IndoRobusta: Towards Robustness Against Diverse Code-Mixed Indonesian Local Languages

Muhammad Farid Adilazuarda[1], **Samuel Cahyawijaya**[3], **Genta Indra Winata**[2], **Pascale Fung**[3], **Ayu Purwarianti**[1]
[1]Institut Teknologi Bandung [2]Bloomberg
[3]The Hong Kong University of Science and Technology
faridlazuarda@gmail.com

Abstract

Significant progress has been made on Indonesian NLP. Nevertheless, exploration of the code-mixing phenomenon in Indonesian is limited, despite many languages being frequently mixed with Indonesian in daily conversation. In this work, we explore code-mixing in Indonesian with four embedded languages, i.e., English, Sundanese, Javanese, and Malay; and introduce IndoRobusta[1], a framework to evaluate and improve the code-mixing robustness. Our analysis shows that the pre-training corpus bias affects the model's ability to better handle Indonesian-English code-mixing when compared to other local languages, despite having higher language diversity.

1 Introduction

Recent developments in Indonesian Natural Language Processing (NLP) have introduced an immense improvement in many aspects, including standardized benchmarks (Wilie et al., 2020; Cahyawijaya et al., 2021; Koto et al., 2020; Winata et al., 2022), large pre-trained language model (LM) (Wilie et al., 2020; Cahyawijaya et al., 2021; Koto et al., 2020), and resource expansion covering local Indonesian languages (Tri Apriani, 2016; Dewi et al., 2020; Khaikal and Suryani, 2021). Despite all these significant efforts, only a few studies focus on tackling the code-mixing phenomenon that naturally occurs in the Indonesian language. Code-mixing [2] is an interesting phenomenon where people change between languages and mix them in a conversation or sentence. In Indonesia, many people speak at least two languages (i.e., Indonesian and a local language) in their day-to-day conversation (Aji et al., 2022), and use diverse written and spoken styles specific to their home regions.

Inspired by the frequently occurring code-mixing phenomenon in Indonesian, we want to answer two research questions "*Is the LMs performance susceptible to linguistically diverse Indonesian code-mixed text?*" and "*How can we improve the model's robustness against a variety of mixed-language texts?*". Therefore, we introduce IndoRobusta, a framework to assess and improve code-mixed robustness. Using our IndoRobusta-Blend, we conduct experiments to evaluate existing pre-trained LMs using code-mixed language scenario to simulate the code-mixing phenomenon. We focus on Indonesian as the matrix language (L1) and the local language as the embedded language (L2) (Myers-Scotton and Jake, 2009).We measure the robustness of Indonesian code-mixed sentences for English (en) and three local languages, i.e, Sundanese (su), Javanese (jv), and Malay (ms)[3] on sentiment and emotion classification tasks. In addition, we explore methods to improve the robustness of LMs to code-mixed text. Using our IndoRobusta-Shot, we perform adversarial training to improve the code-mixed robustness of LMs. We explore three kinds of tuning strategies: 1) code-mix only, 2) two-steps, and 3) joint training, and empirically search for the best strategy to improve the model robustness on code-mixed data.

We summarize our contribution as follows:

- We develop a benchmark to assess the robustness of monolingual and multilingual LMs on four L2 code-mixed languages covering English (en), Sundanese (su), Javanese (jv), and Malay (ms);
- We introduce various adversarial tuning strategies to better improve the code-mixing robustness of LMs. Our best strategy improves the

[1]We will release the code upon acceptance. We provide the anonymized code repository at https://anonymous.4open.science/r/indorobusta-1403/

[2]In our case, code-mixing refers to intra-sentential code-switching where the language alternation occurs in the sentence.

[3]Malay is not a direct Indonesian local language, but it is considered as the parent language to many of Indonesian local languages such as Jambi, Malay, Minangkabau, and Betawi.

Proceedings of the First Workshop on Scaling Up Multilingual Evaluation, pages 25–34
Nov 20–23, 2022.

accuracy by ∼5% on the code-mixed test set and ∼2% on the monolingual test set;

- We show that existing LMs are more robust to English code-mixing rather than to local languages code-mixing and provide detailed analysis of this phenomenon.

2 IndoRobusta Framework

IndoRobusta is a code-mixing robustness framework consisting of two main modules: 1) `IndoRobusta-Blend`, which evaluates the code-mixing robustness of LMs through a code-mixing perturbation method, and 2) `IndoRobusta-Shot`, which improves the code-mixing robustness of LMs using a code-mixing adversarial training technique.

2.1 Notation

Given a monolingual language sentence $X = \{w_1, w_2, \ldots, w_M\}$, where w_i denotes a token in a sentence and M denotes the number of tokens in a sentence, we denote a monolingual language dataset $\mathcal{D} = \{(X_1, Y_1), (X_2, Y_2), \ldots, (X_N, Y_N)\}$, where (X_i, Y_i) denotes a sentence-label pair and N is the number of samples. Given a token w_i, a mask token w^{mask} and a sentence X, we define a sentence with masked w_i token as $X_{\setminus w_i} = \{w_1, w_2, \ldots, w_{i-1}, w^{mask}, w_{i+1}, \ldots, w_M\}$. We further define a code-mixing dataset $\mathcal{D}' = \{(X'_1, Y_1), (X'_2, Y_2), \ldots, (X'_N, Y_N)\}$ where X'_i denotes the code-mixed sentence. Lastly, we define the set of parameters of a language model as θ, the prediction label of a sentence X as $f_\theta(X)$, the prediction score of the label Y given a sentence X as $f_\theta(Y|X)$, and the prediction score of the label other than Y given a sentence X as $f_\theta(\bar{Y}|X)$.

2.2 IndoRobusta-Blend

`IndoRobusta-Blend` is a code-mixing robustness evaluation method that involves two steps: 1) code-mixed dataset generation and 2) model evaluation on the code-mixed dataset. The first step is synthetically generating the code-mixed example using the translation of important words in a sentence. To do so, we formally define the importance I_{w_i} of the word w_i for a given sample (X, Y) as:

$$I_{w_i} = \begin{cases} f_\theta(Y|X) - f_\theta(Y|X_{\setminus w_i}), \\ \quad \text{if} f_\theta(X) = f_\theta(X_{\setminus w_i}) = Y \\ [f_\theta(Y|X) - f_\theta(Y|X_{\setminus w_i})] + \\ \quad [f_\theta(\bar{Y}|X) - f_\theta(\bar{Y}|X_{\setminus w_i})], \text{otherwise.} \end{cases}$$

Algorithm 1 Code-mixed sample generation workflow in IndoRobusta framework

Require: Clean sentence example X, ground truth label Y, language model Θ, similarity threshold α, perturb ratio R, embedded Language L
Ensure: Adversarial Example X_{adv}

```
Y' ← PREDICT(Θ, X)
if Y' ≠ Y then
    return X
end if
W ←R% highest I_{w_i} words in X
W^L ← TRANSLATE(W, target-language=L)
X_adv ← PERTURB(X, W^L)
if SIM(X, X_adv) < α then
    while SIM(X, X_adv) < α do
        W^L ← RESAMPLE(W^L, I_{w_i})
        X_adv ← PERTURB(X, W^L)
    end while
end if
return X_adv
```

`IndoRobusta-Blend` takes $R\%$ words with the highest I_{w_i}, denoted as the **perturbation ratio**, and applies a word-level translation for each word. Using the translated words, `IndoRobusta-Blend` generates a code-mixed sentence by replacing the important words with their corresponding translation. To ensure generating a semantically-related code-mixed samples, we define a similarity threshold α to constraint the cosine distance between X and X_{adv}. When the distance between X and X_{adv} is below α, we resample the perturbed words and generate a more similar X_{adv}.

More formally, we define the code-mixing sample generation as a function $g(X, Y, \theta) = X_{adv}$. To generate the code-mixed dataset $\mathcal{D}'$ from the monolingual dataset $\mathcal{D}$ and a model θ, `IndoRobusta-Blend` applies $g(X_i, Y_i, \theta)$ to each sample (X_i, Y_i) in $\mathcal{D}$. Using $\mathcal{D}$ and $\mathcal{D}'$, `IndoRobusta-Blend` evaluates the robustness of the fine-tuned model θ', trained on $\mathcal{D}$, by evaluating θ on both $\mathcal{D}$ and $\mathcal{D}'$. More formally, we define the code-mixed sample generation in Algorithm 1.

2.3 IndoRobusta-Shot

`IndoRobusta-Shot` is a code-mixing adversarial defense method, which aims to improve the robustness of the model. `IndoRobusta-Shot` does so by fine-tuning the model on the generated code-mixed dataset $\mathcal{D}'$. Similar to `IndoRobusta-Blend`, our `IndoRobusta-`

Model	Orig.	en	jw	ms	su	avg
			EmoT			
IB_B	72.42	9.55	12.35	**9.47**	9.39	**10.19**
IB_L	75.53	**9.24**	**12.12**	10.23	**9.32**	10.23
mB_B	61.14	12.50	14.02	12.73	12.50	12.96
XR_B	72.88	10.98	13.94	13.18	12.50	12.65
XR_L	**78.26**	12.27	13.03	12.42	11.74	12.37
Avg		10.91	13.09	11.61	11.09	
			SmSA			
IB_B	91.00	**1.33**	5.07	3.20	2.40	3.00
IB_L	**94.20**	2.47	4.13	4.00	**2.20**	3.20
mB_B	83.00	2.20	**3.00**	2.93	2.47	**2.65**
XR_B	91.53	3.40	3.80	4.27	4.27	3.94
XR_L	94.07	2.13	3.20	**2.60**	2.73	2.67
Avg		2.31	3.84	3.40	2.81	

Table 1: Delta accuracy with $R = 0.4$ on the test data. A lower value denotes better performance. We **bold** the best score and underline the second-best score.

Model	CM Only		Two-Step		Joint	
	Orig	CM	Orig	CM	Orig	CM
			EmoT			
IB_B	45.13	66.53	69.85	68.31	74.68	67.27
IB_L	63.29	68.58	73.06	69.46	75.90	68.01
mB_B	32.97	58.11	54.72	59.68	62.98	56.54
XR_B	57.59	68.40	72.17	69.11	74.38	67.26
XR_L	**71.61**	**71.56**	**77.13**	**70.44**	**78.31**	**70.06**
			SmSA			
IB_B	45.10	93.51	89.81	92.68	92.52	90.71
IB_L	**68.40**	94.67	**90.60**	94.12	94.73	93.00
mB_B	51.72	83.73	78.95	85.16	85.61	84.31
XR_B	59.31	91.37	68.08	93.87	93.77	92.21
XR_L	63.06	**95.07**	85.96	**95.35**	**95.35**	**93.99**

Table 2: Accuracy on original (Orig.) and code-mixing (CM) test sets after adversarial training with different tuning strategies.

`Shot` generates $\mathcal{D}'$ from $\mathcal{D}$ and θ by utilizing the code-mixed sample generation method $g(\theta, X, Y)$. Three different fine-tuning scenarios are explored in `IndoRobusta-Shot` , i.e., **code-mixed-only tuning**, which fine-tune the model only on $\mathcal{D}'$; **two-step tuning**, which first fine-tune the model on $\mathcal{D}$, followed by a second-phase fine-tuning on $\mathcal{D}'$; and **joint training**, which fine-tunes the model on a combined dataset from $\mathcal{D}$ and $\mathcal{D}'$.

3 Experiment Setting

3.1 Dataset

We employ two Indonesian multi-class classification datasets for conducting our experiments, i.e., a sentiment-analysis dataset, SmSA (Purwarianti and Crisdayanti, 2019), and an emotion classification dataset, EmoT (Saputri et al., 2018). SmSA is a sentence-level sentiment analysis dataset consists of 12,760 samples and is labelled intro three possible sentiments values, i.e., positive, negative, and neutral. EmoT is an emotion classification dataset which consists of 4,403 samples and covers five different emotion labels, i.e., anger, fear, happiness, love, and sadness. The statistics of SmSA and EmoT datasets are shown in Appendix Table 4.

3.2 Code-mixed Sample Generation

For our experiment, we use Indonesian as the L1 language and explore four commonly used L2 languages, i.e., English, Sundanese, Javanese, and Malay. We experiment with different code-mixed perturbation ratio $R = \{0.2, 0.4, 0.6, 0.8\}$ to assess the susceptibility of models. We utilize Google Translate to translate important words to generate the code-mixed sentence X'.

3.3 Baseline Models

We include both monolingual and multilingual pre-trained LMs with various model size in our experiment. For Indonesian monolingual pre-trained LMs, we utilize two models: $\text{IndoBERT}_{\text{BASE}}$ (IB_B) and $\text{IndoBERT}_{\text{LARGE}}$ (IB_L) (Wilie et al., 2020), while for the multilingual LMs, we employ $\text{mBERT}_{\text{BASE}}$ (mB_B) (Devlin et al., 2019), $\text{XLM-R}_{\text{BASE}}$ (XR_B), and $\text{XLM-R}_{\text{LARGE}}$ (IB_L) (Conneau et al., 2020). Note that all of the multilingual models are knowledgeable of the Indonesian language and all L2 languages used since all the languages are covered in their pre-training corpus.

3.4 Training Setup

To evaluate the model robustness, We fine-tune the model on D using the Adam optimizer (Kingma and Ba, 2014) with a learning rate of 3e-6, and a batch size of 32. We train the model for a fixed number of epoch, i.e., 5 epochs for sentiment analysis and 10 epochs for emotion classification. We run each experiment three times using different random seeds and report the averaged score over three runs. For the adversarial training, we train the model using Adam optimizer with a learning rate of 3e-6 and a batch size of 32. We set the maximum epoch to 15, and apply early stopping with the early stopping patience set to 5.

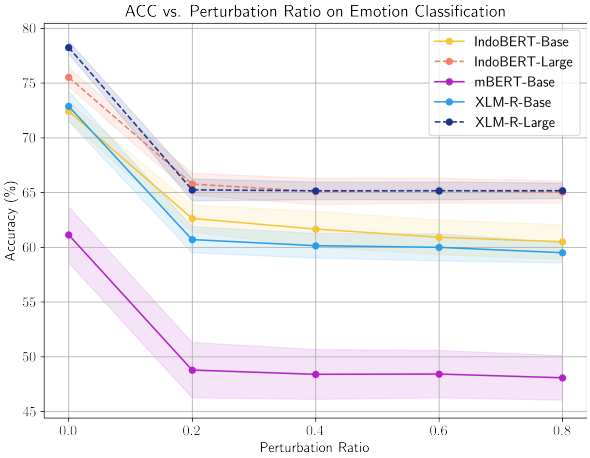

Figure 1: The effect of perturbation ratio to the evaluation accuracy in the emotion classification task.

3.5 Evaluation Setup

To measure the robustness of the models, `IndoRobusta` uses three evaluation metrics: 1) the accuracy on the monolingual dataset, 2) the accuracy on the code-mixed dataset, and 3) delta accuracy (Srinivasan et al., 2018). We measure accuracy before and after adversarial training to analyze the effectiveness of the adversarial training method in the `IndoRobusta-Shot`.

4 Result and Discussion

4.1 Code-Mixing Robustness

The result of the robustness evaluation with $R = 0.4$ is shown in Table 1. Existing LMs are more prone to code-mixing in the emotion classification task, with $> 10\%$ performance reduction, compared to 3% on the sentiment analysis task. Interestingly, monolingual models, i.e., IndoBERT$_{\text{BASE}}$ and IndoBERT$_{\text{LARGE}}$, are more robust in the emotion classification task compared to the multilingual models with 2% higher delta accuracy. While on the sentiment analysis task, all models perform almost equally good in all L2 languages.

We also observe that the robustness on English language are generally lower than Javanese and Malay in all models. We conjecture that this is due to the bias from the pre-training corpus, since pre-training corpus is gathered from online platforms, and Indonesian-English code-mixing is particularly common in such platforms (Nuraeni et al., 2018; Aulia and Laksman-Huntley, 2017; Marzona, 2017). While Indonesian and local language code-mixing are considered a secondary choice in online platforms (Cahyani et al., 2020) and is more commonly used in the day-to-day conversation (Ginting, 2019; Muslimin, 2020).

4.2 Impact of Perturbation Ratio

According to Figure 1, we can clearly observe that LMs performance gets lower as the perturbation ratio R increases. Interestingly, the steepest decline happens when the perturbation ratio $R = 0.4$, and the model performance decreases slightly with a higher perturbation ratio ($R = \{0.4, 0.6, 0.8\}$). This result suggests that translating the words with high importance as mentioned in §2.2, effectively alters the model prediction.

We further analyzed the generated code-mixed sentence, we show the example of the generated code-mixed sentences from `IndoRobusta` in Table 3. To generate the code-mixed sentence, we select important words from the sentence and perform word-level translation into four different L2 languages, i.e English, Sundanese, Javanese, and Malay. We analyze the important word selected by the I_{w_i} over a dataset, we count the total number of times a word is selected as important with $R = \{0.2, 0.4, 0.6, 0.8\}$, denoted as informative frequency (IF). For each word, we divide the IF with its document frequency (DF) to produce a normalized informative frequency (IF/DF). We show the top-20 words with highest IF/DF score for emotion classification task in Table 5 and for sentiment analysis task in Table 6. Most of the words are related to the label in the lexical-sense, e.g.: 'regret', 'disappointing', and 'disappointed' are commonly associated with **negative** sentiment, while 'comfortable', 'fun', 'nice' are commonly associated with **positive** sentiment. Most of the time, the word-translations for all L2 languages are valid and infer similar meaning. We find that the model prediction is still largely shifted even though the important word is translated correctly. This shows that, despite having learned all the languages individually, LMs are unable to generalize well on code-mixed sentences and improving robustness with an explicit tuning is required to achieve comparable performance.

4.3 Improving Code-Mixing Robustness

Table 2 shows the results of the adversarial training using different tuning strategies. **Code-mixing only** and **two-step**-tuning yield a better improvement on the code-mixed data compared to the **joint training**. Nevertheless, **code-mixing only**-tuning significantly hurts the performance on the original

Code-Mixed Text	Translation
sate kambing dan gulai kambing nya **sedap** penyajian makannan nya juga sangat cepat tempat nya cukup bersih	lamb satay and lamb curry are **yummy**, quick serving, and the place is quite clean
hayam goreng, tempe, tahu goreng dengan sambal yang pedas mantap sejak zaman dulu **teu** dan terjangkau	fried **chicken**, tempe, fried tofu with spicy chilli sauce has been **delicious** since ancient times.
tidak bisa **mudhun** galau mikirin lo	I cannot **sleep** because I am thinking about you
meski masa kampanye sudah selesai bukan berati habis pula **effort** mengerek tingkat kedipilihan elektabilitas.	Even though the campaign period is over, it doesn't mean that the **effort** to raise the electability level is over.

Table 3: Example of generated code-mixed sentences with `IndoRobusta`. **Blue** denotes an Malay word, **Orange** denotes a Sundanese word, **Red** denotes a Javanese word and **Violet** denotes an English word. The **bold words** in the translation column are the corresponding colored word translations in English.

data, while the **two-step**-tuning can retain much better performance on the original data. **joint training**, on the other hand, yields the highest performance on the original data, and even outperforms the model trained only on the original data by $\sim 2\%$ accuracy while maintaining considerably high performance on the code-mixing data.

5 Related Work

Code-Mixing in NLP Code-mixing has been studied in various language pairs such as Chinese-English (Lyu et al., 2010; Winata et al., 2019b; Lin et al., 2021; Lovenia et al., 2022), Cantonese-English (Dai et al., 2022), Hindi-English (Banerjee et al., 2018; Khanuja et al., 2020), Spanish-English (Aguilar et al., 2018; Winata et al., 2019a; Aguilar et al., 2020), Indonesian-English (Barik et al., 2019; Stymne et al., 2020), Arabic-English (Hamed et al., 2019), etc. Multiple methods have been proposed to better understand code-mixing including multi-task learning (Song et al., 2017; Winata et al., 2018), data augmentation (Winata et al., 2019b; Chang et al., 2019; Lee et al., 2019; Qin et al., 2020; Jayanthi et al., 2021; Rizvi et al., 2021), meta-learning (Winata et al., 2020), and multilingual adaptation (Winata et al., 2021). In this work, we explore code-mixing in Indonesian with four commonly used L2 languages.

Model Robustness in NLP Prior works in robustness evaluation focus on data perturbation methods (Tan and Joty, 2021; Ishii et al., 2022). Various textual perturbation methods have been introduced (Jin et al., 2019; Dhole et al., 2021), which is an essential part of robustness evaluation. Moreover, numerous efforts in improving robustness have also been explored, including adversarial training on augmented data (Li et al., 2021; Li and Specia, 2019), harmful instance removal (Bang et al., 2021; Kobayashi et al., 2020) and robust loss function (Bang et al., 2021; Zhang and Sabuncu, 2018). In this work, we focus on adversarial training, since the method is effective for handling low-resource data, such as code-mixing.

6 Conclusion

We introduce `IndoRobusta`, a framework to effectively evaluate and improve model robustness. Our results suggest adversarial training can significantly improve the code-mixing robustness of LMs, while at the same time, improving the performance on the monolingual data. Moreover, we show that existing LMs are more robust to English code-mixed and conjecture that this comes from the source bias in the existing pre-training corpora.

Limitations

One of the limitation of our approach is that we utilize Google Translate to generate the perturbed code-mixing samples instead of manually generating natural code-mixing sentences. Common mistake made from the generated code-mixed sentence is on translating ambiguous terms, which produces inaccurate word-level translation and alters the meaning of the sentence. For future work, we expect to build a higher quality code-mixed sentences to better assess the code-mixed robustness of the existing Indonesian large-pretrained language models.

Acknowledgements

We sincerely thank the anonymous reviewers for their insightful comments on our paper.

References

Gustavo Aguilar, Fahad AlGhamdi, Victor Soto, Mona Diab, Julia Hirschberg, and Thamar Solorio. 2018. Overview of the CALCS 2018 Shared Task: named

entity recognition on code-switched data. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, Melbourne, Australia. Association for Computational Linguistics.

Gustavo Aguilar, Sudipta Kar, and Thamar Solorio. 2020. Lince: A centralized benchmark for linguistic code-switching evaluation. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 1803–1813.

Alham Aji, Genta Indra Winata, Fajri Koto, Samuel Cahyawijaya, Ade Romadhony, Rahmad Mahendra, Kemal Kurniawan, David Moeljadi, Radityo Eko Prasojo, Timothy Baldwin, et al. 2022. One country, 700+ languages: Nlp challenges for underrepresented languages and dialects in indonesia. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7226–7249.

M. Aulia and M. Laksman-Huntley. 2017. Indonesian-english code-switching on social media. In *Cultural Dynamics in a Globalized World*, pages 791–796. Routledge.

Suman Banerjee, Nikita Moghe, Siddhartha Arora, and Mitesh M. Khapra. 2018. A dataset for building code-mixed goal oriented conversation systems. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3766–3780, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Yejin Bang, Etsuko Ishii, Samuel Cahyawijaya, Ziwei Ji, and Pascale Fung. 2021. Model generalization on covid-19 fake news detection. In *CONSTRAINT@AAAI*.

Anab Maulana Barik, Rahmad Mahendra, and Mirna Adriani. 2019. Normalization of indonesian-english code-mixed twitter data. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 417–424.

Hilda Cahyani, Umi Tursini, and Nurenzia Yannuar. 2020. Mixing and switching in social media: Denoting the indonesian "keminggris" language. 10:2020.

Samuel Cahyawijaya, Genta Indra Winata, Bryan Wilie, Karissa Vincentio, Xiaohong Li, Adhiguna Kuncoro, Sebastian Ruder, Zhi Yuan Lim, Syafri Bahar, Masayu Leylia Khodra, Ayu Purwarianti, and Pascale Fung. 2021. Indonlg: Benchmark and resources for evaluating indonesian natural language generation.

Ching-Ting Chang, Shun-Po Chuang, and Hung-Yi Lee. 2019. Code-switching sentence generation by generative adversarial networks and its application to data augmentation. *Proc. Interspeech 2019*, pages 554–558.

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.

Wenliang Dai, Samuel Cahyawijaya, Tiezheng Yu, Elham J. Barezi, Peng Xu, Cheuk Tung YIU, Rita Frieske, Holy Lovenia, Genta Winata, Qifeng Chen, Xiaojuan Ma, Bertram Shi, and Pascale Fung. 2022. Ci-avsr: A cantonese audio-visual speech dataset-for in-car command recognition. In *Proceedings of the Language Resources and Evaluation Conference*, pages 6786–6793, Marseille, France. European Language Resources Association.

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.

Nindian Puspa Dewi, Joan Santoso, Ubaidi Ubaidi, and Eka Rahayu Setyaningsih. 2020. Combination of genetic algorithm and brill tagger algorithm for part of speech tagging bahasa madura. *Proceeding of the Electrical Engineering Computer Science and Informatics*, 7(0).

Kaustubh D Dhole, Varun Gangal, Sebastian Gehrmann, Aadesh Gupta, Zhenhao Li, Saad Mahamood, Abinaya Mahendiran, Simon Mille, Ashish Srivastava, Samson Tan, et al. 2021. Nl-augmenter: A framework for task-sensitive natural language augmentation. *arXiv preprint arXiv:2112.02721*.

Carolin Rninta Ginting. 2019. Analysis of code-switching and code-mixing in the learning process of indonesia subject at grade 3 of SD negeri 2 jayagiri. In *Proceedings of the Eleventh Conference on Applied Linguistics (CONAPLIN 2018)*. Atlantis Press.

Injy Hamed, Moritz Zhu, Mohamed Elmahdy, Slim Abdennadher, and Ngoc Thang Vu. 2019. Code-switching language modeling with bilingual word embeddings: A case study for egyptian arabic-english. In *International Conference on Speech and Computer*, pages 160–170. Springer.

Etsuko Ishii, Yan Xu, Samuel Cahyawijaya, and Bryan Wilie. 2022. Can question rewriting help conversational question answering? In *Proceedings of the Third Workshop on Insights from Negative Results in NLP*, pages 94–99, Dublin, Ireland. Association for Computational Linguistics.

Sai Muralidhar Jayanthi, Kavya Nerella, Khyathi Raghavi Chandu, and Alan W Black. 2021. Codemixednlp: An extensible and open nlp toolkit for code-mixing. In *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, pages 113–118.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2019. Is bert really robust? a strong baseline for natural language attack on text classification and entailment.

Muhammad Fiqri Khaikal and Arie Ardiyanti Suryani. 2021. Statistical machine translation dayak language – indonesia language. *Informatika Mulawarman : Jurnal Ilmiah Ilmu Komputer*, 16(1):49.

Simran Khanuja, Sandipan Dandapat, Anirudh Srinivasan, Sunayana Sitaram, and Monojit Choudhury. 2020. Gluecos: An evaluation benchmark for code-switched nlp. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3575–3585.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Sosuke Kobayashi, Sho Yokoi, Jun Suzuki, and Kentaro Inui. 2020. Efficient estimation of influence of a training instance. In *Proceedings of SustaiNLP: Workshop on Simple and Efficient Natural Language Processing*, pages 41–47, Online. Association for Computational Linguistics.

Fajri Koto, Afshin Rahimi, Jey Han Lau, and Timothy Baldwin. 2020. Indolem and indobert: A benchmark dataset and pre-trained language model for indonesian nlp.

Grandee Lee, Xianghu Yue, and Haizhou Li. 2019. Linguistically motivated parallel data augmentation for code-switch language modeling. In *Interspeech*, pages 3730–3734.

Zhenhao Li and Lucia Specia. 2019. Improving neural machine translation robustness via data augmentation: Beyond back-translation. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 328–336, Hong Kong, China. Association for Computational Linguistics.

Zongyi Li, Jianhan Xu, Jiehang Zeng, Linyang Li, Xiaoqing Zheng, Qi Zhang, Kai-Wei Chang, and Cho-Jui Hsieh. 2021. Searching for an effective defender: Benchmarking defense against adversarial word substitution.

Zhaojiang Lin, Andrea Madotto, Genta Indra Winata, Peng Xu, Feijun Jiang, Yuxiang Hu, Chen Shi, and Pascale Fung. 2021. Bitod: A bilingual multi-domain dataset for task-oriented dialogue modeling. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

Holy Lovenia, Samuel Cahyawijaya, Genta Winata, Peng Xu, Yan Xu, Zihan Liu, Rita Frieske, Tiezheng Yu, Wenliang Dai, Elham J. Barezi, Qifeng Chen, Xiaojuan Ma, Bertram Shi, and Pascale Fung. 2022. Ascend: A spontaneous chinese-english dataset for code-switching in multi-turn conversation. In *Proceedings of the Language Resources and Evaluation Conference*, pages 7259–7268, Marseille, France. European Language Resources Association.

Dau-Cheng Lyu, Tien Ping Tan, Chng Eng Siong, and Haizhou Li. 2010. Seame: a mandarin-english code-switching speech corpus in south-east asia. In *INTERSPEECH*.

Yessy Marzona. 2017. The use of code mixing between indonesian and english in indonesian advertisement of gadis.

Afif Ikhwanul Muslimin. 2020. Code-mixing of javanese language and bahasa indonesia in the friday prayer sermon at miftahul hidayah mosque, pendem village, city of batu, east java. *MABASAN*, 14(2):277–296.

Carol Myers-Scotton and Janice Jake. 2009. A universal model of code-switching and bilingual language processing and production. *The Cambridge Handbook of Linguistic Code-switching*, pages 336–357.

Bani Nuraeni, Mochammad Farid, and Sri Cahyati. 2018. The use of indonesian english code mixing on instagram captions. *PROJECT (Professional Journal of English Education)*, 1:448.

Ayu Purwarianti and Ida Ayu Putu Ari Crisdayanti. 2019. Improving bi-lstm performance for indonesian sentiment analysis using paragraph vector. In *2019 International Conference of Advanced Informatics: Concepts, Theory and Applications (ICAICTA)*, pages 1–5. IEEE.

Libo Qin, Minheng Ni, Yue Zhang, and Wanxiang Che. 2020. Cosda-ml: Multi-lingual code-switching data augmentation for zero-shot cross-lingual nlp. *ArXiv*, abs/2006.06402.

Mohd Sanad Zaki Rizvi, Anirudh Srinivasan, Tanuja Ganu, Monojit Choudhury, and Sunayana Sitaram. 2021. Gcm: A toolkit for generating synthetic code-mixed text. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 205–211.

Mei Silviana Saputri, Rahmad Mahendra, and Mirna Adriani. 2018. Emotion classification on indonesian twitter dataset. In *2018 International Conference on Asian Language Processing (IALP)*, pages 90–95. IEEE.

Xiao Song, Yuexian Zou, Shilei Huang, Shaobin Chen, and Yi Liu. 2017. Investigating multi-task learning for automatic speech recognition with code-switching between mandarin and english. In *2017 International Conference on Asian Language Processing (IALP)*, pages 27–30.

Vignesh Srinivasan, Arturo Marban, Klaus-Robert Müller, Wojciech Samek, and Shinichi Nakajima. 2018. Robustifying models against adversarial attacks by langevin dynamics.

Sara Stymne et al. 2020. Evaluating word embeddings for indonesian–english code-mixed text based on synthetic data. In *Proceedings of the The 4th Workshop on Computational Approaches to Code Switching*, pages 26–35.

Samson Tan and Shafiq Joty. 2021. Code-mixing on sesame street: Dawn of the adversarial polyglots. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3596–3616.

Novi Safriadi Tri Apriani, Herry Sujaini. 2016. Pengaruh kuantitas korpus terhadap akurasi mesin penerjemah statistik bahasa bugis wajo ke bahasa indonesia. *Jurnal Sistem dan Teknologi Informasi*, 4(1):168–173.

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

Genta Indra Winata, Alham Fikri Aji, Samuel Cahyawijaya, Rahmad Mahendra, Fajri Koto, Ade Romadhony, Kemal Kurniawan, David Moeljadi, Radityo Eko Prasojo, Pascale Fung, Timothy Baldwin, Jey Han Lau, Rico Sennrich, and Sebastian Ruder. 2022. Nusax: Multilingual parallel sentiment dataset for 10 indonesian local languages.

Genta Indra Winata, Samuel Cahyawijaya, Zhaojiang Lin, Zihan Liu, Peng Xu, and Pascale Fung. 2020. Meta-transfer learning for code-switched speech recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3770–3776, Online. Association for Computational Linguistics.

Genta Indra Winata, Samuel Cahyawijaya, Zihan Liu, Zhaojiang Lin, Andrea Madotto, and Pascale Fung. 2021. Are multilingual models effective in code-switching? In *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, pages 142–153, Online. Association for Computational Linguistics.

Genta Indra Winata, Zhaojiang Lin, and Pascale Ngan Fung. 2019a. Learning multilingual meta-embeddings for code-switching named entity recognition. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*.

Genta Indra Winata, Andrea Madotto, Chien-Sheng Wu, and Pascale Fung. 2018. Code-switching language modeling using syntax-aware multi-task learning. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, pages 62–67, Melbourne, Australia. Association for Computational Linguistics.

Genta Indra Winata, Andrea Madotto, Chien-Sheng Wu, and Pascale Fung. 2019b. Code-switched language models using neural based synthetic data from parallel sentences. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 271–280, Hong Kong, China. Association for Computational Linguistics.

Zhilu Zhang and Mert R. Sabuncu. 2018. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 87928802, Red Hook, NY, USA. Curran Associates Inc.

A Annotation Guideline for Human Evaluation

We introduce a manual annotation to evaluate the generated code-mixed sentences. To validate the quality of our perturbed code-mixing sentences, we hire 3 native annotators for each language to evaluate the generated Sundanese-Indonesian and Javanese-Indonesian code-mixed sentences, and 3 Indonesian annotators with professional English proficiency for assessing the generated English-Indonesian code-mixed sentences. Each human annotator is asked to assess the quality of 40 randomly sampled code-mixed sentences and provide a score in range of $[1, 2, 3, 4, 5]$ with 1 denotes an incomprehensible code-mixing sentence and 5 denotes a perfectly natural code-mixed sentence. The detailed annotation guideline is described in A The score between annotators are averaged to reduce annotation bias.

Dataset	|Train|	|Valid|	|Test|	#Class
EmoT	3,521	440	442	5
SmSA	11,000	1,260	500	3

Table 4: Statistics of EmoT and SmSA datasets.

Table 4 contains more details of the EmoT and SmSA dataset that we used in the sample generation. Sample generated by perturbing these datasets will later be annotated.

First, we compile 40 samples generated from each model into an excel sheet. Then the annotator is given access to the file. Before starting the annotation process, the annotator is given instructions and a definition of the score that can be assigned to the sample sentence. For each row in the given excel file, the annotator is asked to read the code-mixing sentence generated by the model and provide annotation values. Annotation scores are defined as follows:

1 - unnatural (unintelligible sentence)
2 - less natural (sentences can be understood even though they are strange)
3 - adequately natural (sentences can be understood even though they are not used correctly)
4 - imperfect natural (sentences are easy to understand, but some of the words used are slightly inaccurate)
5 - natural (sentences are easy to understand and appropriate to use)

B Annotation Result

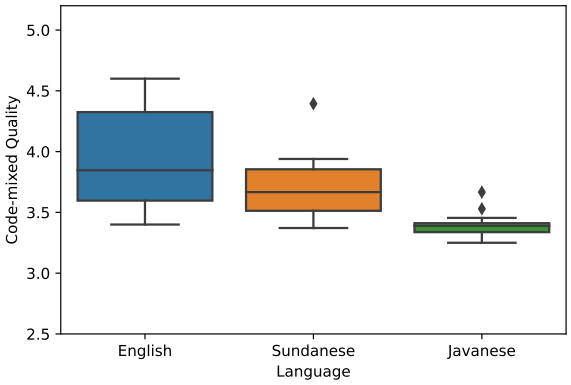

Figure 2: Human evaluation result from the generated code-mixed samples averaged over three annotators.

Figure 2 shows the result of the human assessment on the generated code-mixed sentences. The results indicates that the generated sentences are adequately natural by achieving an average score of 3.94 for English-Indonesian, 3.71 for Sundanese-Indonesian, and 3.39 for Javanese-Indonesian.

Word	IF	DF	IF/DF	jw	ms	su	en
love	1078	1260	0.856	tresna	cinta	cinta	love
tolong	1408	2520	0.559	bantuan	membantu	Tulung	help
km	1183	2520	0.469	km	km	km	km
kasih	2947	6300	0.468	tresna	cinta	cinta	love
pakai	1505	3360	0.448	nggunakake	guna	ngagunakeun	use
udh	1659	3780	0.439	wis	Sudah	Geus	Already
setan	1088	2520	0.432	setan	syaitan	Sétan	Devil
hrs	1078	2520	0.428	**jam**	**jam**	**tabuh**	**hrs**
cinta	5559	13020	0.427	tresna	cinta	**cinta**	love
jam	2495	5880	0.424	jam	pukul	tabuh	o'clock
gua	1594	3780	0.422	aku	saya	abdi	I
jatuh	1768	4200	0.421	tiba	jatuh	ragrag ka handap	fall down
mobil	1057	2520	0.419	mobil	kereta	mobil	car
sehat	1214	2940	0.413	**sehat**	sihat	cageur	healthy
beneran	1351	3360	0.402	tenan	sungguh	saleresna	really
kadang	1175	2940	0.400	kadhangkala	kadang-kadang	sakapeung	sometimes
lu	1505	3780	0.398	**lu**	**lu**	**lu**	**lu**
ketemu	1641	4200	0.391	ketemu	berjumpa	papanggih	meet
dgn	2254	5880	0.383	karo	dengan	kalawan	with
kantor	1127	2940	0.383	kantor	pejabat	kantor	office

Table 5: Top 20 most perturbed word on **emotion classification** experiments conducted on test data and their translation on four languages. **Red** denotes mistranslated words due to ambiguity or translator limitation.

Word	IF	DF	IF/DF	jw	ms	su	en
cocok	1750	2100	0.833	cocok	sesuai	cocog	suitable
asik	2338	2940	0.795	Asik	Asik	Asik	Asik
nyaman	2905	3780	0.769	nyaman	selesa	sreg	comfortable
menyesal	2240	2940	0.76	getun	penyesalan	kaduhung	regret
mantap	8456	11340	0.746	ajeg	mantap	ajeg	steady
mengecewakan	3094	4200	0.737	nguciwani	mengecewakan	nguciwakeun	disappointing
kecewa	21910	30660	0.715	kuciwa	kecewa	kuciwa	disappointed
enak	9443	14700	0.642	**becik**	bagus	hade	nice
jelek	1617	2520	0.642	ala	teruk	goréng	bad
salut	1834	2940	0.624	salam	tabik hormat	salam	salute
memuaskan	2877	4620	0.623	marem	memuaskan	nyugemakeun	satisfying
keren	3136	5040	0.622	**kelangan**	**sejuk**	tiis	cool
kadaluarsa	1827	2940	0.621	kadaluarsa	tamat tempoh	kadaluwarsa	expired
murah	3094	5040	0.614	murah	murah	murah	inexpensive
kartu	2058	3360	0.613	kertu	kad	kartu	card
banget	2434	41160	0.591	banget	sangat	pisan	very
bangga	148	2520	0.589	bangga	bangga	reueus	proud
mending	1974	3360	0.588	luwih apik	lebih baik	Leuwih alus	Better
uang	4396	7560	0.581	dhuwit	wang	duit	money
id	1442	2520	0.572	id	ID	**en**	id

Table 6: Top 20 most perturbed word on **sentiment analysis** experiments conducted on test data and their translation on four languages. **Red** denotes mistranslated words due to ambiguity or translator limitation.