BRCC and SentiBahasaRojak: The First Bahasa Rojak Corpus for Pretraining and Sentiment Analysis Dataset

Nanda Putri Romadhona[†] Sin-En Lu[†] Bo-Han Lu[†] Richard Tzong-Han Tsai[†‡*]

[†]Department of Computer Science and Information Engineering,
National Central University, Taiwan
[‡]Center for GIS, Research Center for Humanities and Social Sciences,
Academia Sinica, Taiwan
nandadona61@gmail.com
{alznn, lu110522028, thtsai}@g.ncu.edu.tw

Abstract

Code-mixing refers to the mixed use of multiple languages. It is prevalent in multilingual societies and is also one of the most challenging natural language processing tasks. In this paper, we study Bahasa Rojak, a dialect popular in Malaysia that consists of English, Malay, and Chinese. Aiming to establish a model to deal with the code-mixing phenomena of Bahasa Rojak, we use data augmentation to automatically construct the first Bahasa Rojak corpus for pre-training language models, which we name the Bahasa Rojak Crawled Corpus (BRCC). We also develop a new pre-trained model called "Mixed XLM". The model can tag the language of the input token automatically to process code-mixing input. Finally, to test the effectiveness of the Mixed XLM model pre-trained on BRCC for social media scenarios where code-mixing is found frequently, we compile a new Bahasa Rojak sentiment analysis dataset, SentiBahasaRojak[1], with a Kappa value of 0.77.

1 Introduction

Code-mixing is common in multilingual societies (Bukhari et al., 2015). People tend to use one primary language for grammar and scripting (Lal et al., 2019), and other languages as auxiliary. Code-mixing is commonly found on social media, such as Facebook, Twitter, or any other microblog services.

Malaysia reflects a multilingual society that considers Malay to be the national language but uses mixed languages in daily life. Bahasa Rojak (Bukhari et al., 2015) is one of the code-mixing examples that combines Malay and English into a certain level of language structure. Bahasa Rojak or Malaysian English (Vollmann and Wooi, 2019) is often mixed with Chinese because the ethnic Chinese population in Malaysia is quite large. Bakar and Mazzalan (2018) show that many users in Malaysia use Bahasa Rojak on Facebook. We can find many code-mixing combinations like Malay-English, Hindi-English, Spanish-English, and others on social media. People not only express their feelings on social media, but also exchange information such as the latest news from their hometowns or countries, and financial topics are also popular. Therefore, natural language processing (NLP) studies are increasingly focused on code-mixing (Thara and Poornachandran, 2018).

One of the most advanced trends in natural language processing is to prepare a large unlabeled corpus to pre-train a language model for representing the input text. The source of this unlabeled corpus is usually Wikipedia (Qiu et al., 2020). However, Wikipedia does not have any Bahasa Rojak pages, which hinders the training of a pre-trained model that can represent Bahasa Rojak input texts.

As a result, in this study, we employ data augmentation to automatically construct a new Bahasa Rojak code-mixing corpus, called *BRCC*, for pre-training language models. To find the best way of exploiting the corpus, we not only pre-train language models including BERT and XLM on BRCC, but also revise the original XLM model to make it able to handle code-mixing input text. The revised model is called *Mixed XLM*. As we mentioned, Bahasa Rojak is most frequently used in social media texts. Hence, we compile the first Bahasa Rojak sentiment analysis dataset, called *SentiBahasaRojak*, to evaluate each language model's performance and reflect its ability to represent Bahasa Rojak input texts.

*Corresponding author.

[1]Both BRCC and SentiBahasaRojak are available at https://data.depositar.io/dataset/brcc_and_sentibahasarojak

Proceedings of the 29th International Conference on Computational Linguistics, pages 4418–4428
October 12–17, 2022.

2 Related Work

2.1 Code-Mixing

Code-mixing is a term in mixed language research, representing a common phenomenon in a multilingual society. Code-mixing means that a single sentence or a single utterance contains different languages (Ho et al., 2007).

Pratapa et al. (2018b) compares three existing bilingual word embeddings and a novel method based on the skip-gram language model. They found that bilingual word embeddings obtained from a mixture of two languages rather than from multilingual monolingual texts are more suitable for code-mixing tasks. Lal et al. (2019) proved that the traditional method, which only takes surface and semantic features into account, is not effective for code-mixing sentiment analysis, so they proposed a new method called "demixing". Choosing English-Hindi as the target of code-mixing research, they used a convolutional neural network to generate sub-words and constructed a dual encoder network composed of two parallel BiLSTMs.

A recent code-mixing paper related to this paper is Qin et al. (2020), in which the authors use a code-mixing corpus to fine-tune mBERT. They also increase the size of the code-mixing corpus through data augmentation. Their method aligns the representations of the source language and multiple target languages by using contextual information. Even though there are multiple target languages, the model only needs to be pre-trained once.

2.2 Code-Mixing and Sentiment Analysis of Social Media in Malaysia

Malay is the official language of Malaysia. In the 19th century, under British colonial rule, English had a profound influence on Malay. English replaced the original writing system completely with Latin script, and many words of economic, political and technical fields in modern Malay are borrowed from English. Due to the large population of Chinese people in Malaysia, there are many Chinese schools and companies leading the spread of Chinese.

Because of cultural blending, Malaysian citizens often use code-mixing language. The phenomena have led to the emergence of "Bahasa Rojak", a new language in Malaysia that combines English, Malay, and Chinese words and structures, and often appears on social media such as Facebook and Twitter. Some Chinese users (Shafiee et al., 2019) introduce Chinese words in text-based communication. For example, Table 1 shows comments in three financial forums. These comments are presented in the form of English or code-mixing.

So far, research on sentiment analysis on social media used by Malaysians is still quite limited. Al-Saffar et al. (2018) use an emotional dictionary construction method to obtain a set of predefined features (emotional words). These features are used to build a machine-learning classifier model to determine the sentiment polarity of the given input social media text. There are also a few labeled pure Malay sentiment analysis datasets (Husein, 2018), but to the best of our knowledge, there is no labeled Bahasa Rojak dataset.

2.3 Pre-trained Language Models

BERT (Devlin et al., 2019) is a pre-trained encoder model based on the Transformer architecture. Since it is bidirectional, context semantics can be considered. Usually, we use a huge corpus, such as Wikipedia, to pre-train BERT. Since its launch in 2018, BERT has achieved leading performance on many sentence-level and token-level natural language processing tasks, such as question answering (Rajpurkar et al., 2016; Joshi et al., 2017), machine translation (McCann et al., 2017), and sentiment analysis (Socher et al., 2013).

RoBERTa (Liu et al., 2019) was made with some modifications based on the BERT model. RoBERTa removes the pre-training task of next sentence prediction, uses dynamic masking in pre-training, and adopts a larger byte-level BPE as its text encoding method. RoBERTa has more parameters than BERT and uses a larger corpus for pre-training to obtain better performance.

XLM-R (Conneau et al., 2020) is a transformer-based cross-lingual model, which combines XLM (Conneau and Lample, 2019) and RoBERTa. The difference with the monolingual XLM is that XLM-R is pre-trained on corpus containing multiple languages, so it can represent sentences containing multiple languages. The basic XLM-R uses Masked Language Model (MLM) as the pre-training task. When a bilingual parallel corpus is available, additional pre-training with the Transla-

Sentence	Translate	Language
dont surprise..Armada will touch below 40sen..	No need to be surprised, Armada will touch under 40 cents.	Manglish
The bad news is finished and the good things have come...charting showing good sign 物极必反	The bad news is finished and the good things have come...charting showing good sign things must be reversed.	Manglish + Chinese
Bukan ex Umno saja, tapi x der integrity, penipu, senyum kambing yang bodoh!	Not just ex Umno, but not integrity, liar, stupid goat smile!	Bahasa Rojak

Table 1: Sample comments on a financial forum in Malaysia

Short form	Original word
a.n.	atas nama
awk	awak
bsh	bodoh
bkn	bukan
bln	bulan

Table 2: Malay short form words and corresponding original words

Language	Passages	Tokens
Code-mixing	2M	62,703,287
Malay	2M	60,519,134
English	2M	75,032,902

Table 3: Statistics on the corpora used for pre-training, including the total number of subwords in each language based on BPE Tokenizer segmentation.

tion Language Model (TLM) task can be used to improve performance.

2.4 Transfer Learning

Recently, transfer learning methods have become popular in natural language processing, especially for low-resource tasks. Transfer learning improves failed steps by transferring the resources of related tasks, languages, or domains of high-resource source settings to low-resource target settings.

Transfer learning is not a new method to solve NLP tasks, since it has long been applied on many NLP tasks, such as latent semantic analysis (Deerwester et al., 1990), Brown clusters (Brown et al., 1992), and pre-trained word embeddings (Mikolov et al., 2013). Ruder et al. (2019) have created a taxonomy that makes it easier for researchers to design solutions based on transfer learning methods. For example, if the target task is the same as the source task and there is only annotated data of the source task, the domain adaptation method can be applied. For low-resource language tasks, the usual practice is to train on the annotated data of high-resource source language and apply to low-resource target languages, such as Farra (2019).

Note that the main purpose of adopting transfer learning methods in cross-language tasks is to transfer lexical knowledge across languages, that is, to establish a cross-language word embedding model.

3 Corpus Compilation

In this study, we mainly deal with three languages: English, Malay, and Bahasa Rojak. For each language, we construct a corpus to pre-train each language model and a sentiment analysis dataset to evaluate each language model's performance. The reason for choosing sentiment analysis is that the frequency of code-mixing on social media texts is relatively high (Thara and Poornachandran, 2018). We will continue to discuss more details about the reason for choosing sentiment analysis in 3.4.

3.1 Data Preprocessing

We use common rules to pre-process our English microblogging corpora, including removing noise or unnecessary characters, tags, URLs, certain symbols, etc. The same rules are also used to pre-process Malay microblogging corpora.

However, there are still differences between English and Malay microblog corpora. In Malay, words or sentences are more often abbreviated into shorter forms, such as dialects, word abbreviations, grammatical neglect, and many more. For example, it is common that the word *because* is written as *bcz*, which causes high noise and a distinct text structure. The short form manner in Malay becomes a serious issue in Malay's NLP research (Ariffin and Tiun, 2020). Also, we have the concern that if the short-form text in Malay is not regularized, the vocabulary size will be too large to train the model and increases the cost of training. Past studies show that normalizing such short-form

Algorithm 1: Bahasa Rojak's Data Augmentation

Input :
Source languages : $sl \leftarrow \{en, ms\}$;
Target languages : $tl \leftarrow \{en : [ms, zh], ms : [en, zh]\}$;
Set of source language sentences : $S_{sl} = \{s^n\}_{n=1}^{N}$;
Replace ratio : $[\alpha, \beta, \gamma]$, Type of replaced word : $rw\_list = [V, N, Adj]$;

Output :
Set of Code-Mixing sentences : $T = \{t^n\}_{n=1}^{N}$;

for i *in* $1...N$ **do**
  $count = 0$;
  $t^i \leftarrow s^i$; // Initialize code-mixed Sentence
  if $random() \leq \alpha$ **then**
    // phrase extraction and syntax analysis, return list
    $phrases \leftarrow get\_phrase(rw\_list, s^i)$;
    while $count \leq \beta * len(phrases)$ **do**
      $rid = random\_int(0, len(phrases))$;
      $rw = random(phrases[rid])$;
      $trans\_phrase \leftarrow Translate(rw, random(tl\{sl\})$;
      $t^i \leftarrow Replace\_and\_Aligment(phrases, trans\_phrase, rid)$;
      $count \leftarrow count + 1$;
  else
    if $random() \leq \gamma$ **then**
      $t^i \leftarrow Translate(s^i, random(tl\{sl\})$ // Translate complete sentences

Phrase Name	Pattern
Noun Phrase	{<DET|ADJ|NOUN.*>+ <DET|ADJ|NOUN.*>+}
Prepositional Phrase	{<ADP> <NP>}{<ADP> <PROPN>}
Verb Phrase	{<VERB.*> <NP|PP|CLAUSE|ADP>+$} {<VERB*> <NOUN*>} {<PART> <VERB>}

Table 4: For the regular expression patterns, we use the NLTK parser to identify noun, prepositional and verb phrases.

text increases data quality and has a positive effect on NLP research (Samsudin et al., 2013; Saloot et al., 2014; Kassim et al., 2020). Chekima and Alfred (2017) collected some Malay SMS rules to normalize short form words, and we continue to add some rules. For more examples, please refer to Table 2.

3.2 Bahasa Rojak Crawled Corpus (BRCC)

In order to pre-train a model that works in both monolingual and code-mixing environments, we first construct monolingual corpora and then derive a code-mixing corpus from them. For English and Malay corpora, we crawl English and Malay pages from Wikipedia. We mainly use our data augmentation method to generate the Bahasa Rojak code-mixing corpus, which is called BRCC (Bahasa Rojak Crawled Corpus). Each of these three corpora has 2 million passages. Table 3 shows the detailed information of the three corpora.

To generate BRCC through data augmentation, we first scrape 93,584 Bahasa Rojak passages from the Malaysia Bursa forum. In order to expand the Bahasa Rojak corpus, we modify the CoSDA-ML method to generate 2 million Bahasa Rojak passages from our English and Malay corpora. The main difference is that CoSDA-ML randomly selects words and translates them into a specific target language, while our method parses the sentence to identify phrases and then randomly selects the phrases to be translated. Algorithm 1 explains the details of our data augmentation method. For a detailed description, please refer to A.1.

Take the phrase "the book in your schoolbag" for example. Suppose we randomly choose a word to translate into another language; if "the" is selected, because of the lack of context, "the" is not suitable to be translated by itself. To alleviate this problem, our method first analyzes sentences with part-of-speech (POS) tags, and identifies noun phrases, prepositional phrases, and verb phrases through patterns composed of POS tags. Table 4 shows the regular expression patterns used with the NLTK

Original Sentence	Augmentation Sentence
Di sepanjang pesisirnya terdapat teluk dan tanjung yang berpontensi dimajukan sebagai kawasan pelancongan	Along the coast terdapat teluk dan tanjung yang berpontensi dimajukan sebagai tourist areas
He was previously offered X Men membership but he declined opting instead to work at the Muir Island research center Polaris Havok s long time lover and also a former X Man who can control magnetism	He was previously offered X Men membership but he declined opting instead to work at pusat penyelidikan Pulau Muir Polaris Havok s 長期戀人 and also bekas X Man who can 控制磁

Table 5: Samples of data augmentation results

(Bird and Loper, 2004) parser to identify the three types of phrases mentioned above. These patterns have been confirmed by native Malay speakers. All-caps words denote POS tags, angle brackets denote sub-patterns, and the rest of the symbols are used the same way as they are in regular expressions. Taking a simplified noun phrase pattern *{<ADJ>+ <NOUN>+}* as an example, this means that if the parser finds at least one adjective (ADJ) followed by at least one noun (NOUN), it finds a noun phrase.

Lastly, we randomly select a phrase to translate into the target language. The purpose of the modification is to choose "the book" instead of "the" or "book". Table 5 shows the sentences generated by our data augmentation method.

3.3 BRCC Quality

To evaluate the quality of BRCC, we conduct a test inspired by the Turing Test on two native Malay speakers. We randomly sample 500 sentences each from the BRCC corpus (using our data augmentation method to synthesize Bahasa Rojak sentences) and the KLSE forum (klse.i3investor.com), where Bahasa Rojak frequently appears. After mixing the two, we ask two native speakers to judge whether it is Bahasa Rojak, sentence by sentence, and if so, the sentence is labeled as positive, otherwise negative. As shown in Table 6, we get similar positive ratios in BRCC and KLSE, which indicates that most of the synthesized Bahasa Rojak sentences in BRCC are considered to be real Bahasa Rojak.

3.4 Sentiment Analysis Datasets

To verify that the Bahasa Rojak code-mixing corpus generated by our data augmentation method can be used to pre-train a code-mixing language model, we choose a natural language processing task for testing. According to Thara and Poornachandran (2018), people tend to using social media to share their opinions and thoughts, making code-mixing texts common on all kinds of social platforms in a multilingual society. We choose sentiment analysis of social media texts as our natural language processing task. We construct the first Bahasa Rojak sentiment analysis dataset, named SentiBahasaRojak, to evaluate the code-mixing model's performance on Bahasa Rojak sentiment analysis. This dataset contains three domains: product review, movie review, and stock review. To determine whether this code-mixing model can perform well on Bahasa Rojak and remain accurate in English or Malay, we have also compiled English and Malay datasets containing the same three domains.

3.4.1 English Sentiment Analysis Dataset

For English, we crawl product reviews from Kaggle[2] and use the IMDB dataset (Maas et al., 2011) as movie review data. As for stock reviews, we choose SemEval 2017 task 5 subtask 1 (Kar et al., 2017) and StockTwits[3]. The former is composed of financial microblogging data. Each post has been labeled with a value of -1 to 1, which corresponds to the most bearish (negative) to the most bullish (positive). To match other datasets, we convert the values into binary labels. StockTwits is a microblogging platform focusing on stock market discussions and supported by Twitter. According to StockTwits restrictions, users must label their posts as bullish or bearish. We collected all the posts of ten companies on StockTwits from 2016 to 2020.

3.4.2 Malay Sentiment Analysis Dataset

For Malay, we use the product review dataset and the movie review dataset from the Malay dataset (Husein, 2018). As for the stock review dataset, we hired experts to manually translate the dataset

[2] https://www.kaggle.com/bittlingmayer/amazonreviews

[3] https://stocktwits.com/

	BRCC Positive	BRCC Negative	KLSE Positive	KLSE Negative
Participant 1	408 (81.6%)	92 (18.4%)	438 (87.6%)	62 (12.4%)
Participant 2	442 (88.4%)	58 (11.6%)	377 (75.4%)	123 (24.6%)

Table 6: BRCC and KLSE each with 500 sentences assessed by native speakers (Participants 1 and 2). Positive means that the native speaker thinks the sentence is fluent and conforms to Bahasa Rojak's grammar, otherwise it is negative.

Dataset	**# of post**		
	Product Review	**Movie Review**	**Stock Forum**
English	2600	2600	1106
Malay	893	699	1106
sentiBahasaRojak	893	699	693

Table 7: The statistics of sentiment analysis datasets

Language	**# of tokens**
Malay (ms)	60,628,280
English (en)	1,867,773
Chinese (zh)	1,553,360
Undefined (other)	438,937

Table 8: BRCC's language auto-tagging statistics

from task 5 subtask 1 of SemEval 2017 to Malay.

3.4.3 Bahasa Rojak Sentiment Analysis Dataset (SentiBahasaRojak)

To the best of our knowledge, there is no public Bahasa Rojak sentiment analysis dataset. In this research, we employ different methods to construct three datasets of product reviews, movie reviews, and stock reviews, which collectively are called the SentiBahasaRojak sentiment analysis dataset. For product and movie reviews, since there are already publicly available Malay datasets, we use the data augmentation method mentioned in Algorithm 1 to generate Bahasa Rojak code-mixing datasets based on the Malay datasets.

For stock reviews, we intend to test the effectiveness of our Mixed XLM method on real data, so we crawl posts from Malaysia's financial and stock market websites, such as I3investor[4], and hired five native Malaysian experts who can read and write in Bahasa Rojak to manually annotate these posts. Experts must be able to distinguish the difference between bullish (positive) and bearish (negative). These five experts first annotated all the posts, and then majority voting was carried out to determine the final label of each post. The kappa value of the Bahasa Rojak stock review dataset is 0.77, which means that the annotations of these five experts have substantial agreement (Kasmuri and Basiron, 2019). Table 7 shows the number of posts in English, Malay, and Bahasa Rojak in the three domains.

[4] https://klse.i3investor.com/jsp/scl/community.jsp

4 Experiments and Results

In this section, we will explain our proposed method for dealing with code-mixing data. BERT and XLM are chosen as the baseline language models for comparison, and use a variety of sentiment analysis task datasets to evaluate each model.

We use the movie reviews, product reviews, and stock market comments described in the previous section as the experimental datasets. We will also report the accuracy and F1 score of each model for each dataset.

4.1 Baseline Model

We use the code-mixing data constructed in the previous section to pre-train the BERT and XLM models from scratch. Based on the results obtained from preliminary experiments, we remove the next sentence prediction (NSP) task when pre-training the BERT model.

We use the Masked Language Model (MLM) task to pre-train our baseline models. Due to the limitation of computing resources, our configuration is six layers, eight heads, 512 embedding dimensions, and the learning rate fixed at 2e-5. We use the Adam optimizer and adopt the early stopping method to terminate the training process.

4.2 Mixed XLM

In this work, we propose a new model called Mixed XLM. The main difference from vanilla XLM is that Mixed XLM automatically recognizes the language of each input token and handles code-mixing input, as shown in the Figure 1. In the Mixed XLM for Bahasa Rojak, we develop a language tagging algorithm to label the language of each word, as

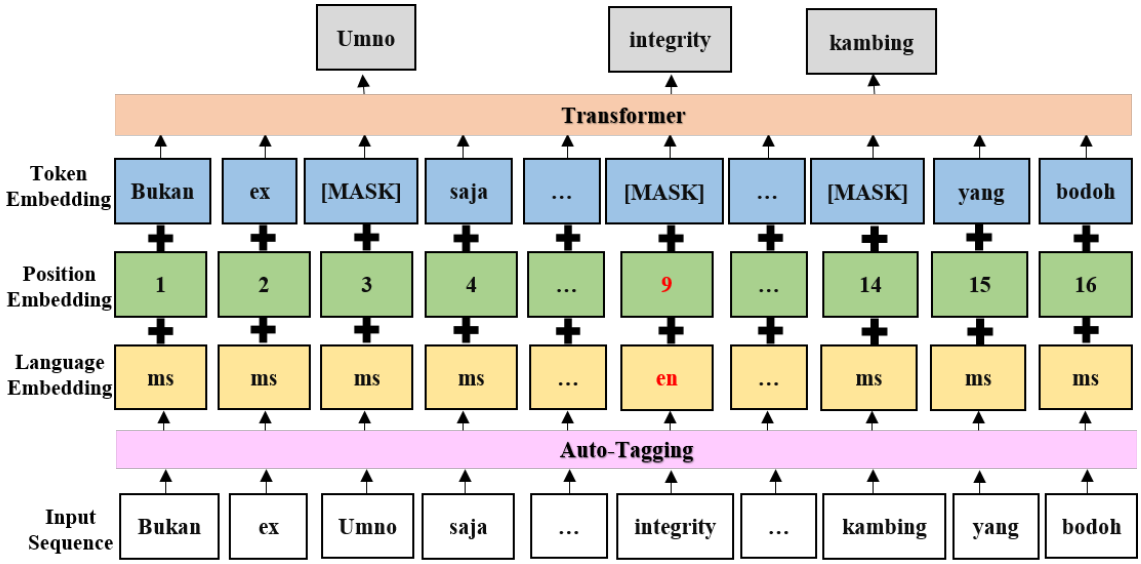

Figure 1: Input representation of Mixed XLM

shown in Algorithm 2. For each token *t*, the algorithm searches which language dictionary it is in. Suppose *t* is found in the dictionary of language *l*, then *t* is labeled as language *l*. Some words can be found in both Malay and English dictionaries, and language tagging labels them as Malay because these words are loanwords from English. We evaluated the language tagging module as having an accuracy of 0.973. The number of tokens for each language in our BRCC corpus after language tagging is shown in Table 8.

Algorithm 2: Language Tagging

Input :
Vocabulary of Malay words : V_{ms};
Vocabulary of Chinese words : V_{zh};
Vocabulary of English words : V_{en};
Input sentence : S_{cm};
Words in sentence : $S_{cm} = \{w^{(n)}\}_{n=1}^{N}$;

Output :
Language tagging : $langTag[N]$;

for $w^{(n)}$ *in* S_{cm} **do**
  if $w^{(n)}$ *in* V_{ms} **then**
    $langTag[n] = ms$
  else if $w^{(n)}$ *in* V_{zh} **then**
    $langTag[n] = zh$
  else if $w^{(n)}$ *in* V_{en} **then**
    $langTag[n] = en$
  else
    $langTag[n] = other$
  end
end

Finally, in the setting of our Mixed XLM, which is the same as BERT and XLM, we use six transformation layers, eight head layers, and 512 embedding dimensions, and use the masked language model as the pre-training task.

4.3 Evaluation of Pre-trained Language Models

We use the sentiment analysis task to evaluate our pre-trained language models. We also use 10-fold cross-validation to strengthen the credibility of the results. Table 9 shows the performance of our proposed models and baseline models, fine-tuned on SentiBahasaRojak. Remember that there are three domains in our SentiBahasaRojak, including product reviews, movie reviews, and stock market forums.

In Table 9, the best performing baseline model is XLM (EN-MS), which achieves 0.698 and 0.637 in accuracy ACC and F1 score, respectively. As for our proposed model Mixed XLM, it scores 0.718 and 0.666 when using only the Code-Mixing dataset (CM) for training. If the entire dataset (EN-MS-CM) is used for training Mixed XLM, 0.745 and 0.705 can be achieved, meaning it outperforms the best baseline model XLM (EN-MS) by 0.047 and 0.068. As shown in Table 9, we can observe that in the code-mixing sentiment analysis task, our proposed Mixed XLM achieves the best performance in all three datasets.

In addition, experiments are conducted on En-

Model	Product Review		Movie Review		Stock Market		Avg.	
	acc	f1	acc	f1	acc	f1	acc	f1
mBERT (CM)	0.652	0.603	0.661	0.653	0.563	0.496	0.625	0.584
mBERT (EN-MS-CM)	0.653	0.651	0.631	0.576	0.571	0.562	0.618	0.596
XLM (EN-MS)	0.658	0.592	0.764	0.751	0.672	0.568	0.698	0.637
Mixed XLM (CM)	0.703	0.671	0.761	0.746	0.690	0.581	0.718	0.666
Mixed XLM (EN-MS-CM)	**0.718**	**0.696**	**0.812**	**0.803**	**0.706**	**0.615**	**0.745**	**0.705**

Table 9: Results of different models on SentiBahasaRojak

Model	Product Review		Movie Review		Stock Market		Avg.	
	acc	f1	acc	f1	acc	f1	acc	f1
mBERT (CM)	0.801	0.793	0.691	0.663	**0.717**	**0.773**	0.736	0.743
mBERT (EN-MS-CM)	0.803	0.794	0.755	0.745	0.702	0.771	0.753	0.770
XLM (EN-MS)	0.813	0.812	0.701	0.689	0.675	0.746	0.730	0.749
Mixed XLM (CM)	0.807	0.804	0.792	0.771	0.643	0.712	0.747	0.762
Mixed XLM (EN-MS-CM)	**0.823**	**0.826**	**0.813**	**0.787**	0.677	0.743	**0.771**	**0.785**

Table 10: Results of different models on English

Model	Product Review		Movie Review		Stock Market		Avg.	
	acc	f1	acc	f1	acc	f1	acc	f1
mBERT (CM)	0.813	0.802	0.782	0.756	0.690	0.756	0.762	0.771
mBERT (EN-MS-CM)	0.815	0.743	0.780	0.782	0.683	0.765	0.759	0.763
XLM (EN-MS)	0.823	0.802	0.751	0.744	0.683	0.742	0.752	0.763
Mixed XLM (CM)	0.824	0.805	0.783	0.764	0.661	0.736	0.756	0.768
Mixed XLM (EN-MS-CM)	**0.828**	**0.818**	**0.805**	**0.785**	**0.696**	**0.769**	**0.776**	**0.791**

Table 11: Results of different models on Malay

glish and Malay monolingual datasets. We fine-tune all language models on monolingual datasets. Tables 10 and 11 show that our Mixed XLM model pre-trained on all corpora including BRCC achieves the highest average score on each language, which demonstrates the robustness of our approach.

5 Conclusion

In this paper, for Bahasa Rojak, we build a corpus called BRCC to pre-train Bahasa Rojak's language model, and compile a sentiment analysis dataset called SentiBahasaRojak. BRCC and SentiBahasaRojak are the first resources available for Bahasa Rojak in this area. We also propose a new pretrained model, Mixed XLM, which not only achieves the best performance on code-mixing data, but also maintains performance on monolingual data.

Our new Bahasa Rojak corpus is generated by our new data augmentation algorithm that recognizes three types of phrases in sentences and randomly selects some of those three phrases for translation to generate Bahasa Rojak sentences.

Our proposed Mixed XLM model is able to label input tokens to deal with code-mixing phenomena. As long as the Mixed XLM model is pre-trained on a code-mixing corpus, it can be used in downstream tasks containing code-mixing sentences, just as in this study, the Mixed XLM model was used in Bahasa Rojak's Sentiment Analysis.

Finally, we evaluate the Mixed XLM model pre-trained on BRCC through the sentiment analysis task on three different language settings (English, Malay, Bahasa Rojak). The sentiment analysis task includes three domains. The results show our Mixed XLM model achieves the best performance in all domains. In the monolingual setting experiment, Mixed XLM also achieves comparable performance, which proves the robustness of the model and the effectiveness of BRCC.

References

Ahmed Al-Saffar, S. Awang, Hai Tao, N. Omar, Wafaa Al-Saiagh, and M. Albared. 2018. Malay sentiment analysis based on combined classification approaches and senti-lexicon algorithm. *PLoS ONE*, 13.

SNAN Ariffin and Sabrina Tiun. 2020. Rule-based text normalization for Malay social media texts. *Interna-*

tional Journal of Advanced Computer Science and Applications, 11(10):156–162.

Mohd Syuhaidi Abu Bakar and Aliffluqman Mohd Mazzalan. 2018. Aliran pertuturan bahasa rojak dalam kalangan pengguna facebook di malaysia. *e-Academia Journal*, 7(1).

Steven Bird and Edward Loper. 2004. NLTK: The natural language toolkit. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, pages 214–217, Barcelona, Spain. Association for Computational Linguistics.

Peter F. Brown, Vincent J. Della Pietra, Peter V. deSouza, Jenifer C. Lai, and Robert L. Mercer. 1992. Class-based *n*-gram models of natural language. *Computational Linguistics*, 18(4):467–480.

N. A. Bukhari, A. Anuar, Khairunnisa Mohad Khazin, and T. T. Aziz. 2015. English-Malay code-mixing innovation in Facebook among Malaysian university students. *Researchers World*, 6:01–10.

Khalifa Chekima and Rayner Alfred. 2017. Sentiment analysis of Malay social media text. In *International Conference on Computational Science and Technology*, pages 205–219. Springer.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R.A. Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science 41*, pages 391–407.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Noura Farra. 2019. *Cross-Lingual and Low-Resource Sentiment Analysis*. Columbia University.

Björn Gambäck and Amitava Das. 2016. Comparing the level of code-switching in corpora. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1850–1855, Portorož, Slovenia. European Language Resources Association (ELRA).

Souvick Ghosh, Satanu Ghosh, and Dipankar Das. 2017. Complexity metric for code-mixed social media text. *Computación y Sistemas*, 21.

Deepak Gupta, Asif Ekbal, and Pushpak Bhattacharyya. 2020. A semi-supervised approach to generate the code-mixed text using pre-trained encoder and transfer learning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2267–2280, Online. Association for Computational Linguistics.

Judy Woon Yee Ho et al. 2007. Code-mixing: Linguistic form and socio-cultural meaning. *The International Journal of Language Society and Culture*, 21(7):1–8.

Zolkepli Husein. 2018. Malay-dataset. `https://github.com/huseinzol05/Malay-Dataset`.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Sudipta Kar, Suraj Maharjan, and Thamar Solorio. 2017. Ritual-uh at semeval-2017 task 5: Sentiment analysis on financial data using neural networks. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 877–882, Vancouver, Canada. Association for Computational Linguistics.

Emaliana Kasmuri and Halizah Basiron. 2019. Building a Malay-English code-switching subjectivity corpus for sentiment analysis. *Int. J. Advance Soft Compu. Appl*, 11(1).

Mohamad Nizam Kassim, Shaiful Hisham Mat Jali, Mohd Aizaini Maarof, Anazida Zainal, and Amirudin Abdul Wahab. 2020. Enhanced text stemmer with noisy text normalization for Malay texts. In *Smart Trends in Computing and Communications*, pages 433–444, Singapore. Springer Singapore.

Yash Kumar Lal, Vaibhav Kumar, Mrinal Dhar, Manish Shrivastava, and Philipp Koehn. 2019. De-mixing sentiment from code-mixed text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 371–377, Florence, Italy. Association for Computational Linguistics.

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

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6297–6308, Red Hook, NY, USA. Curran Associates Inc.

Tomas Mikolov, Kai Chen, Greg S. Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space.

Adithya Pratapa, Gayatri Bhat, Monojit Choudhury, Sunayana Sitaram, Sandipan Dandapat, and Kalika Bali. 2018a. Language modeling for code-mixing: The role of linguistic theory based synthetic data. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1543–1553, Melbourne, Australia. Association for Computational Linguistics.

Adithya Pratapa, Monojit Choudhury, and Sunayana Sitaram. 2018b. Word embeddings for code-mixed language processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3067–3072, Brussels, Belgium. Association for Computational Linguistics.

Libo Qin, Minheng Ni, Yue Zhang, and Wanxiang Che. 2020. CoSDA-ML: Multi-lingual code-switching data augmentation for zero-shot cross-lingual nlp. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 3853–3860. International Joint Conferences on Artificial Intelligence Organization.

XiPeng Qiu, TianXiang Sun, YiGe Xu, YunFan Shao, Ning Dai, and XuanJing Huang. 2020. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, 63(10):1872–1897.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Sebastian Ruder, Matthew E. Peters, Swabha Swayamdipta, and Thomas Wolf. 2019. Transfer learning in natural language processing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials*, pages 15–18, Minneapolis, Minnesota. Association for Computational Linguistics.

Mohammad Arshi Saloot, Norisma Idris, and Rohana Mahmud. 2014. An architecture for Malay tweet normalization. *Information Processing & Management*, 50(5):621–633.

Norlela Samsudin, Mazidah Puteh, Abdul Razak Hamdan, and Mohd Zakree Ahmad Nazri. 2013. Mining opinion in online messages. *Int. J. Adv. Comput. Sci. Appl*, 4(8).

Hidayah Shafiee, Ahmad Fahmi Mahamood, Abdul Rahman Abdul Manaf, Tengku Kastriafuddin Shah Tengku Yaakob, Abdul Jalil Ramli, Zuraidi Ahmad Mokhdzar, Jamsari Jamaluddin, Maskor Bajuri, and Mohd Erpi Mohd Ali. 2019. Pengaruh bahasa rojak di media baharuterhadap bahasa kebangsaan. *International Journal of Law, Government and Communication*, 4(15):141–153.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

S. Thara and P. Poornachandran. 2018. Code-mixing: A brief survey. *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 2382–2388.

Ralf Vollmann and Soon Tek Wooi. 2019. The sociolinguistic status of Malaysian English. *Grazer Linguistische Studien*, 91:133–150.

A Appendix

A.1 Detail of Algorithm 1

Having three different languages (Malay, English and Chinese) in a sentence is one of the features of Bahasa Rojak. In general, the matrix language in Bahasa Rojak is either English or Malay, while the other of that pair and Chinese serve as inserted language independently or jointly. Considering the characteristics of Bahasa Rojak, we collect both English and Malay data in our source language, and then translate them to our target sentences in Bahasa Rojak that consist of at least one matrix and inserted language.

In Algorithm 1, we represent the source language as sl, and $sl \leftarrow \{en, ms\}$ to denote which language is the source language. The same concept applies to target language, tl represents the target language, and $tl \leftarrow \{en : [ms, zh], ms : [en, zh]\}$ means that in the source language en, we hope to translate to a target language, ms or zh. Next, we set the substitution ratio manually, in which α represents the probability of translating a source sentence to a code-mixed sentence. β means how many phrases need to be translated during the process. Finally, γ means the probability of translating a source sentence to a target sentence completely. Note that the target language is randomly selected in our algorithm.

In our augmentation algorithm, there are two important functions: $get\_phrase()$ and $Replace\_and\_Alignment()$. In the $get\_phrase()$ function, we first use the NLTK for syntax analysis, parse each input sentence, and then use POS tagging to label the phrases. In this way, we rephrase the sentence from a word-based to phrase-based tokenization structure. After building the sentence structure, we use the regular expression patterns defined in Table 4 to extract specific phrases to translate.

As we reconstruct our Bahasa Rojak sentences in the phrase-based tokenization manner, we have to implement two kinds of phase alignment methods to perfect our code-mixing sentences. Therefore, through aligning the sentence index and replacing the translated phrase in the source language, we successfully generate our Bahasa Rojak sentences.

A.2 Code-Mixing Complexity

Due to time constraints, we only sample 1% of the data from our BRCC dataset and evaluate it with the following metrics.

Switch-Point Fraction (SPF) The switch point refers the point in a sentence where two adjacent tokens are in different languages. We follow the definition proposed by Pratapa et al. (2018a), but make slight adjustments to fit our BRCC corpus. We calculate the number of switch points in a sentence, and divide it by the total number of phrase boundaries.

Code-Mixing Index (CMI) CMI is used to measure the amount of code-mixing in a corpus to account for the language distribution (Gupta et al., 2020), which was proposed by Ghosh et al. (2017); Gambäck and Das (2016). In our BRCC dataset, we use the CMI formula from Pratapa et al. (2018a), as follows:

$$C_u(x) = \frac{(N(x) - \max_{L_i \in L}\{t_{L_i}\}(x)) + P(x)}{N(x)},$$

where N denotes the number of language tokens, x is an utterance; t_{L_i} represents the tokens in language Li, P is the number of code-switching points in utterance x. Then, we compute our data at the sentence level by averaging all sentences sampled from the BRCC dataset.

Our SPF and CMI values are 0.158 and 0.384 respectively.