure 8.7 t-SNE Analysis of Basic Japanese Characters



Plots the full vocabulary of hiragana and katakana, which cluster separately. The next plot adds fifty randomly selected emoji, which emerges as a distinct class.

Figure 8.8 t-SNE Analysis of Extended Japanese Characters



The first plots the full vocabulary of hiragana, katakana, and lower case romaji with fifty randomly selected kanji. Hiragana clusters separately, but the other three families mingle. The next plot adds fifty randomly selected emoji. Hiragana maintains its distance and katakana branches of into its own area of embedding space. Emoji cohabits with romaji and kanji.

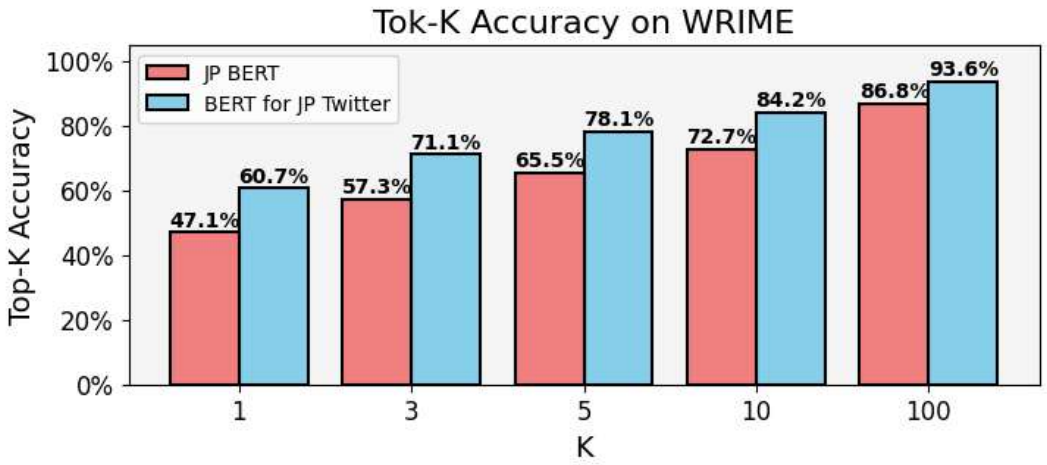
Quality of Masked Token Predictions

Masked token prediction is challenging to evaluate because traditional metrics like accuracy are not applicable. During training, performance is measured using cross-entropy loss, an indirect metric of the difference between true and predicted tokens. While loss is effective for guiding training, it lacks explainability. To address this, the project assessed the Twitter model through tasks focusing on the quality of token prediction rankings. At inference, a language model predicts every token in its vocabulary, with K defining the token rank. The first test measures Top-K accuracy; when a model is given a sequence with a masked token, top-K accuracy indicates how often the true token is among the top K predictions. The second test measures acceptability as a function of K, where human annotators evaluate the naturalness and relevance of token predictions at varying K values.

The analysis of top-K accuracy compares the predictions by the original and Twitter adapted BERT models on the WRIME dataset, which was sourced from social media (Suzuki et al., 2022). WRIME was chosen because it is large and it aligns with the target domain, but its character set is relatively clean, which reduces bias against the original BERT model. A single token from each of the 30,000 examples was masked. To safeguard against out-of-vocabulary masking, only whole word

tokens from the shared vocabulary were masked. Figure 8.10 plots the top-K accuracy earned by both models with K values ranging from 1 to 100.

Figure 8.9 Top-K Accuracy of Masked Token Predictions on WRIME



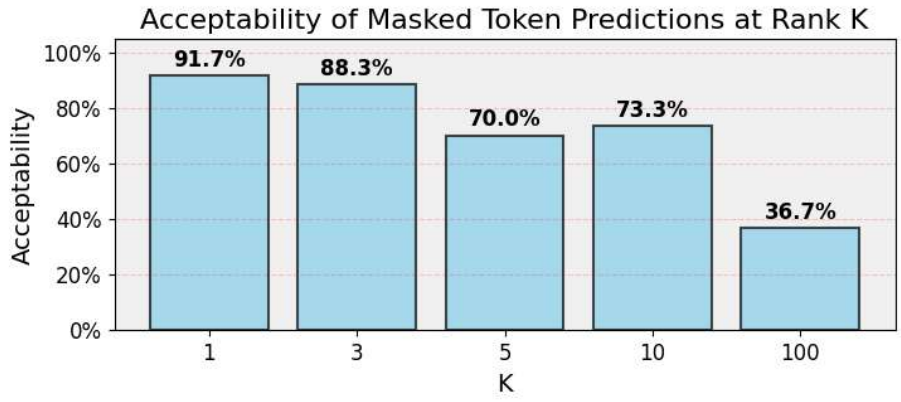
The Twitter model surpasses the original when evaluated by top-K accuracy of token predictions on WIME. Surprisingly, the top-1 accuracy indicates that the Twitter model perfectly predicted the True token in over 60% of examples. This level of performance was not expected, and it raises questions about the linguistic diversity of social media. A more informative study would consider masked token predictions on a more general corpus. Nevertheless, these results a considered a positive signal.

The results from the top-K accuracy tests were promising. However, this methodology has limitations. Token prediction, being a generative task, is better analysed by the acceptability of predictions over a range of K values. Quality is best assessed with a human-in-the-loop approach. Due to the resource-intensive nature of human annotation, this study focused solely on the Twitter model. The two native Japanese that participated in developing JTS1k were enlisted to appraise masked token prediction.

Building the annotation set involved selecting appropriate examples and substituting tokens with predictions. The study targeted average length sequences ranging from 20 to 40 tokens, creating a selection of 250 examples. The dataset was divided into five groups representing varying K values. From each sequence, a single token was substituted with the token at the group K value. That token was highlighted when presented to the annotators, who were instructed to label the prediction as either acceptable or unacceptable. For evaluating the acceptability of Twitter data, more grammatical flexibility was permitted compared to other acceptability datasets (Warstadt et al, 2018). Annotators were instructed to focus on the highlighted token and to accept marked grammar if it appeared

natural. Unacceptable tokens were identified as those that were either unnatural or irrelevant.[13] The K values within this study ranged from 1 to 100. The expectations were that acceptability would remain high within lower K values and drop close to zero at 100.

Figure 8.10 Acceptability of Masked Token Predictions at Varying K Values



K	Annotator 1	Annotator 2	Mean
1	93.3%	90.0%	**91.7%**
3	90.0%	86.7%	**88.3%**
5	66.7%	73.3%	**70.0%**
10	70.0%	76.7%	**73.3%**
100	43.3%	30.0%	**36.7%**

Reliability	
Unique	250
Shared	50
Agreement	76%
Kappa	**0.38**

The appraisals were consistent with expectations, showing that acceptability is inversely related to K. The case-by-case agreement was moderate, with a kappa of 0.38, but the distribution of acceptability scores between annotators was consistent. The acceptability at rank 100 was higher than expected. It would be worthwhile to repeat this experiment with other corpora and models to better understand the relationship between acceptability and K ranking.

Performance within the Social Media Domain

Compatibility with social media was assessed with tasks connected to sentiment analysis, market research, and content moderation:

- **JTS1k & JTDD** were used to select the top performing model.
- **JTBR** is large sentiment analysis dataset for Japanese Twitter that specifically targets brand perceptions (Keshi, et al., 2017). The tweets are related to a narrow selection of topics, mostly mobile brands. Annotators were instructed to classify the user's opinion towards a specific brand. Therefore, this dataset better aligns with aspect-based sentiment analysis.

[13] The instructions and a translation are provided in Appendix 6.

- **WRIME** is a selection of social media posts labeled by emotional intensity and sentiment polarity (Kurihara et al., 2021; Suzuki et al., 2022). Emotion and sentiment labels are used to train separate models with unique task configurations. Emotion is treated as a multi-label classification problem and is evaluated by Top-1 accuracy and F1 score, reflecting the model's ability to detect the strongest emotion and all emotions that exceed a threshold. Sentiment is treated as a single-label regression problem where the model is trained to produce a score representing sentiment polarity and intensity. The quality of predictions is measured by the correlation between true and predicted values.

JTS1k, JTBR and WRIME approach sentiment analysis from different directions. While both Collectively, these datasets not only test model performance across different contexts but also facilitate the exploration of alternate sentiment analysis strategies. This experiment only compares the original to the Twitter model, and the expectation was that the Twitter model would outperform on every task.

Table 8.4 Performance of Models on Social Media Tasks

	JTS1k Acc	JTDD Acc	JTBR Acc	WRIME (Emo) Top-1 / F1	WRIME (Sent) Pear / Spear
BERT for Japanese Twitter	**0.737**	**0.649**	**0.864**	**0.694 / 0.639**	**0.865 / 0.868**
Japanese BERT	0.644	0.618	0.855	0.672 / 0.612	0.842 / 0.844

Compares the Twitter model to the original, with the best scores highlighted in bold.

The Twitter model surpassed the original BERT on all tasks. Better performance on WRIME is significant for several reasons. First, WRIME has a cleaner character that is more compatible with the original BERT. For instance, 2.5% of the JTS1k tokens are out-of-vocabulary for the original BERT, compared to 0.1% from WRIME. The Twitter model's superior performance signals the acquisition of general linguistic features beyond a more inclusive vocabulary. Furthermore, WRIME is an exceptional dataset that closely aligns with the aims of this project. It is substantial in size, balanced by emotional content, reliably annotated, and broadly representative of the Japanese social networking domain. Proficiency with WRIME enhances the Twitter model's value proposition by suggesting compatibility with alternative social networking services (SNS). Other advantages of WRIME are its accessibility and longevity. In compliance with Twitter policy, the three Twitter datasets only share IDs, requiring interface with the Twitter API to acquire text. Over time, Twitter data deteriorates due to deleted

posts and policy changes. Over 80% of the JTBR and 20% of the recently published JTDD is inaccessible, with similar loss expected for the JTS1k. In constructing WRIME, corpus developers secured the consent of its contributing authors[14] which allows for distribution without the attrition and complexity associated with the Twitter API. Superior performance on a high-quality and accessible dataset like WRIME may entice future researchers to build applications with the Japanese Twitter model.

Performance across the General Domain

One of the challenges of domain adaptation is the risk of catastrophic forgetting (French, 1999). This phenomenon is particularly evident during sequential training, where knowledge is transferred across different phases of adaptation. In Chapter 8, candidate models were evaluated on a pair of tasks that varied in domain alignment. Successful performance on both signaled that the model had acquired Twitter competency while retaining general linguistic proficiency. The best model was further scrutinized using the **J**apanese **G**eneral **L**anguage **U**nderstanding **E**valuation (**JGLUE**) (Kurihara et al., 2022). JGLUE is a benchmarking collection that includes six datasets:

- **MARC-ja** is a binary sentiment analysis dataset that classifies Amazon reviews as positive or negative based on star rating.
- **JCoLA** (Someya et al., 2024) classifies examples extracted from linguistic texts as either acceptable or unacceptable.
- **JNLI** evaluates models on their ability to discern logical relationships between a pair of sentences, classified as entailment, contradiction, or neutral.
- **JSTS** focus on the semantic similarity between pairs of sentences rated on a continuous scale. Predictions are evaluated with correlation values.
- **JCSQA** tests commonsense reasoning through multiple choice questions.
- **JSQuAD** tests reading comprehension using Japanese Wikipedia, where given a question and a passage, models are trained to extract the answer from the passage. Responses are evaluated by exact match (EM) and F1 score, reflecting the frequency with which the model chose a perfect answer and the accuracy and precision of responses on a token level.

Work with JGLUE began by replicating published scores with the original Japanese BERT. Confirming the reliability of the procedure, peak performance scores aligned closely with published

[14] Twitter policy enforces ownership over its data, regardless of user consent. In building WRIME, the contributors manually retrieved their past SNS posts for annotation. Twitter, nor any other micro-blogging platform, is explicitly named as a source of data (Kajiwara et al., 2021; Suzuki et al., 2022). Their dataset publishes the full text, and so far, social media platforms have not raised any concerns.

values, and benchmarking continued with the Twitter model. Table 8.3. records the performance of the best fine-tuned models alongside published scores. The expectation was the Twitter model would underperform on these tests, although there was hope that it would perform better on MARC-ja.

Table 8.5 Performance of Models on General Tasks

Model	MARC-ja Acc	JSTS Pear / Spear	JCoLA Acc	JNLI Acc	JSQuAD EM / F1	JCSQA Acc
Human	**0.989**	0.899 / 0.861	0.760	**0.925**	0.871 / 0.944	**0.986**
BERT for Japanese Twitter	0.959	0.894 / 0.850	0.865	0.870	0.831 / 0.892	0.745
Japanese BERT (v3)	0.967	0.917 / 0.878	**0.872**	0.909	**0.904 / 0.946**	0.854
Japanese BERT (v1)	0.958	0.909 / 0.868	0.838	0.899	0.871 / 0.941	0.808
XLM-R (large)	0.964	**0.918 / 0.884**	0.831	0.919	---	0.840
XLM-R (base)	0.961	0.877 / 0.831	0.827	0.893	---	0.687

Compares the performance scores of the Twitter models with published values[15]

The original Japanese BERT outperformed the Twitter model across all datasets, and it trained robustly on a wider range of hyperparameters. The Twitter model was least effective with the encyclopaedic JSQuAD and JCommonsenseQA datasets, suggesting distance from the Wikipedia-based training of the original model. For the remaining datasets, which are more informal, the performance gap was smaller, yet still significant. Excluding MARC-ja and JCoLA, the Twitter model lagged behind even the first version of Japanese BERT. Nevertheless, it surpassed the base version of XLM-R in most tasks. This experiment confirmed that some general aptitude had been lost to Twitter adaptation, but the loss was not catastrophic.

8.7: Interim Conclusion

This chapter concluded the comparison between the Deduplicated and the Balanced training corpora. Initially, it was expected that the Balanced corpus would feature greater linguistic diversity and reduced business interest. It was also anticipated to train more efficiently due to its smaller size, providing more opportunities for experimentation and potentially leading to a better-performing model. However, analysis of the corpus content revealed no change in linguistic diversity and an

[15] Initial Evaluations: https://github.com/yahoojapan/JGLUE/tree/main
Updated Evaluations: https://github.com/cl-tohoku/bert-japanese

increased representation of business interests. Although training epochs completed in half the time, the model required twice as many epochs to reach optimal results. Consequently, the range of training epochs explored doubled, necessitating more pre-training runs to cover the search space. Only a quarter of the models pre-trained on the Balanced corpus surpassed Japanese BERT on JTDD. In contrast, half of the models pre-trained on the Deduplicated corpus achieved this benchmark. Overall, the Deduplicated corpus yielded better results, proving superior to the Balanced corpus by every measure.

The critical mass of training corpora is an important consideration for research. If equivalent results can be achieved with a smaller corpus, it extends the accessibility of model development. It is significant that the best models pre-trained on the Balanced corpus were nearly equivalent to those trained on the Deduplicated corpus. Given a smaller, strategically sampled corpus with superior linguistic diversity, there is potential to achieve even better results.

Chapter 9: Exploring Opportunities for Transfer Learning

Although BERT for Japanese Twitter fine-tunes well on JTS1k, it underperforms when compared to the state-of-the-art OpenAI models. This was addressed with a more sophisticated fine-tuning approach that leverages transfer learning. The primary limitation of JTS1k is its size. With fewer examples, it is challenging to achieve representativeness. Furthermore, testing is volatile, with just a few examples changing results by whole percentage points. Combining JTS1k with either the WRIME or JTBR datasets may enhance results. However, this is complex, as JTBR differs in its analytical scope and WRIME uses a different labeling system. To assess the feasibility of combining datasets, a cross-task transfer study was conducted between the Japanese social media datasets.

9.1: Datasets for Sentiment Analysis of Japanese Social Media

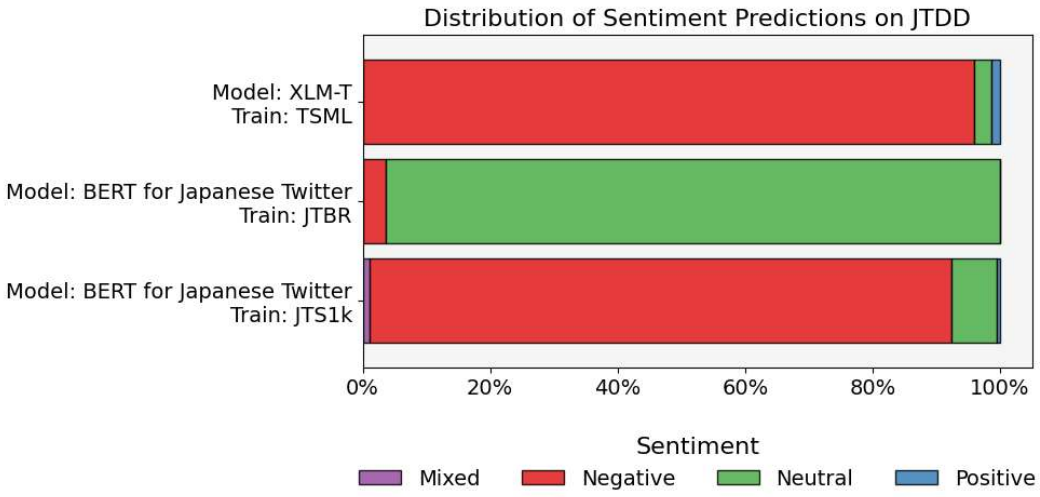
- **JTS1k** is the target dataset.
- **WRIME** includes 30,000 SNS posts with continuous polarity scores ranging from -2 to 2.
- **JTBR** uses the same labels as JTS1k. It better aligns with aspect-based sentiment analysis.

BERT for Japanese Twitter was fine-tuned on each of these datasets. The analysis also considers predictions from the state-of-the-art XLM-T sentiment model that was fine-tuned on TSML. This study is looking for evidence of positive transfer, where models perform well outside of their training context. Positive transfer between datasets is a strong indicator of robustness. Furthermore, if two-way transfer learning is observed, there may be an opportunity to pursue a combined training approach. This could enhance model performance by leveraging the strengths of multiple datasets, leading to more generalizable and effective models.

9.2: Exploration of Cross-Task Transfer

The first experiment concentrated on JTDD. The language in these defamatory tweets is slanderous and should prompt a nearly universal negative classification. However, this dataset comprises language that is subject to content moderation, introducing the possibility that these offensive expressions may not be adequately represented in the training material. This study evaluates the generality of the sentiment models by analyzing their predictions on this tangentially related dataset. If the models are broadly applicable, they should yield predominantly negative classifications. Additionally, this analysis seeks examples that are classified as positive despite being defamatory. Identifying such misclassifications can reveal patterns of language usage that are challenging for models to interpret accurately.

Figure 9.1 Sentiment Analysis of Defamatory Tweets



XLM-T Sentiment and the JTS1k models assign most of the dataset negative labels, but the JTBR model prefers the neutral category.

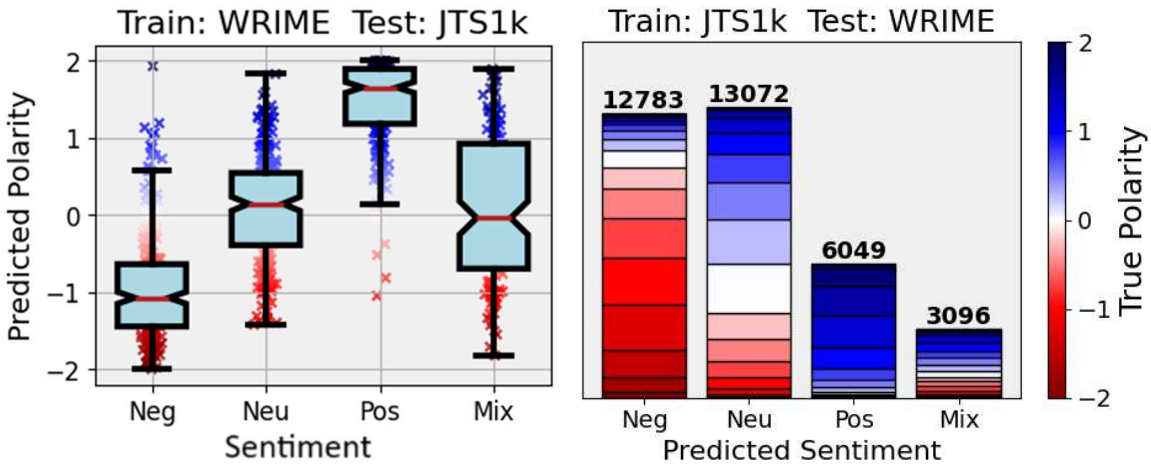
This behavior observed in Figure 9.1 was consistent across datasets. When the JTBR model made predictions on the JTS1k and WRIME datasets, it continued to assign over 90% of the examples neutral labels. On the other hand, the XLM-T and JTS1k models demonstrated some competence with JTBR. This limited one-way compatibility implies that general sentiment encompasses some features of the brand-focused JTBR. However, JTBR is too narrow to be applied generally. Consequently, JTBR was excluded from further consideration.

Figure 9.2 Example of a Hateful Tweet with Positive Language

This is one of the few tweets that was marked as positive by the other models. Annotators of JTDD recognized the nationalistic undertones of this tweet.

The remainder of this exploration assessed the compatibility of JTS1k with WRIME. Comparing these two datasets is more involved because they use different labelling systems.

Figure 9.3 Transfer between JTS1k and WRIME



Box-and-whisker plots were used to represent the predictions of the WRIME model on JTS1k (left). Stacked bar charts were used to represent the predictions of the JTS1k models on WRIME, where the shading of the bar represents the polarity (right).

Figure 9.3 suggests compatibility between the JTS1k and WRIME datasets. Focusing on the WRIME model predictions on JTS1k, the median values of the positive and negative labels are strongly polar, while the neutral and mixed labels are close to zero. The interquartile range for negative, neutral, and positive labels holds tightly in the appropriate polar zone. The full range of positive predictions, not including outliers, remains above 0. The range of negative labels spans into the neutral and weakly positive zones. Mixed labels span the entire polar zone, showing no relationship with polarity.

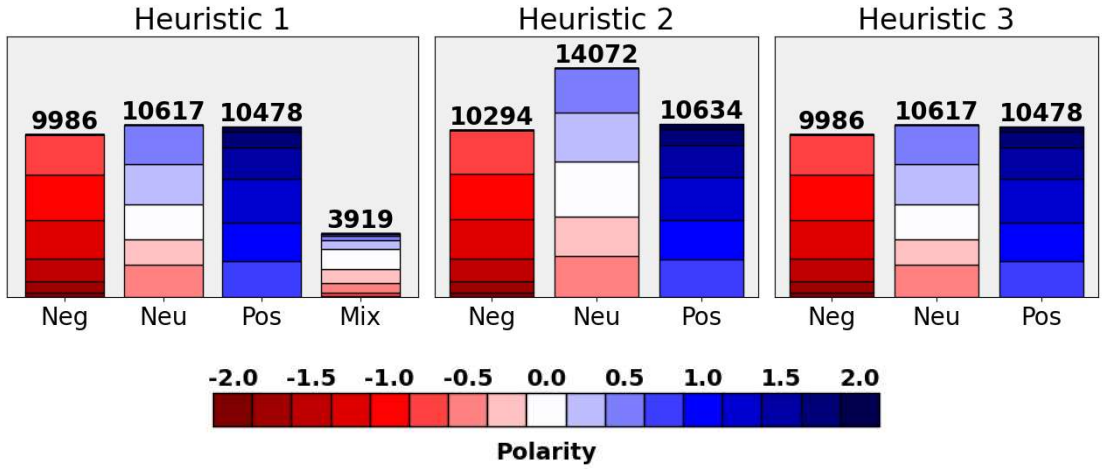
For the JTS1k model predictions on WRIME, strongly polar examples are generally classified appropriately. Examples in the weakly polar zone from -0.5 to 0.5 are classified as neutral with high recall but low precision. The positive and negative categories are distinct, but there is more confusion between the negative and neutral categories. Mixed predictions, while less frequent, are distributed evenly across the polar range. These findings align with previous experiments, showing a distinct separation between positive and negative sentiments, recurring confusion between negative and neutral sentiments, and the consistent challenge presented by the mixed category.

9.3: Converting WRIME to a Categorically Labelled Dataset

Combining tasks is a complex practice with various approaches (Ruder, 2017). This project employed a straightforward method by converting WRIME polarity scores to discrete labels and combining datasets. Polarity is determined by the average score assigned by four annotators, and sentiment labels were assigned based on that score. In most examples, annotators agreed on the

polarity, with scores typically ranging from 0 to one polar direction or the other. However, in 11% of cases, annotators disagreed, resulting in a range of annotations representing both polarities. An experiment was conducted testing three heuristics for handling these cases of disagreement.

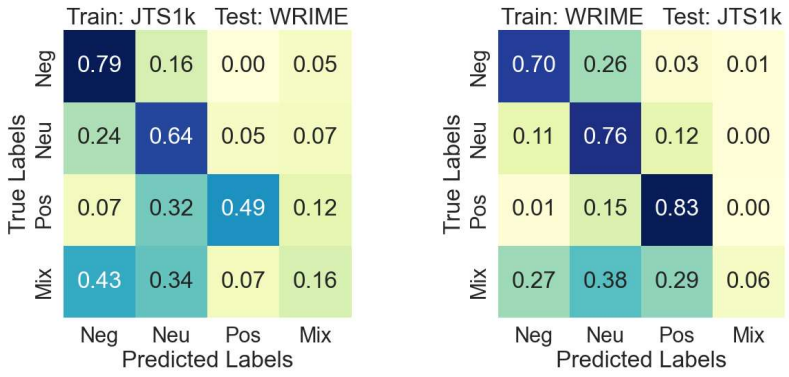
Figure 9.4 Methods for Converting WRIME to Categorical Labels



Polarity scroes range from -2 to 2 with a step of 0.25. Examples within -0.5 to 0.5 are Neutral, above are Positive, and below are Negative. Three heuristics handle cases of polar disagreement differently. Heuristic 1 assigns these cases the Mixed label, and the other two forgo the Mixed category. Heuristic 2 ignores disagreement, labelling all examples by their average score. Heuristic 3 excludes examples with diagreement.

The heuristics were tested by labelling WRIME accordingly, fine-tuning the Twitter model, and testing on JTS1k. The optimal method for labelling will evaluate the best. These experiments demonstrated that the mixed category does not align well with a polar approach to sentiment analysis. Heuristic 1 attempted to maintain the mixed label, but with poor results.

Figure 9.5 Evaluation of Heuristic 1



The models fine-tuned on JTS1k and the relabeled WRIME transferred well with the three main sentiment labels, but failed to generalize the mixed category.

To take full advantage of WRIME, it was necessary to sacrifice the mixed label. Moving forward, sentiment analysis was treated as a three-label problem. It is unfortunate to overlook the mixed examples, which constitute 15% of an already small dataset. On the positive side, this provided for a more fair and favorable comparison with XLM-T. Within a classification task, performance scores naturally increase when reducing the number of labels. By simplifying the problem, it was expected that BERT for Japanese Twitter would significantly outperform the XLM-T sentiment model. Table 9.1 documents the performance of Twitter models fine-tuned on the reconfigured datasets against XLM-T sentiment.

Table 9.1 Positive Transfer from WRIME to JTS1k

Model	Train		Test: JTS1k Precision	Recall	F1
BERT for JP Twitter	JTS1k		**0.802**	**0.798**	**0.799**
	WRIME	Heuristic 2	0.771	0.731	0.73
		Heuristic 3	0.792	0.753	0.758
XLM-T	TSML		0.756	0.754	0.748

With the problem reduced to three labels, the Twitter model fine-tuned on JTS1k gains a significant lead. WRIME models evaluate well on JTS1k, but Heuristic 3 yielded a superior dataset, even exceeding XLM-T sentiment

WRIME and JTS1k have demonstrated strong compatibility, indicating the potential for effective transfer learning between these datasets. To train a higher quality, more general sentiment classifier, it was decided to combine the negative, neutral, and positive examples from JTS1k with the most reliable examples from WRIME. This approach aims to leverage the strengths of both datasets, enhancing the classifier's ability to generalize across different contexts. By integrating the well-defined examples from JTS1k with the robust examples from WRIME, the resulting model is expected to achieve improved performance and reliability in sentiment analysis tasks. This combined training strategy is anticipated to produce a more versatile and accurate sentiment classifier.

9.4: Interim Conclusion

This chapter justifies the decision to combine the WRIME and JTS1k datasets. It involved reducing the number of sentiment labels, which fundamentally changes the problem. Therefore, this decision entailed another round of benchmarking. Chapter 10 will conclude this exploration by applying a combined approach with the experimental and control models. By simplifying the problem, it was hoped that BERT for Japanese Twitter would outperform top-tier models.

The mixed sentiment class is not a commonly used category and introduces complexity that makes it less compatible with a strictly polar approach to sentiment analysis. The study of balanced opinion and conflicting emotions is an important area of research for NLP, but these areas might be more properly addressed with aspect-based sentiment analysis. It seems that a simpler approach is more appropriate for sentiment analysis of Twitter posts. With hindsight, this project would have implemented the three-label scheme from the outset.

The labelling scheme used by WRIME is advantageous due to its flexibility. This experiment demonstrated that it is possible to reliably convert continuous polarity scores into categorical labels. However, it is not feasible to convert the categorically labelled JTS1k to polarity scores. While the WRIME dataset offers enhanced functionality, it imposes more work on the annotators, asking them to consider both the polarity and the intensity of sentiments. WRIME contains 35,000 examples. Unlike the crowdworker approach, Suzuki et al. (2022) recruited three experts to annotate the entire dataset. The expert approach is advantageous because it allows for more opportunity to build rapport and enforce consistency sin annotations. However, such a large workload raises concerns about the threat of burnout among the annotators.

We are entering an age where participation in simple tasks, such as dataset annotation, has monetary value. The demand for reliably annotated and ethically sourced data presents an emerging opportunity for start-ups. Reviewing other publications that pay for crowdsourced data, it is uncommon to see exactly how much was spent on annotation. This study cost €220, and it would be interesting to see how much was spent on similar datasets. Understanding the financial aspects of dataset annotation could provide valuable insights for future research and development in the field of NLP.

Chapter 10: Fine-Tuning BERT for Japanese SNS Sentiment

This chapter details the fine-tuning of the target model, BERT for Japanese SNS Sentiment, using a transfer learning strategy. In Chapter 9, it was observed that JTS1k and WRIME datasets exhibit good transferability with each other. This final experiment will employ a combined training approach. Multi-task training is a complex practice with various potential outcomes (Ruder, 2017). This study aimed for one of three outcomes:

- **Best Outcome:** Models fine-tuned on both JTS1k and WRIME will perform better on both datasets, indicating strong positive transfer across the board.
- **Favorable Outcome:** Models fine-tuned on both will perform better on one dataset and near optimally on the other, which is also a strong indicator of positive transfer.
- **Acceptable Outcome:** Models fine-tuned on both will perform near optimally on both datasets and significantly better than models fine-tuned on one and evaluated on the other. This scenario still supports positive transfer but involves a compromise between performance and generality.

10.1: Experimental Setup

The final round of benchmarking compared BERT for Japanese Twitter with the other top performing models from previous experiments:

- **Japanese BERT**: Provides the lower bound baseline.
- **LUKE lite**: Outperformed LUKE in Chapter 8. This model provides an upper bound baseline for general Japanese.
- **XLM-T:** Provides an upper bound baseline for Twitter tasks.
- **GPT-4o**: OpenAI's latest release, represents state-of-the-art GenAI.

Chapter 9 explained how the target datasets were modified so that they could be combined: JTS1k by dropping the mixed examples, and WRIME by converting the labels from continuous to categorical. Each encoder model was fine-tuned on each dataset individually as well as with a combined approach. The fine-tuned models were evaluated using the test splits from JTS1k and WRIME. GPT-4o was benchmarked using the few-shot learning approach implemented in Chapter 6.

WRIME and JTS1k were combined by simply concatenating and shuffling each split. In multi-task training setup, balance is an important consideration. Considering that WRIME is much larger, the possibility of augmenting the representation of JTS1k is worth exploring. This study adopted the

simple approach, accepting that WRIME would be the dominant influence on training. Table 10.1 and Figure 10.1 document the results of this final benchmark study.

10.2: Results

Table 10.1 Final Benchmark on Sentiment Analysis

Model	Version	Train	Test (F1) JTS1k	Test (F1) WRIME
BERT for Japanese Twitter		JTS1k + WRIME	0.787	0.757
		JTS1k	0.799	0.669
		WRIME	0.758	0.774
GPT-4o			0.767	0.758
Japanese BERT	large	JTS1k + WRIME	0.718	0.770
		JTS1k	0.743	0.605
		WRIME	0.736	0.764
	base	JTS1k + WRIME	0.735	0.749
		JTS1k	0.734	0.646
		WRIME	0.717	0.748
LUKE lite	large	JTS1k + WRIME	0.750	0.792
		JTS1k	0.801	0.708
		WRIME	0.779	**0.792**
	base	JTS1k + WRIME	0.737	0.775
		JTS1k	0.757	0.673
		WRIME	0.739	0.778
XLM-T	large	JTS1k + WRIME	0.774	0.779
		JTS1k	**0.816**	0.688
		WRIME	0.757	0.780
	base	TSML	0.729	0.663
		JTS1k + WRIME	0.750	0.744
		JTS1k	0.779	0.675
		WRIME	0.770	0.743

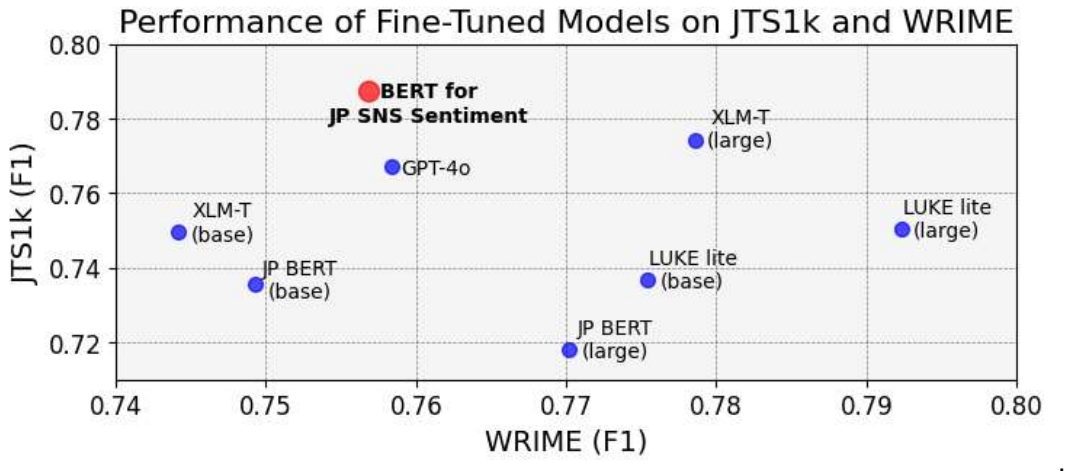
Compares the performance of models on the JTS1k and WRIME sentiment tasks. The best values are highlighted in bold.

Combining the datasets resulted in the 'Acceptable Outcome' for most models, meaning that the best results were attained by models fine-tuned specifically for that dataset. Models fine-tuned on WRIME transferred better to JTS1k than the opposite direction. Notably, the base version of Japanese BERT returned the ‘Best Outcome’, earning slightly higher scores on both tests with the combined training approach. The large LUKE model returned a less favourable outcome, outperforming on JTS1k with the WRIME model over the combined approach. This round of

benchmarking partially supported positive transfer between WRIME and JTS1k. The simple approach for combining datasets was somewhat effective. Further exploration and more sophisticated techniques might yield better optimized models.

10.3: Interim Conclusion

Figure 10.1 Final Benchmark on Sentiment Analysis



Plots performance from models fine-tuned with the combined approach, which achieved near optimal scores on both tasks.

The aim of the combined approach was to train a higher quality sentiment classifier. BERT for Japanese Twitter evaluated better on the target task with dedicated training on JTS1k. However, the JTS1k set is small, and therefore volatile. Higher performance on the WRIME dataset is considered a more reliable indication of quality. Compared to other models that used the combined approach, the target model earned the highest score on JTS1k, but it underperformed on WRIME. This chapter addressed the hypothesis that the Twitter model could outperform state-of-the-art on Japanese Twitter sentiment. In this regard, **BERT for Japanese SNS Sentiment** is considered a partial success.

Notably, this analysis shows that the target model exceeds OpenAI's latest release on JTS1k. However, it is important to remember that GPT-4 performed better with the original four-label problem. Additionally, GPT-4o maintained an edge over the target model on the WRIME dataset. Therefore, OpenAI models are either equal or more reliable classifiers for the target domain of Japanese Twitter. A key distinction is operational cost. This study involved expenditures of €25 on OpenAI API access. The money spent funded exploration, repeated studies, and a token-inefficient procedure, yet it remains quite costly for just a few thousand classifications. Chapter 12 demonstrates a workflow that classified over a million tweets, which would likely be too expensive, even with a well-

optimized procedure. OpenAI's latest budget release, GPT-4o mini, might address this issue. However, as it stands, it is too costly to be used as a simple classifier, making BERT for Japanese SNS Sentiment a superior choice.

BERT for Japanese SNS Sentiment was proposed as a top-performing model compared to others of the same parameter size. However, the base LUKE model performed significantly better on the WRIME evaluation with only 20% more parameters. If this study were to be repeated, exploring other models, LUKE would be an excellent candidate for a foundational model. Another exciting release is the latest Japanese DeBERTa by Kyoto University. When evaluated by JGLUE, this model outperforms the Tohoku model in every task category and even surpasses the large LUKE model in some tasks. Furthermore, it is bilingual in English and Japanese, which should transfer well to the Twitter domain.

The method of domain adaptation with an established model and an adapted vocabulary was an interesting and mostly successful study. Given the opportunity to pre-train another model, a foundation model with a vocabulary better suited for Twitter would be selected. Pre-training would take full advantage of transfer learning by keeping the original vocabulary intact.

Chapter 11: Demonstration of Sentiment Analysis

Benchmarking in Chapter 10 culminated in fine-tuning the target model, **BERT for Japanese SNS Sentiment**. Additionally, as a peripheral contribution, WRIME was used to fine-tune another model, **BERT for Japanese SNS Emotion**. These models are designed to structure data sourced from Japanese social networking services (SNS). SNS data is difficult to manage because it is voluminous and chaotic. Traditional methods organize SNS data by objective features such as conversations, keywords, or timestamps. These models allow for finer categorization based on subjective features like sentiment and emotion. This enhanced capability provides a deeper insight into online discourse, benefiting organizations, policymakers, and researchers who rely on public sentiment for decision-making. This chapter demonstrates the practical uses of these models. The first exercise uses tweets about Malta to demonstrate the behavior of models in a familiar context. The next section utilizes a larger corpus of tweets about a controversial Japanese athlete. By clustering tweets with objective and subjective features, linguistic analysis is used to depict the progression of an unfolding story with distinct polarity shifts.

11.1: Sentiment Analysis of Tweets about Malta

Japan and Malta have an economic relationship that focuses on technology, tourism, and maritime trade. Malta is a prime destination for English language education, enticing students with its picturesque Mediterranean scenery and historical appeal. Furthermore, Malta's low crime rate has marked it as a safe destination, attracting Japanese travelers that value security. However, recent controversies in the online casino industry have cast a shadow over Malta’s safety reputation. Following an embarrassing incident in 2022, Japanese authorities clarified that online casinos are, and always have been, illegal. They have committed to prosecuting those involved in the industry as criminals, and their tactics have steadily ramped up in aggression. On June 29, 2024, Japan's national broadcaster, NHK, aired an exposé that criticized Malta's involvement in this industry. The undercover reporter spoke with young Japanese students and travelers who were lured into high-paying positions as online casino dealers. With stricter regulation, many fear stigmatization as criminals upon their return to Japan. This initial exercise introduces the models by showcasing outputs on individual tweets. Two topics were selected with distinct underlying sentiment. Positive sentiment was targeting by querying pastizzi (パスティッツィ), a popular Maltese snack. Negative sentiment was sourced from reactions to the NHK broadcast. Figures 10.1-10.2 show an example from each sentiment class for both topics. The caption contains a qualitative description of recurring themes from the sample.

Figure 11.1 Sentiment Analysis of Tweets about Pastizzi

*Tweets about pastizzi were mostly positive, but the sentiments range. The **Positive** tweets commonly mention the snack's rich ricotta filling and affordability, often reflecting a nostalgic longing to return to Malta for another taste. The **Neutral** tweets typically present factual observations, such as price and availability, without an enthusiastic endorsement. On the other hand, the **Negative** tweets highlight frustrations, such as availability issues or unmet expectations. The selected examples demonstrate divergence between general and aspectual sentiment analysis. Each tweet reflects a positive attitude towards pastizzi. The short-term visitor that authored the positive example relished the experience of connecting with a local. The neutral example's explanation reflects respect for the culture and language. The negative example laments the missed opportunity to enjoy a classic snack pairing.*

Figure 11.2 Sentiment Analysis of Tweets about the NHK Broadcast[16]

*Reactions to the NHK broadcast universally condemned online casinos. The **Negative** tweets focused on concerns over gambling addiction and fraud. Some commentors criticize Malta for enabling the industry. The **Neutral** offers a more balanced view, focusing on the factual details of NHK's investigation, underlining the ethical dilemmas and regulatory challenges. The few **Positive** tweets praised the quality of reporting, expressing gratitude to NHK for educating the public about a little-known problem.*

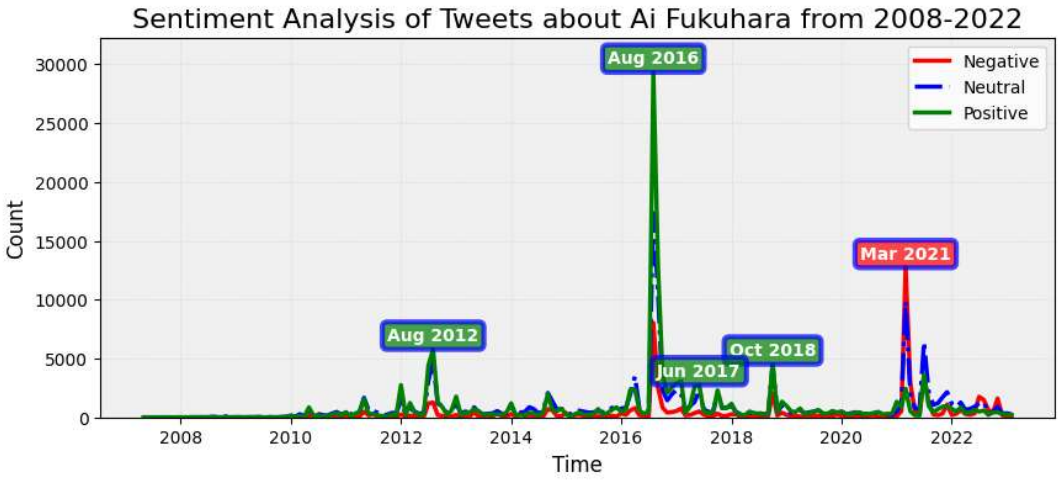
[16] https://www.nhk.jp/p/special/ts/2NY2QQLPM3/episode/te/689LG7QGGZ/

11.2: Sentiment Analysis of Tweets about Ai Fukuhara

The next exercise scales up the analysis, focusing on overarching features as well as the content of individual tweets. The aim is to validate the models by correlating fluctuations in public sentiment with real-world events (Barnaghi et al., 2016). This analysis focuses on Ai Fukuhara, a popular table tennis athlete. Fukuhara was beloved in her early years, captivating the public during her Olympic performances. Later in life, she became the subject of scandal. This analysis examines tweets about Fukuhara during this tumultuous period of her life.

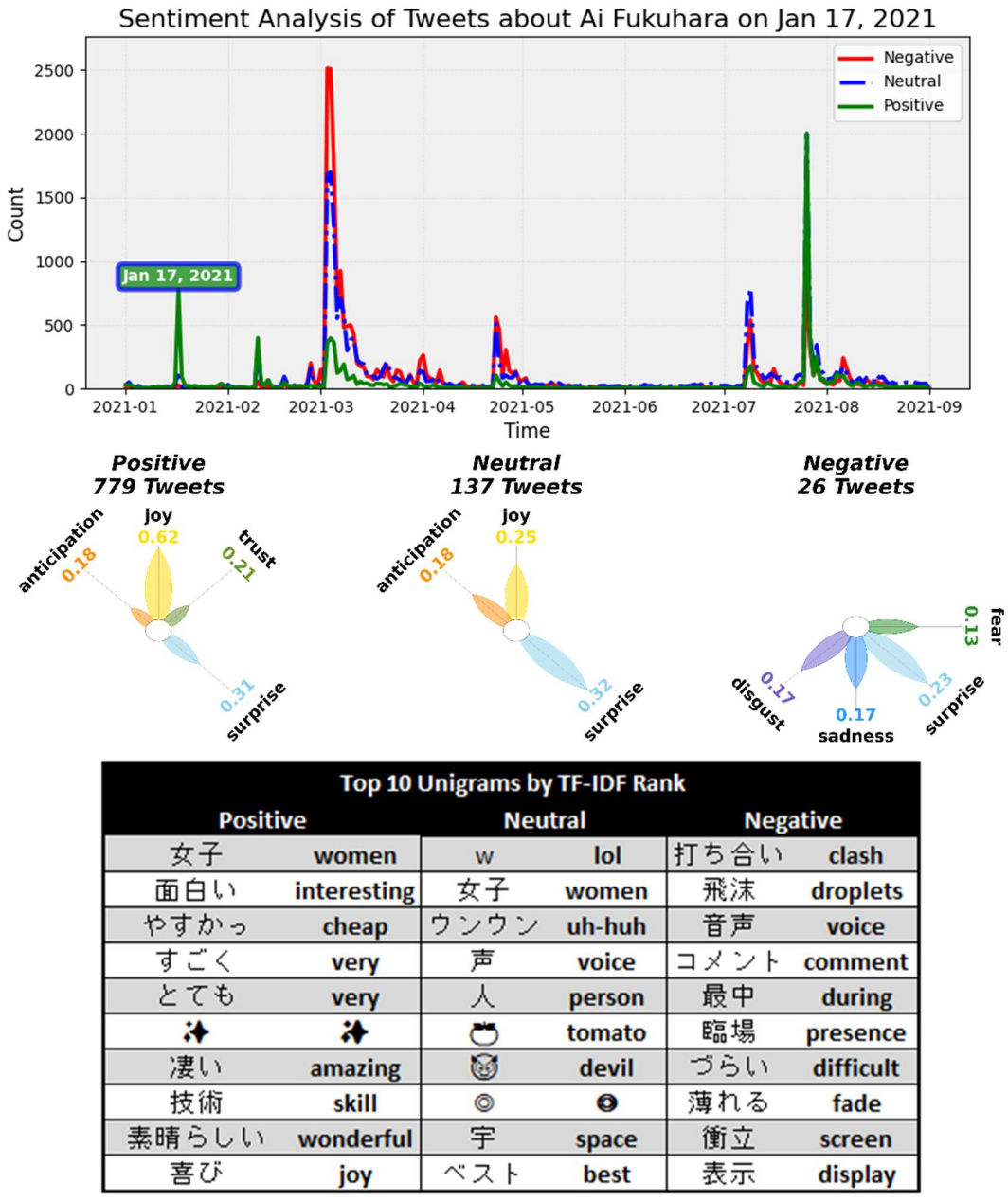
The sentiment analysis includes four components. Figure 10.2 exemplifies the first output, a line plot that visualizes fluctuations in sentiment over time. Peaks are flagged and color-coded to highlight dates of interest. This exercise focuses on four days that varied in dominant sentiment. On each target day, tweets are clustered by sentiment, and two analyses are performed. First, emotional intensity scores are averaged and plotted for each sentiment class. If all systems function correctly, emotion and sentiment should align—such as joy and trust for *positive* sentiment, and disgust and fear for *negative* sentiment. The second analysis ranks unigrams using a TF-IDF score. The top-ranking terms provide context on the topic and characterize the unique vocabulary from each sentiment class. For qualitative analysis, thirty tweets were sampled for each day and sentiment. The samples were balanced by their dominant emotion to ensure variety. The captions provide context through a comparison of recurring themes by sentiment.

Figure 11. Overview of Tweets about Fukuhara



These tweets were collected by querying Fukuhara's name in kanji, 福原 愛. The corpus includes 1.2 million tweets from over 500,000 users. Strong positive peaks in the summers of 2012 and 2016 correspond to her Olympic performances. In the spring of 2021, the media exposed her marital problems and began to publish allegations of infidelity. This analysis focuses on public sentiment during this period of scandal, from January to September 2021.

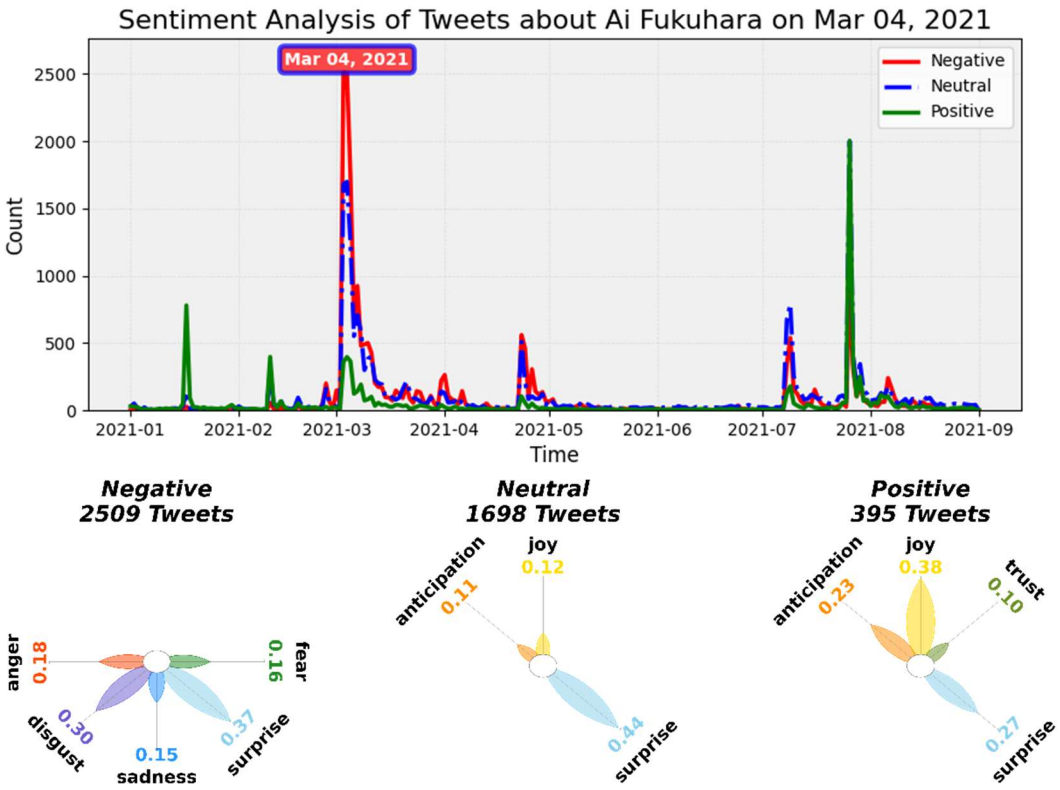
Figure 11.3 Tweets about Fukuhara on Jan 17, 2021



Top 10 Unigrams by TF-IDF Rank					
Positive		Neutral		Negative	
女子	women	w	lol	打ち合い	clash
面白い	interesting	女子	women	飛沫	droplets
やすかっ	cheap	ウンウン	uh-huh	音声	voice
すごく	very	声	voice	コメント	comment
とても	very	人	person	最中	during
✨	✨	🍅	tomato	臨場	presence
凄い	amazing	😈	devil	づらい	difficult
技術	skill	◎	❸	薄れる	fade
素晴らしい	wonderful	宇	space	衝立	screen
喜び	joy	ベスト	best	表示	display

*On **January 17, 2021**, Fukuhara commentated on the television broadcast of the women's division of a national table tennis tournament. **Positive** tweets congratulate the winners for their thrilling performance in an exciting series of matches. Fukuhara's commentary is praised for being clear, insightful and engaging. **Neutral** tweets also commend Fukuhara but focus more on the broader tournament context. Recurring topics include COVID-19, player withdrawals, and the men's competition. In contrast, **Negative** tweets criticize Fukuhara, citing issues such as talking too much, being distracting, or using poor analogies, with some suggesting that she ruined the experience.*

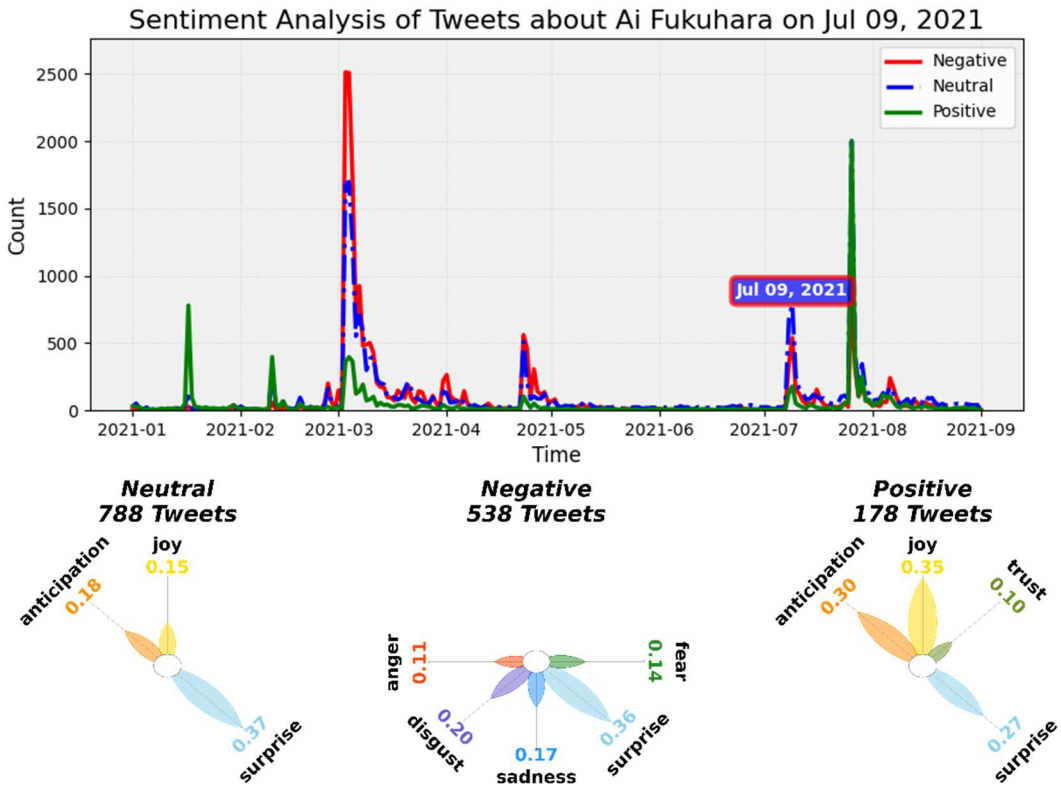
Figure 11.4 Tweets about Fukuhara on Mar 4, 2021



Top 10 Unigrams by TF-IDF Rank					
Negative		Neutral		Positive	
否定	denial	事実	fact	興奮	excitement
宿泊	accommodation	版	edition	本日	today
Web	Web	アピール	appeal	www	lol
可哀想	pitiful	宿泊	accommodation	是非	by all means
深く	deeply	速攻	quickly	ＯＯ	ＯＯ
詫び	apology	対策	measure	タイムリー	timely
務める	serve	違約	breach of contract	かわいい	cute
違約	breach of contract	まとめ	summary	²	²
悲報	sad news	円満	harmonious	競演	competition
浮気	infidelity	否定	denial	ssi	ssi

*On **March 4, 2021,** Twitter users reacted to media reports about Fukuhara that alleged marital stress and infidelity. **Negative** tweets primarily express shock, disappointment, and sadness. Many are supportive of Fukuhara, expressing concern about her family's reputation and criticizing the invasion of her privacy. **Neutral** tweets also reflect mixed reactions, discussing the media's handling of the situation and its impact on her public image and personal life. **Positive** tweets, though varied, express excitement or amusement by the scandal. Some share personal anecdotes about events and locations that circumstantially connect the commentors with the scandal. Many were defamatory, garnering a positive classification by framing derogatory comments with symbols of laughter.*

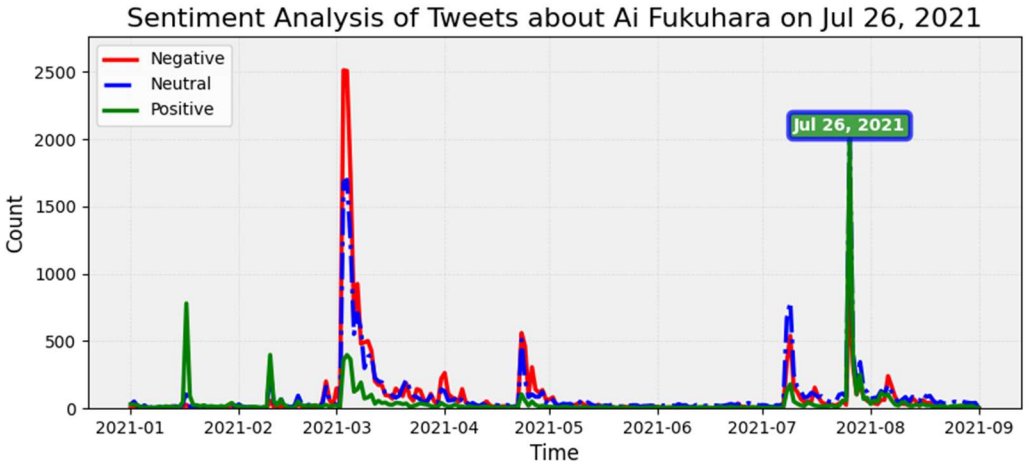
Figure 11.5 Tweets about Fukuhara on Jul 9, 2021

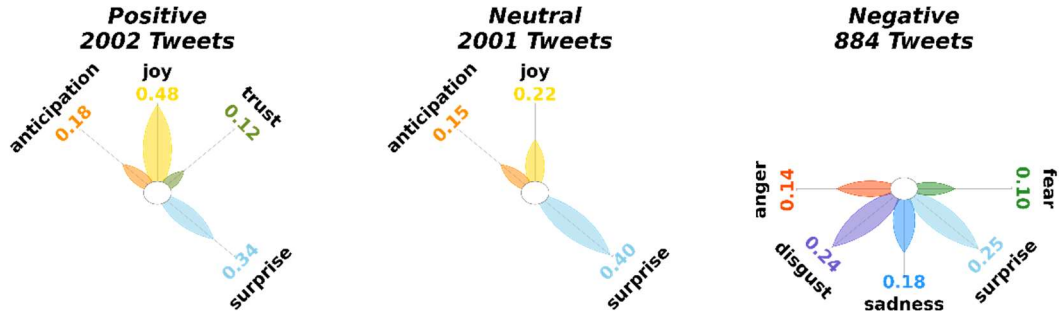


Top 10 Unigrams by TF-IDF Rank					
Neutral		Negative		Positive	
五輪	Olympics	多難	many difficulties	♥	♥
LINE	LINE	前途	future prospects	ボディ	body
記者	reporter	面目	repetition	画像	image
暮らす	live	保つ	keep	安心	relief
linenews	linenews	うやむや	ambiguous	激す	intense
ビジネス	business	悲報	sad news	よろしく	best regards
タピオカ	tapioca	女性	female	11	11
主に	mainly	疑惑	suspicion	07	7
店	store	PRIME	PRIME	人選	selection
説明	explanation	五輪	Olympics	権限	authority

*After a sustained period of negative attention, Fukuhara officially announced intent to divorce on **July 9, 2021.** The **Neutra**l tweets reference official statements made by the family. Some express concern for their children's future. They contain a mix of critical and supportive comments regarding her alleged infidelity and public image, occasionally using humour and personal anecdotes. The **Negative** tweets highlight distressing emotional responses to the news. They were more critical of Fukuhara and fearful for her family. Some are supportive, angrily speaking out about societal double standards. The **Positive** tweets commend her for raising awareness about joint custody in Japan, framing it as a step forward for children's welfare. Fans of Fukuhara praise her charm and endearing presence in the public eye. Her plight inspires messages of hope and resilience.*

Figure 11.6 Tweets about Fukuhara on Jul 26, 2021





Top 10 Unigrams by TF-IDF Rank					
Positive		Neutral		Negative	
綺麗	beautiful	あら	oh	叩く	hit
笑う	laugh	垢	account	興醒め	killjoy
❤	❤	文化	culture	😭	😭
復活	revival	○	○	嫌悪	aversion
😊	😊	w	lol	偉	great
w w w	lol	説	theory	萎える	wither
突破	breakthrough	えっ	eh	不快	discomfort
😭	😭	ビビっ	startled	😔	😔
本日	today	両論	both arguments	ほしく	want
我	I	優先	priority	直前	just before

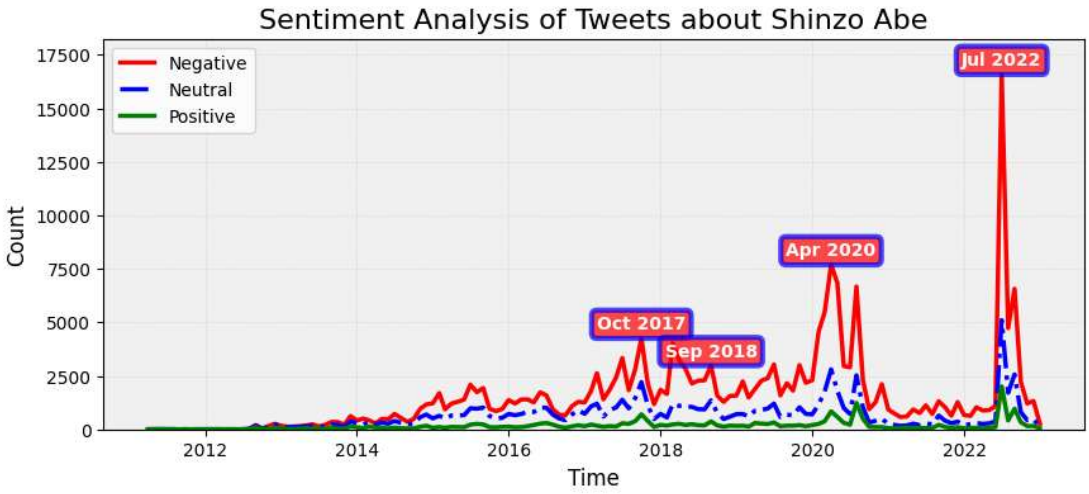
*On **July 26, 2021**, Fukuhara commentated on a dramatic table tennis match at the Tokyo Olympics where Japan won gold. The **Positive** tweets praise the performance of the winning duo. Many commend Fukuhara, complimenting her enthusiasm and strong presence despite her recent challenges. Overall, they emphasize joy and pride in Japan's achievement and admiration for Fukuhara. The **Neutral** tweets are more mixed. Many acknowledge the historic gold medal win, but express discomfort about her presence. While the athletes are celebrated, Fukuhara's presence tempers their excitement. The **Negative** tweets focus more on Fukuhara than the event. These tweets are blunt and critical, with some using derogatory language to express their dissatisfaction. They openly discuss her scandals, complain about her presence, and criticize organizers for including her.*

11.3: Interim Conclusion

The models effectively generalized to the Fukuhara dataset, demonstrating correlation between polarity spikes in time plots and real-world events. Unigram analysis identified distinct terms that are either topically or sentimentally relevant. Emotion scores across sentiment categories remained consistent between target days. *Negative* tweets predominantly expressed anger, sadness, fear, and disgust, while *positive* tweets conveyed joy, anticipation, and trust. Suprise was a consistent presence across sentiments, but most strongly in *neutral* tweets. It is unknown whether the prevalence of surprise is connected to the subject material or is a general linguistic feature. Ramos et al. (2022) used WRIME to analyze COVID-19 related tweets. Sentiment did not factor into their analysis, but the publication reports that surprise was the most observed emotion. The interplay of sentiment, emotion, and real-world events is a practical subject that warrants continued academic investigation. The models may prove useful for further investigation of Japanese Twitter.

Sentiment analysis of Fukuhara revealed a divergence between general and aspect-based sentiment, especially on the day the scandal broke (Figure 11.4). Some positive tweets mocked the controversy, while negative tweets condemned media indiscretion and public bullying. Similar divergence was observed in an analysis of controversial politician, Shinzo Abe (Figure 11.7). Many negative tweets criticized his scandals and policies, but the majority were defensive, attacking detractors and portraying him positively. The bias in these samples likely stems from the querying strategy that focused on the target's name. Different keywords may yield a more rounded sample.

Figure 11.7 Overview of Tweets about Shinzo Abe



These tweets, showing a strong negative distribution, were collected by querying Abe's kanji name, 安倍 晋三. The dataset consists of 350,000 tweets by 55,000 authors. Spikes in activity, correspond with scandals, political events, and his assassination in the summer of 2022.

Chapter 12: Conclusion

This concluding chapter revisits the research questions from Chapter 2 and proposes some future directions.

12.1: Responding to the Research Questions

What are the essential qualities of a training corpus for unsupervised pre-training? Which sampling and preprocessing methods produce the best corpus?

The study began with the development of a corpus from Twitter data, focusing on the key qualities of a training corpus: size, balance, and representativeness. The Twitter API proved effective in gathering a substantial volume of data, and initial analysis supported that the training material represented the target features of language, place, and time. However, two balance issues became apparent. First, highly active users were overrepresented, with most content produced by users that contributed hundreds or even thousands of times. Second, the corpus contained a high degree of redundancy, with numerous near-duplicate posts. It was hypothesized that these issues stemmed from a common cause: users with unusually high posting frequencies likely represented business interests, contributing spam-like content that compromised corpus quality. If this assumption held, the primary refinement method, near-duplicate deduplication, would have significantly improved user balance in the training corpus. However, although some balancing was achieved, the training corpus remained unbalanced after the deduplication process.

Following deduplication, a second training corpus was prepared that compromised size for user balance. It was expected that the balanced corpus would maintain the diversity of content of deduplicated corpus. Because it was smaller, there would be more opportunity for hyperparameter tuning, ultimately leading to a higher performing model. In fact, the analysis carried out in tables 5.5 to 5.8 show that while deduplication achieved its intended effect, balancing did not improve content diversity by any measure. During hyperparameter turning, although nearly twice as many models were tuned using the balanced corpus, those pre-trained on the deduplicated corpus performed at a higher level.

Upon further reflection, the assumption that highly active users were producing lower-quality content was found to be problematic. For example, more sophisticated opinion-spamming strategies often involve multiple accounts, meaning post frequency alone is not a reliable predictor of spam likelihood. Additionally, user balancing suppresses prominent voices on Twitter, such as content creators, news organizations, and influencers. Diverse, timely, and niche content is essential for

maintaining engagement, and these users may be more likely to employ unique language that is valuable for training a language model. When working with web corpora, researchers should anticipate an uneven contribution from users. N-gram based deduplication proved effective for improving corpus quality. Beyond that, priority should be given to training on the maximum corpus size allowed by hardware constraints.

What characteristics define an effective sentiment analysis training set?

As with pre-training, key considerations for fine-tuning include size, balance, and representativeness. The JTS1k training set was particularly effective given its smaller size, a strength best demonstrated in cross-lingual transfer studies with XLM-T, where it outperformed much larger datasets. WRIME, while proposed as a superior dataset due to its larger size, emotional content balance, and accessibility, was derived entirely from past SNS posts of only 60 users (Suzuki et al., 2022). In contrast, the smaller JTS1k set is more diverse, with each tweet contributed by a different user, enhancing its representativeness. Additionally, JTS1k includes non-standard characters, a valuable feature for the target domain. Thus, despite WRIME's size, JTS1k fulfills an important niche in sentiment analysis. With the innovation of Transformers, the minimum threshold of required training examples has significantly decreased; more important now is the accurate representation of the target task and domain.

What are the best practices for directing crowdworkers to ensure high-quality annotations?

The methodology of Participatory Design proved effective in building the JTS1k dataset. Involving native Japanese speakers in the design process ensured that the procedure was robust and clearly defined. Crowdworkers completed tasks diligently, with no signs of burnout. Although this strategy worked well for this study, it highlighted equity concerns within the human annotation industry. Dataset annotation is labor-intensive, and past studies have often relied on unpaid labor from graduate students. With rising demand for labeled datasets to train AI, advertisements for low-level annotation work have become more common, but the compensation offered is frequently low. In this study, crowdworkers were compensated approximately €2 for 15 minutes of work, a rate that appeared slightly above average compared to other listings on *CrowdWorks*. On the other end of the spectrum, established organizations like OpenAI and Cohere for AI reportedly compensate highly qualified annotators at rates of $40 per hour or more. While such rates improve conditions for annotators, they are often unaffordable for smaller-scale startups. Given the challenges of validating annotation quality at scale, especially for open-ended tasks, it is essential to establish clearly defined

task instructions and equitable terms, ensuring that data collection remains both reliable and mutually beneficial.

Does domain adaptation result in a model that outperforms on Twitter-specific tasks? How does it compare with larger, general-purpose models? Does the Twitter-adapted model maintain general task proficiency? Was the investment in continued pre-training justified?

The pre-training approach met the minimum criteria for success, and domain adaptation notably enhanced performance on social media-related tasks. While the largest state-of-the-art models continued to outperform the Twitter model, they did so at the cost of a significantly larger parameter footprint. As expected, the Twitter model exhibited some loss of competence in the general domain. Although this reduction in performance appeared acceptable, it remains not well understood. A comparison with hottoSNS-BERT, which was trained from scratch on a Twitter corpus (Sakaki et al., 2019), would have been beneficial.

Assessing whether the investment in continued pre-training was justified remains challenging. In retrospect, resources devoted to hyperparameter tuning were excessive, as it became clear that the quality of training material was the most critical factor. Allocating these resources toward a broader exploration of training material could have been more impactful. A noteworthy resource is the National Institute for Japanese Language and Linguistics (NINJAL), which has published various domains and modalities of corpora[17]. It would be interesting to see if a combined training approach that incorporates a web corpus or spoken-word corpus would yield more generalizable results.

Does the fine-tuned sentiment model perform as expected in real-world applications?

The demonstration of sentiment analysis supports that the models have real-world applications. Pre-training the models was extremely resource intensive, and the costs are only justified if third parties use and benefit from the models. Proper documentation is crucial for attracting the interest of third parties. TweetNLP sets an important example by packaging the functionalities of their models into concise code and providing clear documentation (Camacho-Collados, et al., 2022). Documentation for this project's models is a work in progress, with a commitment to providing it in Japanese to better serve the target audience.

[17]NINJAL Corpus Catalogue: https://clrd.ninjal.ac.jp/en/subscription.html

12.2: Future Directions

Parameter Efficient Fine-Tuning

An emerging area of research interest is parameter efficient fine-tuning. This approach focuses on fine-tuning models using a targeted selection of parameters, representing a significant advancement in modular deep learning (Pfeiffer et al., 2024). The goal is to improve the efficiency of model design by exploring new paradigms in training and architecture. This study experimented with adapter layers (Pfeiffer et al., 2021), which absorb task-level knowledge without requiring updates to the entire network of parameters. Adapter models perform nearly as well as those fine-tuned across the full network. The primary advantage of adapter layers is that they are compact. The addition of more adapter layers to a base model enhances the functionality of the model while minimizing the additional strain on hardware.

Table 12.1 Performance of Adapter Models on Social Media Tasks and JGLUE

	JTS1k Acc	JTDD Acc	JTBR Acc	WRIME (emo) Top-1 / F1	WRIME (sent) Pear / Spear
Full	**0.737**	**0.649**	**0.864**	0.694 / 0.639	**0.865 / 0.868**
Adapter	0.719	0.618	0.854	**0.697 / 0.641**	0.863 / 0.866

	MARC-ja Acc	JSTS Pear / Spear	JCoLA Acc	JNLI Acc	JCSQA Acc
Full	0.959	**0.899 / 0.861**	**0.865**	**0.87**	**0.745**
Adapter	**0.961**	0.874 / 0.828	0.854	0.865	0.729

The adapter models performed optimally or near-optimally on every task, introducing the opportunity for a parameter efficient, multi-task model.

Alternative Sources of SNS Data

This study explored some functionalities in the Twitter API that are not well documented in other publications. It is believed that using keyword-based sampling techniques, reinforced by better monitoring of the contents of samples, would lead to a far superior training corpus. Unfortunately, researchers are no longer welcome on the Twitter API. Given the high cost of entry[18], it seems that Twitter is not interested in sharing their data with anyone. Fortunately, TikTok has stepped up their access policies[19]. Not only does this fulfill an important need in research, but it also presents new opportunities for multi-modal sentiment analysis.

[18] $5,000 per month, according to their documentation. https://developer.x.com/en/docs/twitter-api
[19] https://developers.tiktok.com/products/research-api/

Works Cited

Al-Rfou, R., Choe, D., Constant, N., & Jones, L. (2019). Character-Level Language Modeling with Deeper Self-Attention. *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence.* Honolulu, Hawaii, USA: AAAI Press.

Association of Internet Researchers. (2019). *Ethics Guidelines for Internet Research, Version 3.0.* Retrieved from https://aoir.org/reports/ethics3.pdf

Barbieri, F., Anke, L. E., & Camacho-Collados, J. (2022). XLM-T: Multilingual Language Models in Twitter for Sentiment Analysis and Beyond. *Proceedings of the Language Resources and Evaluation Conference* (pp. 258--266). Marseille, France: European Language Resources Association.

Barbieri, F., Camacho-Collados, J., Espinosa Anke, L., & Neves, L. (2020). TweetEval: Unified Benchmark and Comparative Evaluation for Tweet Classification. *Findings of the Association for Computational Linguistics: EMNLP 2020* (pp. 1644--1650). Online: Association for Computational Linguistics.

Barnaghi, P., Ghaffari, P., & Breslin, J. G. (2016). Opinion Mining and Sentiment Polarity on Twitter and Correlation between Events and Sentiment. *16 IEEE Second International Conference on Big Data Computing Service and Applications (BigDataService)*, (pp. 52-57).

Bedrick, S., Beckley, R., Roark, B., & Sproat, R. (2012). Robust kaomoji detection in Twitter. *Proceedings of the Second Workshop on Language in Social Media* (pp. 56-64). Montréal, Quebec: Association for Computational Linguistics.

Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of Computational Science*, 1877-7503.

Bowman, S. R., Angeli, G., Potts, C., & Manning, C. D. (2015). A large annotated corpus for learning natural language inference. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (pp. 632--642). Lisbon, Portugal: Association for Computational Linguistics.

Broder, A. Z. (1997). On the resemblance and containment of documents. *Proceedings. Compression and Complexity of SEQUENCES 1997* (pp. 21-29). IEEE.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., . . . Child, R. (2020). Language Models are Few-Shot Learners. *Proceedings of the 34th International Conference on Neural*

Information Processing Systems. Vancouver, BC, Canada: Curran Associates Inc. Retrieved from https://arxiv.org/abs/2005.14165

Buscemi, A., & Proverbio, D. (2024). *ChatGPT vs Gemini vs LLaMA on Multilingual Sentiment Analysis.* arXiv.

Camacho-Collados, J., Rezaee, K., Riahi, T., Ushio, A., Loureiro, D., Antypas, D., . . . Neves, L. (2022). *TweetNLP: Cutting-Edge Natural Language Processing for Social Media.* arXiv.

Cieliebak, M., Deriu, J. M., Egger, D., & Uzdilli, F. (2017). A Twitter Corpus and Benchmark Resources for German Sentiment Analysis. *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media* (pp. 45-51). Valencia, Spain: Association for Computational Linguistics. doi:10.18653/v1/W17-1106

Conneau, A., & Lample, G. (2019). Cross-lingual Language Model Pretraining. *Proceedings of the 33rd International Conference on Neural Information Processing Systems.* Red Hook, NY, USA: Curran Associates Inc.

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., . . . Stoyanov, V. (2020). Unsupervised Cross-lingual Representation Learning at Scale. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 8440-8451). Online: Association for Computational Linguistics.

Dave, K., Lawrence, S., & Pennock, D. M. (2003). Mining the peanut gallery: opinion extraction and semantic classification of product reviews. *Proceedings of the 12th International Conference on World Wide Web* (pp. 519–528). New York, NY, USA: Association for Computing Machinery.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (pp. 4171--4186). Minneapolis, Minnesota: Association for Computational Linguistics.

Elangovan, A., He, J., & Verspoor, K. (2021). Memorization vs. Generalization: Quantifying Data Leakage in NLP Performance Evaluation. *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume* (pp. 1325--1335). Online: Association for Computational Linguistics.

French, R. M. (1999). Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 128-135.

Giray, L. (2023, June 7). Prompt Engineering with ChatGPT: A Guide for Academic Writers. *Annals of Biomedical Engineering, 51*, 2629–2633. Retrieved from https://doi.org/10.1007/s10439-023-03272-4

Godbole, V., Dahl, G. E., Gilmer, J., Shallue, J., C., & Nado, Z. (2023). *Deep Learning Tuning Playbook.* Retrieved from http://github.com/google-research/tuning_playbook

Godey, N., Castagné, R., de la Clergerie, É., & Sagot, B. (2022). MANTa: Efficient Gradient-Based Tokenization for End-to-End Robust Language Modeling. *Findings of the Association for Computational Linguistics: EMNLP 2022* (pp. 2859--2870). Abu Dhabi, United Arab Emirates: Association for Computational Linguistics.

Gururangan, S., Marasovic, A., Swayamdipta, S., Lo, K., Beltagy, I., Downey, D., & Smith, N. A. (2020). Don't Stop Pretraining: Adapt Language Models to Domains and Tasks. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 8342--8360). Online: Association for Computational Linguistics.

Hayes, A., & Krippendorff, K. (2007). 2007. *Answering the Call for a Standard Reliability Measure for Coding Data*, 77-89.

He, P., Liu, X., Gao, J., & Chen, W. (2021). DeBERTa: Decoding-enhanced BERT with Disentangled Attention. *International Conference on Learning Representations.* Retrieved from https://openreview.net/forum?id=XPZIaotutsD

Hong, Y., & Skiena, S. (2010). The Wisdom of Bookies? Sentiment Analysis vs. the NFL Point Spread. *Proceedings of the International Conference on Weblogs and Social Media (ICWSM-2010).*

Kajiwara, T., Chu, C., Takemura, N., Nakashima, Y., & Nagahara, H. (2021). WRIME: A New Dataset for Emotional Intensity Estimation with Subjective and Objective Annotations. *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 2095-2104). Online: Association for Computational Linguistics.

Keshi, I., Suzuki, Y., Yoshino, K., Neubig, G., Ohara, K., Mukai, T., & Nakamura, S. (2017, April). Reputation Information Extraction from Twitter Using a Word Semantic Vector Dictionary. *IEICE Transaction, J100-D*(4), 530--543.

Keung, P., Lu, Y., Szarvas, G., & Smith, N. A. (2020). The Multilingual Amazon Reviews Corpus. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 4563--4568). Online: Association for Computational Linguistics.

Kingma, D. P., & Ba, J. (2017). *Adam: A Method for Stochastic Optimization.* arXiv. Retrieved from https://arxiv.org/abs/1412.6980

Krogh, A., & Hertz, J. A. (1991). A simple weight decay can improve generalization. *Proceedings of the 4th International Conference on Neural Information Processing Systems* (pp. 950–957). Denver, Colorado: Morgan Kaufmann Publishers Inc.

Krugmann, J. O., & Hartmann, J. (2024). Sentiment Analysis in the Age of Generative AI. *Customer Needs and Solutions*. Retrieved from https://doi.org/10.1007/s40547-024-00143-4

Kudo, T., & Richardson, J. (2018). SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations* (pp. 66--71). Brussels, Belgium: Association for Computational Linguistics.

Kudo, T., Yamamoto, K., & Matsumoto, Y. (2004). Applying conditional random fields to Japanese morphological analysis. *Proceedings of the 2004 conference on empirical methods in natural language processing*, (pp. 230-237).

Kurihara, K., Kawahara, D., & Shibata, T. (2022). JGLUE: Japanese General Language Understanding Evaluation. *Proceedings of the Thirteenth Language Resources and Evaluation Conference* (pp. 2957-2966). Marseille, France: European Language Resources Association.

Lai, S., Hu, X., Xu, H., Ren, Z., & Liu, Z. (2023). *Multimodal Sentiment Analysis: A Survey.* doi:https://doi.org/10.1016/j.displa.2023.102563

Lee, K., Ippolito, D., Nystrom, A., Zhang, C., Eck, D., Callison-Burch, C., & Carlini, N. (2022). Deduplicating Training Data Makes Language Models Better. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 8424--8445). Dublin, Ireland: Association for Computational Linguistics. Retrieved from https://arxiv.org/abs/2107.06499

Liu, B. (2012). Sentiment Analysis and Opinion Mining.

Loshchilov, I., & Hutter, F. (2017). Fixing Weight Decay Regularization in Adam. *CoRR, abs/1711.05101*. Retrieved from http://arxiv.org/abs/1711.05101

Loureiro, D., Barbieri, F., Neves, L., Espinosa Anke, L., & Camacho-collados, J. (2022). TimeLMs: Diachronic Language Models from Twitter. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations* (pp. 251--260). Dublin, Ireland: Association for Computational Linguistics.

McCann, P. (2020). *fugashi, a Tool for Tokenizing Japanese in Python.* arXiv.

McGlohon, M., Glance, N., & Reiter, Z. (2010). Star Quality: Aggregating Reviews to Rank Products and Merchants. *Proceedings of the International AAAI Conference on Web and Social Media* (pp. 114-121). AAAI. Retrieved from https://doi.org/10.1609/icwsm.v4i1.14019

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient Estimation of Word Representations in Vector Space.* arXiv.

Miyauchi, Y., Akiyama, K., Kajiwara, T., Ninomiya, T., Takemura, N., Nakashima, Y., & Nagahara, H. o. (2022). A Japanese Dataset for Subjective and Objective Sentiment Polarity Classification in Micro Blog Domain. *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, 7022-7028.

Miyazaki, T., & Shimizu, N. (2016). Cross-Lingual Image Caption Generation. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 1780--1790). Berlin, Germany: Association for Computational Linguistics.

Mohammad, S. M. (2012). From once upon a time to happily ever after: Tracking emotions in mail and books. *Decision Support Systems*, 730-741.

Mohammad, S. M., & Yang, T. (2013). Tracking Sentiment in Mail: How Genders Differ on Emotional Axes. *CoRR*. Retrieved from http://arxiv.org/abs/1309.6347

Mozetič, I., Grčar, M., & Smailović, J. (2016). Multilingual Twitter Sentiment Classification: The Role of Human Annotators. *PLoS ONE*.

Nakov, P., Ritter, A., Rosenthal, S., Sebastiani, F., & Stoyanov, V. (2016). SemEval-2016 Task 4: Sentiment Analysis in Twitter. *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)* (pp. 1--18). San Diego, California: Association for Computational Linguistics.

Nguyen, D. Q., Vu, T., & Nguyen, A. T. (2020). BERTweet: A pre-trained language model for English Tweets. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations* (pp. 9--14). Online: Association for Computational Linguistics.

Petrović, S., Osborne, M., & Lavrenko, V. (2010). The Edinburgh Twitter Corpus. *Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media,* (pp. 25–26). Los Angeles, California, USA: Association for Computational Linguistics.

Pfeiffer, J., Kamath, A., Rücklé, A., Cho, K., & Gurevych, I. (2021). AdapterFusion: Non-Destructive Task Composition for Transfer Learning. *Proceedings of the 16th Conference of the European*

Chapter of the Association for Computational Linguistics: Main Volume (pp. 487–503). Online: Association for Computational Linguistics.

Pfeiffer, J., Ruder, S., Vulić, I., & Ponti, E. M. (2024). *Modular Deep Learning.* Retrieved from https://arxiv.org/abs/2302.11529

Poria, S., Hazarika, D., Majumder, N., & Mihalcea, R. (2023). Beneath the Tip of the Iceberg: Current Challenges and New Directions in Sentiment Analysis Research. *IEEE Transactions on Affective Computing, 14*(1), 108-132.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language Models are Unsupervised Multitask Learners. *OpenAI Blog*. Retrieved from https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

Rajpurkar, P., Zhang, J., Lopyrev, K., & Liang, P. (2016). Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. *Association for Computational Linguistics* (pp. 2383--2392). Austin, Texas: Association for Computational Linguistics.

Ramos, P. J., Ferawati, K., Liew, K., Aramaki, E., & Wakamiya, S. (2022). Emotion Analysis of Writers and Readers of Japanese Tweets on Vaccinations. *Proceedings of the 12th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis* (pp. 95-103). Dublin, Ireland: Association for Computational Linguistics.

Ruder, S. (2017). *An Overview of Multi-Task Learning in Deep Neural Networks.* arXiv.

Ruder, S., Peters, M. E., Swayamdipta, S., & Wolf, T. (2019). Transfer learning in natural language processing. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials* (pp. 15-18). Minneapolis, Minnesota: Association for Computational Linguistics.

Sakaki, T., Mizuki, S., & Gunji, N. (2019). BERT Pre-trained model Trained on Large-scale Japanese Social Media Corpus. Retrieved from https://github.com/hottolink/hottoSNS-bert

Sanders, E. B.-N., & Stappers, P. J. (2008). Co-creation and the new landscapes of design. *Co-Design, 4*(1), 5-18. doi:10.1080/15710880701875068

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper. *CoRR*. Retrieved from http://arxiv.org/abs/1910.01108

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 379-423.

Shoeybi, M., Patwary, M., Puri, R., LeGresley, P., Casper, J., & Catanzaro, B. (2019). Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism. *CoRR*. Retrieved from http://arxiv.org/abs/1909.08053

Sloan, L., Jessop, C., Al Baghal, T., & Williams, M. (2020). Linking Survey and Twitter Data: Informed Consent, Disclosure, Security, and Archiving. *Journal of Empirical Research on Human Research Ethics*, 63-76.

Someya, T., Sugimoto, Y., & Oseki, Y. (2024). CoLA: Japanese Corpus of Linguistic Acceptability. *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)* (pp. 9477--9488). Torino, Italia: ELRA and ICCL.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 1929–1958.

Suzuki, H., Miyauchi, Y., Akiyama, K., Kajiwara, T., Ninomiya, T., Takemura, N., . . . Nagahara, H. (2022). A Japanese Dataset for Subjective and Objective Sentiment Polarity Classification in Micro Blog Domain. *Proceedings of the Thirteenth Language Resources and Evaluation Conference* (pp. 7022--7028). Marseille, France: European Language Resources Association.

Talmor, A., Herzig, J., Lourie, N., & Berant, J. (2019). Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). *Proceedings of the 2019 Conference of the North {A}merican Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (pp. 4149--4158). Minneapolis, Minnesota: Association for Computational Linguistics.

van der Maaten, L., & Hinton, G. (2008). Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 2579-2605.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., . . . Polosukhin, I. (2017). Attention Is All You Need. *Proceedings of the 31st International Conference on Neural Information Processing Systems* (pp. 6000–6010). Long Beach, California, USA: Curran Associates Inc.

Warstadt, A., Singh, A., & Bowman, S. R. (2018). *Neural Network Acceptability Judgments.*

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., . . . Scao, L. (2020). *HuggingFace's Transformers: State-of-the-art Natural Language Processing.* arXiv.

Yamada, I., Asai, A., Shindo, H., Takeda, H., & Matsumoto, Y. (2020). LUKE: Deep Contextualized Entity Representations with Entity-aware Self-attention. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 6442--6454). Online: Association for Computational Linguistics.

You, Y., Li, J., Hseu, J., Song, X., Demmel, J., & Hsieh, C.-J. (2019). Reducing BERT Pre-Training Time from 3 Days to 76 Minutes. *CoRR*. Retrieved from http://arxiv.org/abs/1904.00962

Zaheer, M., Guruganesh, G., Dubey, A., Ainslie, J., Alberti, C., Ontanon, S., . . . Ahmed, A. (2020). Big Bird: Transformers for Longer Sequences. *Advances in Neural Information Processing Systems.* Curran Associates, Inc.

Zhuang, L., Wayne, L., Ya, S., & Jun, Z. (2021). RoBERTa: A Robustly Optimized BERT Pretraining Approach. *Proceedings of the 20th Chinese National Conference on Computational Linguistics* (pp. 1218--1227). Huhhot, China: Chinese Information Processing Society of China.

Appendix 1: Jaccard Similarity of N-Gram Shingles

The deduplication algorithm follows three steps.:

1. Texts are split into n-gram shingles.
2. Pair-wise Jaccard similarity is calculated between texts.
3. Those that exceed a similarity threshold are flagged as duplicates.

This example demonstrates how Jaccard similarity is calculated between two texts based on n-gram shingles.

Equation A1.1: Jaccard Similarity

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

Given two sets, A and B, the Jaccard similarity, $J(A,B)$, is given by the intersection, $|A \cap B|$, divided by the union, $|A \cup B|$. The intersection denotes shared items, and the union denotes all items.

Tweet 1: Just had an amazing coffee! !

Tweet 2: I just had an amazing coffee!

Tweet 1: n-grams: ['just had', 'had an', 'an amazing', 'amazing coffee!', 'coffee! !']

Tweet 2: n-grams: ['i just', 'just had', 'had an', 'an amazing', 'amazing coffee!']

Intersection: 4

['just had', 'had an', 'an amazing', 'amazing coffee!']

Union: 6

['just had', 'had an', 'an amazing', 'amazing coffee!', 'coffee! !', 'i just']

Jaccard Similarity: 0.66666

Appendix 2: Instructions and Solicitation for Crowdworkers

作業指示書

【ステップ 1】

まず、タイトルの下にあるリンクをクリックして、ツイートをご覧ください。

【ステップ 2】

ツイートをお読みになり、ツイートの内容が４つのうちどちらに属するかを選択してください。具体例については、次の 2 ページに記載されています。なお、ツイートを読んだご自身がどう感じるかではなく、ツイートの投稿者の感情や主張について選択していただくようお願い致します。

【ステップ 3】

全２９問への回答が終了した後は、回答を提出してください。回答は 7 日以内に確認されます。回答が承認された場合、報酬が支給されます。

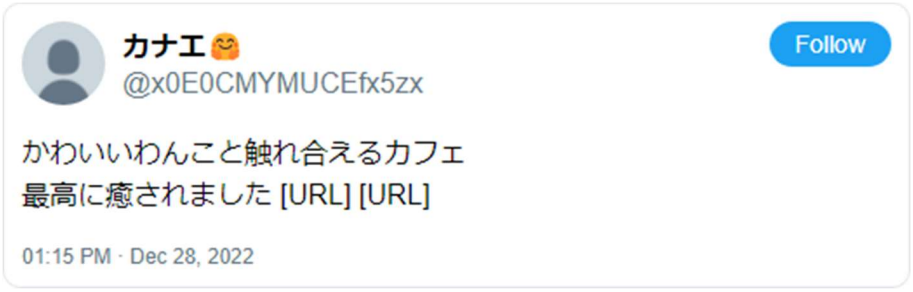
【最終注意事項】

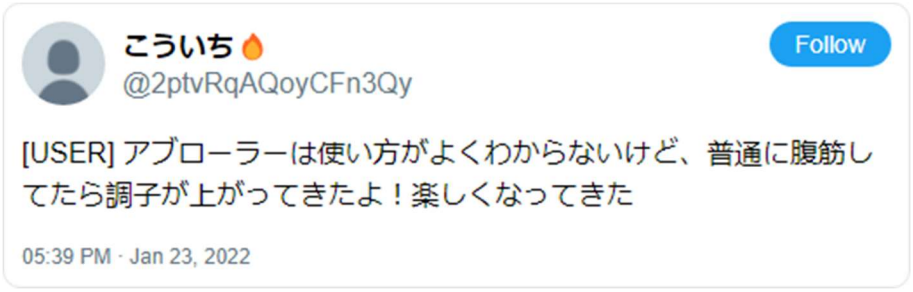
指示に従い、ツイートを慎重に確認してください。どちらに属するものか曖昧なツイートもありますが、この後の説明をよくお読みになり、最も近いと思われるものを選択してください。そして、各ツイートには 60 秒以上の時間をかけないようにしてください。

２９件のツイートの中に意見が非常に明確なツイート５件が含まれます。これらのツイート 5 つ中 4 つ以上に正解し、残りも熱心に完了していただいた場合、回答は受理され、報酬が支給されます。

【ポジティブ (肯定的)】

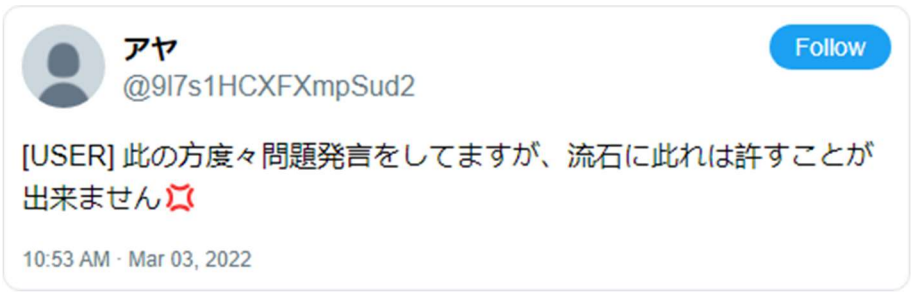
ツイートが完全に幸福感、満足感、または楽観的な態度を示している場合、または複数の感情が混在している中でポジティブな感情が最も強い場合、ポジティブな感情と見なされます。

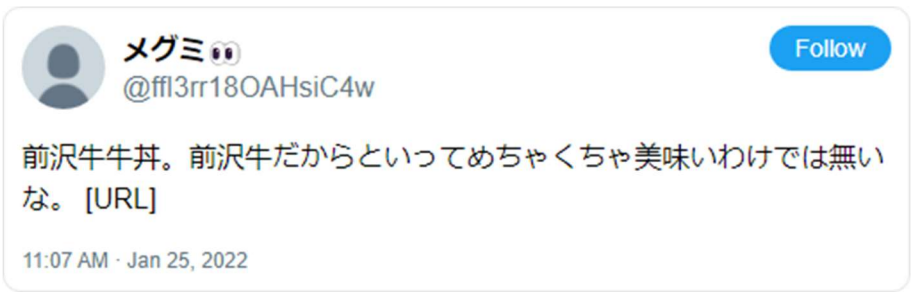




【ネガティブ (否定的)】

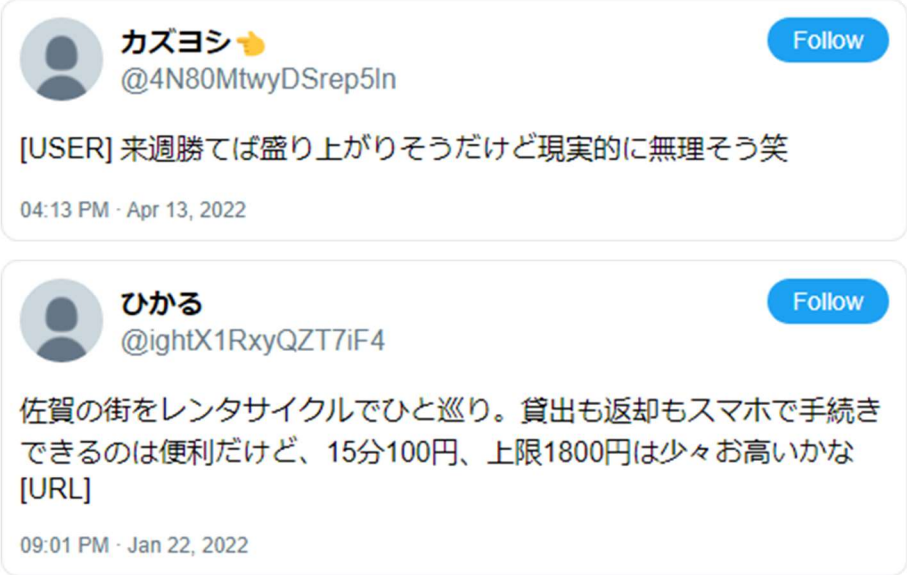
ツイートが不満、悲しみ、怒り、またはその他の否定的な態度を主に示している場合、または複数の感情が混在している中でネガティブな感情が最も強い場合、それはネガティブな感情と見なされます。





【ミックス (混合)】

ツイートがポジティブな感情とネガティブな感情を共に表現または暗示している場合、それは混合感情と見なされます。この場合、喜びと不満など、相反する感情が同時に示されていることが特徴です。



【ニュートラル (中立的)】

ニュートラルなツイートは感情を全く表現していないか、または極めて少ない場合です。ニュートラルな感情の例としては、事実を述べている場合、質問をしている場合、または他人の意見を求めている場合などがあります。

[Step 1]

First, click on the link below the heading to see the tweet.

[Step 2]

Read the tweet and select which of the four categories the tweet belongs to. Specific examples can be found on the next two pages. Please make your choice based on the feelings of the tweeter, not on how you feel after reading the tweet.

[Step 3]

After all 29 questions have been answered, please submit your responses. Your answers will be reviewed within 7 days. If your answers are approved, you will be paid.

[FINAL NOTICE]

Follow the instructions and review the tweets carefully. Some tweets may be ambiguous. Please select the one that seems closest. Do not spend more than 60 seconds on each tweet.

Five tweets with very clear opinions are included among the 29 tweets. If you correctly answer 4 or more of these 5 tweets and diligently complete the rest, your response will be accepted, and you will be rewarded.

Positive

A tweet is considered positive if it expresses a completely happy, satisfied, or optimistic attitude, or if the positive sentiment is the strongest in a mix of multiple sentiments.

Negative

If a tweet primarily expresses frustration, sadness, anger, or other negative attitudes, or if the negative emotion is the strongest in a mix of several emotions, it is considered a negative emotion.

Mixed

When a tweet expresses or implies both positive and negative emotions, and it is not clear which is dominant, it is considered mixed emotion. In this case, it is characterized by the simultaneous expression of conflicting emotions, such as joy and frustration.

Neutral

A neutral tweet expresses no emotion at all or very little. Examples of neutral emotions include stating a fact, asking a question, or seeking someone else's opinion.

仕事の詳細
【概要】

大学の研究者が、Twitter 上の意見を分類する AI モデルを開発しています。このプロジェクトでは AI の正確性を確認するため、人間がツイートの意見をラベル付けする必要があります。簡単な指示書を読んだ後、ツイートがポジティブ、ネガティブ、ミックス、またはニュートラル(中立)の中でどちらに当てはまるかを選択し、ラベル付けします。

【求める応募者】

応募は日本語を理解できる方であればどなたでも可能です。Twitter にある程度慣れている方ですとより望ましいです。

【報酬】

参加者は 29 件のツイートをラベル付けすることになります。ラベル付けには 15〜20 分程度かかると見込まれます。29 件のツイートにつき税込み 300 円の報酬が支払われます。指示書をよくお読みになり、理解された上でご参加ください。指示が理解されていないと判断された場合、データは受け入れられず、報酬はお支払いできかねます。

このプロジェクトへのご参加をご検討いただきありがとうございます。皆様のご参加が AI コミュニティにとって有益となります。

Job Details
[Summary]
University researchers are developing an AI model to classify opinions on Twitter. This project requires a human to label the opinions of tweets to ensure the accuracy of the AI. After reading a brief set of instructions, annotators will classify tweets by one of four categories: positive, negative, mixed, or neutral.

Applicants we are looking for:
We are looking for applicants who can understand Japanese and have some familiarity with Twitter.

Compensation:
Participants will be expected to label 29 tweets. You will be paid 300 yen plus tax for 29 tweets. Please read and understand the instructions carefully before participating. If it is determined that the instructions are not understood, your data will not be accepted and you will not be paid.

Thank you for considering your participation in this project. Your participation will be beneficial to the AI community.

Appendix 3: Prompts for Few-Shot Classification

This is an annotated example of the prompt for few-shot classification that was used to benchmark the generative AI models. The prompt was generated using a slot-fill mechanism that incorporated the train and test data.

Step 1: Define the problem and the labels.

```
The task is Sentiment Classification. Read the following Tweet
and classify the sentiment as Positive, Negative, Neutral, or
Mixed.



Labels:

Positive: Positive sentiment refers to expressions of
happiness, approval, or optimism in the text. While the text
might contain mixed sentiments, the dominant sentiment is
positive.

Negative: Negative sentiment refers to expressions of sadness,
disapproval, or pessimism in the text. The text primarily
conveys adverse emotions or opinions. While the text might
contain mixed sentiments, the dominant sentiment is negative.

Neutral: Neutral sentiment signifies the absence of any strong
emotions, opinions, or bias in the text. It represents factual
or objective statements.

Mixed: Mixed sentiment indicates the presence of both positive
and negative sentiment with no clear indication as to which is
dominant. It may be conflicting emotions or a balanced
presentation of pros and cons.
```

Step 2: Provide examples in the target format.

Input:

{"tweet_id": "1604327556198457344", "text": "[USER] クソだわ😂"}

{"tweet_id": "1609569777105915905", "text": "[USER] やり直してもまともになるかしら、、😳と思うぐらいぐっちゃぐちゃでした🥴"}

{"tweet_id": "1649261899270483968", "text": "鍼灸治療終わり、"}

{"tweet_id": "1567830275513815041", "text": "よくもまぁ･･･文春によると、北側一雄副代表、山口那津男代表もこうした問題行動を把握しながらも議員に再選させたそうな。 [URL]"}

What are the sentiments of these tweets? You must return your answer as JSON object. The output tweet_id must be identical to the input tweet_id.

Output:

{"tweet_id": "1604327556198457344", "label": "neutral"}

{"tweet_id": "1609569777105915905", "label": "neutral"}

{"tweet_id": "1649261899270483968", "label": "positive"}

{"tweet_id": "1567830275513815041", "label": "neutral"}

NOTE: Most models prefer to respond with natural language, which is difficult to parse. The JSON format is preferred for its ease of parsing, but the smaller models have trouble adapting. Modelling the exact input-output cycle enhances compliance. This cycle is repeated two more times using new examples.

Step 3: Elicit the response.

Input:

{"tweet_id": "1551175748592943104", "text": "[USER] お声かけ出来ず残念…\n 何となくお席は分かっていたんですが、お邪魔かなぁ…と思って、やめました…🙇‍♀️"}

{"tweet_id": "1511838555135303682", "text": "新入社員大量😳"}

{"tweet_id": "1632226900331622401", "text": "[USER] 血液は恐ろしい。"}

{"tweet_id": "1607607420238573568", "text": "なう。食べ納めです🍜 [URL]"}

What are the sentiments of these tweets? You must return your answer as JSON object. The tweet_id must be identical to the input tweet_id.

Output:

NOTE: The final cycle incorporates the test data. It breaks after the 'Output:' token, signaling the model to complete the response.

Step 1: Define the problem and the labels.

センチメントの分類 次の日本語のツイートを読んで、センチメントをポジティブ、ネガティブ、ニュートラルのいずれかに分類してください。この分類を行うには、日本語のニュアンスを正確に理解することが重要です。

Labels:

ポジティブ: 肯定的な感情とは、文章中の幸福、承認、楽観の表現を指す。テキストには様々な感情が含まれているかもしれないが、支配的な感情はポジティブである。

ネガティブ: 否定的な感情とは、文章中の悲しみ、不承認、悲観の表現を指す。主に不利な感情や意見を伝えている。

ニュートラル: 中立的な感情とは、文章に強い感情や意見、偏見がないことを意味する。事実や客観的な記述を表します。

Step 2: Provide examples.

入力:

{"tweet_id": "1644700447863881728", "text": "なんなら有名メーカーのやつ欲しいけどたけぇ"}

{"tweet_id": "1602199650005573632", "text": "[URL] アニソン帝王 俳優 水木一郎 肺がんで死去 74 歳\n\n 早すぎるよ😖💦😖💦 日本終わりだ！\n#水木一郎\n#日本終了"}

{"tweet_id": "1596763625175519232", "text": "[USER] そこならまだ補修が効くから大丈夫だ！"}

{"tweet_id": "1559838942220152832", "text": "通常営業 昼のみ ねw"}

これらのツイートの感情は何ですか？ 必ず JSON オブジェクトとして返してください。tweet_id は入力の tweet_id と同じでなければなりません。

出力：

{"tweet_id": "1644700447863881728", "label": "ニュートラル"}

{"tweet_id": "1602199650005573632", "label": "ニュートラル"}

{"tweet_id": "1596763625175519232", "label": "ネガティブ"}

{"tweet_id": "1559838942220152832", "label": "ポジティブ"}

Step 3: Elicit a response.

入力：

{"tweet_id": "1608433884324302848", "text": "隣のバカップルの女が男に注文を禁じられたにんにくラーメン、めちゃうまい\n\n#岐阜屋 #にんにく}

{"tweet_id": "1583005702725201920", "text": "#ポケモン GO \n ゲンガーをレイドでゲットした。\n 参加してるの 2 名だけだったけど意外と上手く行ったぜ。 [URL]"}

{"tweet_id": "1622589899303170049", "text": "[USER] お疲れさまでした✨\n 大丈夫？体調でもくずした？"}

{"tweet_id": "1577659957163368451", "text": "ようやく涼しくなってきて幸せ💕"}

これらのツイートの感情は何ですか？ 必ず JSON オブジェクトとして返してください。tweet_id は入力の tweet_id と同じでなければなりません。

出力：

Appendix 4: Core Functions of the Japanese WordPiece Tokenizer

This appendix illustrates the mechanism of the BertJapaneseTokenizer [20] from the Transformers library (Wolf, et al., 2020). It demonstrates five functions: *Normalization*, *Pre-Tokenization*, *Tokenization*, *Encoding*, and *Decoding*.

Example A4.1: Input

ねえ、兄ちゃん！ごちそうさま😋🍴 ｳﾏｶｯﾀﾖ～！😊

This sequence will be processed by the Japanese WordPiece tokenization pipeline.

Example A4.2: Normalization

Input: ねえ、兄ちゃん！ごちそうさま😋🍴 ｳﾏｶｯﾀﾖ～！😊

Output: ねえ、兄ちゃん!ごちそうさま😋🍴 ウマカッタヨ～!😊👍

Normalization *augments the vocabulary by transforming irregular characters to more standard forms. The BertJapneseTokenizer normalizes text using the Python library unicodedata[21]. This example shows how Japanese half-width characters are converted to their full-width forms.*

Example A4.3: Pre-Tokenization

Input: ねえ、兄ちゃん!ごちそうさま😋🍴 ウマカッタヨ～!😊👍

Output: ねえ, 、, 兄, ちゃん, !, ご, ちそう, さま, 😋🍴, ウマカッタヨ, ～ , ！ , 😊👍

Pre-tokenization separates the normalized sequence into words. *The JapaneseBertTokenizer uses MeCab for pre-tokenization.*

[20] https://huggingface.co/docs/transformers/model_doc/bert-japanese
[21] https://docs.python.org/3/library/unicodedata.html

Example A4.4: Tokenization

Input: ねえ，、，兄，ちゃん，！，ご，ちそう，さま，😋🍴，ウマカッタヨ，～ ， ！ ， 😊👍

Output: ねえ，、，兄，ちゃん，！，ご，ち，##そう，さま，[UNK]，ウマ，##カ，##ッタ，##ヨ，～，！，[UNK]

For words that are out of vocabulary, the WordPiece algorithm segments tokens into sub-words (yellow). If the sub-word vocabulary is insufficient to represent the word, it is substituted with the unknown token (green).

Example A4.5: Encoding

Input: ねえ，、，兄，ちゃん，！，ご，ち，##そう，さま，[UNK]，ウマ，##カ，##ッタ，##ヨ，～，！，[UNK]

Output: [2, 29821, 384, 914, 13579, 16, 438, 451, 14599, 14365, 1, 23626, 7134, 15302, 7177, 409, 16, 1, 3]

Encoding converts the tokenized sequence into a list of indices, directing the model to the appropriate word vectors in the embedding matrix for deeper encoding.

Example A4.5: Decoding

Input: [2, 29821, 384, 914, 13579, 16, 438, 451, 14599, 14365, 1, 23626, 7134, 15302, 7177, 409, 16, 1, 3]

Output: [CLS] ねえ 、 兄 ちゃん！ ご ちそう さま [UNK] ウマカッタヨ ～！ [UNK] [SEP]

Decoding converts a list of token into a string. This decoded representation includes the classification token, [CLS], which is for generating sequence embeddings, and the separation token, [SEP], which informs the model about sequence breaks.

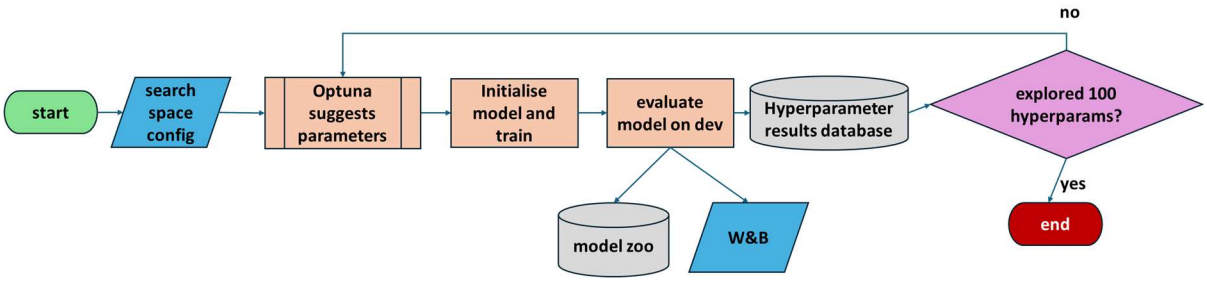
Appendix 5: Workflow for Fine-Tuning

Task fine-tuning requires its own session of hyperparameter optimization. Hyperparameter sweeps involved 100 trials with varying settings. Between trials, performance on the development set fed into a Bayesian search algorithm, leading to convergence on the optimal configuration. At the conclusion of the study, the candidate models are evaluated on the test set. Meanwhile, models are saved to disk in the 'model zoo'. Without proper management, hundreds of trials would quickly deplete even the extensive resources budgeted for this project. After evaluation, predictions were stored in confusion matrices, a compact data structure that allows for flexible analysis.

Two Python libraries were used to automate these processes:

- **Weights & Biases[22]:** This library provides an API that interfaces with an experiment tracking dashboard. Live training updates are posted to their web application, which offers many useful functionalities for post-processing.
- **Optuna[23]:** This library provides a wide range of tools for hyperparameter tuning. In this study, Optuna's sampler was utilized. Given an objective, a search field, and a database of past trials, Optuna suggests parameters for the next trial using a Bayesian search algorithm that favours configurations proximal to those that previously performed well.

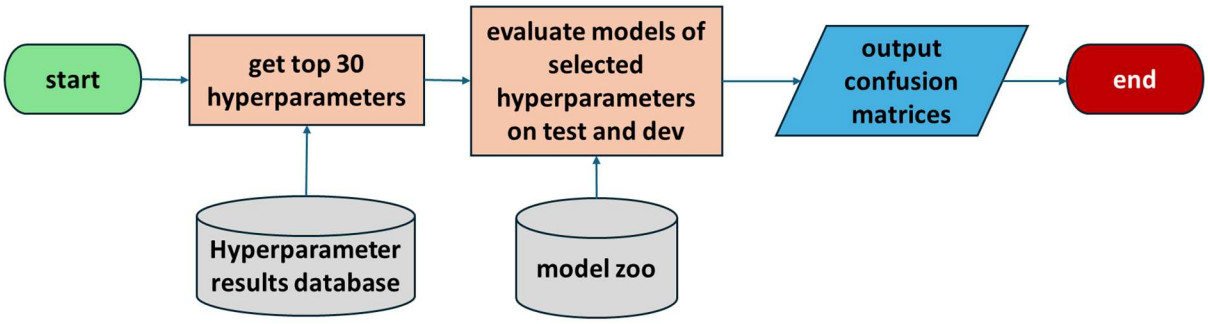
Figure A6.1: Flowchart of Hyperparameter Sweep



The hyperparameter is initialized with a search space configuration, which defines the variable parameters, and n trials, which is the number of configurations that the sweep will explore. On the first trial, Optuna suggests a configuration that is within the search field. A model and trainer are initialized with the suggested configuration. The model is trained and evaluated on the development set. That trained model is saved in the model zoo, and data is shuttled to Weights and Biases for visualization. The trial parameters and development results are saved to a database that interfaces with Optuna. Optuna suggests hyperparameters for the next trial, considering and results of previous trials. The suggestions will converge on a narrow range of hyperparameters where the model trained well. After n trials, the sweep concludes.

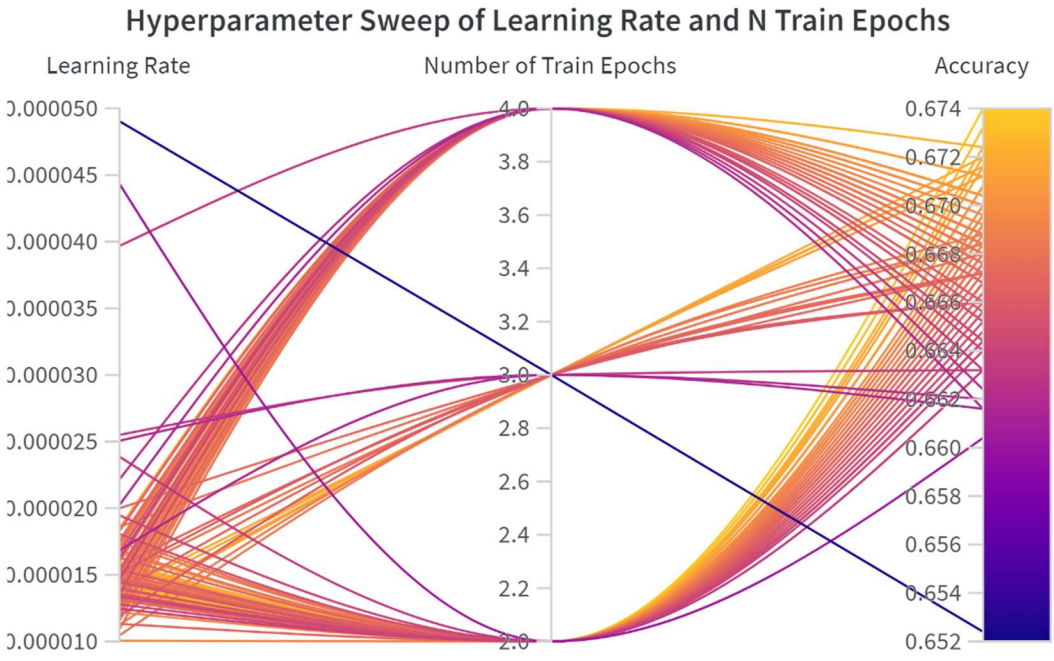
[22] https://wandb.ai/home
[23] https://optuna.org/

Figure A6.2: Flow Chart of Evaluation



After the hyperparameter sweep, the models are evaluated together. The top 30 trials, as measured by their performance on the development set, are loaded for evaluation. The models are evaluated on both the development set and the test set. Comparing scores on the development set with the record on database validates that the pipeline is intact. The test set provides the final measure to compare models. Confusion matrices are generated to retain a record of model performance. Sub-optimal models are deleted from the model zoo.

Figure A6.2: Visualization of Hyperparameter Sweep with Weights & Biases



During the hyperparameter sweep, results are sent to the Weights & Biases web interface for visualization. This sweep demonstrates how Optuna converges on the optimal parameters, achieving the best results with a lower learning rate and sampling more heavily in that area. Training remains robust across multiple epochs, but the best results are obtained with two epochs.

Appendix 6: Instructions for Evaluating Masked Token Predictions

These are the instructions provided to the native Japanese annotators for appraising the quality of masked token predictions.

一部（一文字または一つの言葉）が隠れた文章を AI に与えると、AI が言葉を予測し、その間違いから学びます。時にいい予測をしますが、時には変な予測もします。

以下は予測の例です。予測した言葉が「OK」であったか、「MISTAKE」であったかを回答いただきたいと思います。

Sentence(元の文章):

おうちカフェオープン。ちょっとかためプリン。夫が氷**コーヒー**を作ってくれました。

Masked(一部が隠された文章):

おうちカフェオープン。ちょっとかためプリン。夫が氷**[MASK]**を作ってくれました。

Prediction 1（予測 1）:

おうちカフェオープン。ちょっとかためプリン。夫が氷餅を作ってくれました。

OK（氷餅というスイーツありそうですね）

Prediction 2（予測 2）:

おうちカフェオープン。ちょっとかためプリン。夫が氷水を作ってくれました。

OK（なんで氷水？という感じですが氷水はまぁあってもおかしくはないかな）.

Prediction 3（予測 3）:

おうちカフェオープン。ちょっとかためプリン。夫が氷柱を作ってくれました。

MISTAKE（おうちカフェで氷柱は作らないですよね）.

Prediction 4（予測 4）:

おうちカフェオープン。ちょっとかためプリン。夫が氷魚を作ってくれました。

MISTAKE

“Masked Language Modelling”

One of the techniques to train AI is called “Masked Language Modelling”.

You give the model a sentence with one word masked. The model predicts the word, and it learns from its mistakes.

I will show you some predictions, and I want you to tell me if it is **OK** or a **MISTAKE.**

Sentence:

Home café opened. A little bit of pudding. My husband made iced coffee.

Masked:

Home café opened. A little bit of pudding. My husband made ice [MASK].

Prediction 1:

Home café opened. A little bit of pudding. My husband made iced cakes.

OK: Iced cakes sounds realistic.

Prediction 2:

Home café opened. A little bit of pudding. My husband made iced water.

OK: This is strange, but it is acceptable.

Prediction 3:

Home café opened. A little bit of pudding. My husband made icicles.

MISTAKE: You don't make icicles at your home café.

Prediction 4:

Home café opened. A little bit of pudding. My husband made “ice fish”.

MISTAKE: “Ice Fish” this is a preserved fish product typical of Hokkaido. Not acceptable.