2023 International Conference on Computational Science and Computational Intelligence (CSCI) | 979-8-3503-6151-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/CSCI62032.2023.00036

Benchmarking Current State-of-the-Art Transformer Models on Token Level Language Identification and Language Pair Identification

*Howard Prioleau
Computer Science
Howard University
Washington DC, USA
howard.prioleau@bison.howard.edu

Saurav K. Aryal
Computer Science
Howard University
Washington DC, USA
saurav.k.aryal@gmail.com

***Abstract*—With the rise of internet usage, code-switching, where multiple languages or dialects intermingle, has surged. Traditional linguistic analysis struggles with this mixed text, as they're typically monolingual-focused. This paper delves into two core tasks for analyzing code-switched data: Token Level Language Identification (LID) and our newly proposed Language Pair Identification (LPI). We benchmarked and compared current state-of-art transformer models across both tasks to gauge their applicability to the tasks. Our results showed that state-of-the-art multilingual transformer models could achieve state-of-the-art performance on both tasks. The impressive performance on LPI suggests that this will be the first step to utilizing Language Pair Identification to assist in various facets related to Code-Switched corpora and classification performance.**

***Index Terms*—Language identification, Token Level Analysis, Language Pair Recognition, BERT, Transformer**

I. Introduction

The meteoric rise in the use of the internet through social media and communications is due to the ever-increasing ease of access worldwide. The increased internet access has generated massive amounts of human-generated data, specifically textual-based data, which has led to an emergence of a field of Natural Language Processing to analyze and handle massive amounts of data. However, one major drawback of this is that the research of analyzing and handling the data has been focused on monolingual analysis. At the same time, this is important to conduct monolingual research, with the globe becoming more interconnected through the internet leading to the mixing of languages and blending across cultures and regions. With this, Bilingual textual data is being produced at a high rate and will only increase with the further expansion of the internet. Bilingual textual data has produced a unique problem of accurately analyzing the text through the use of 2 or more languages. This has created a new Natural Language Processing genre, Linguistic code-switching.

Linguistic code-switching is using two or more languages/dialects in conversation. While it has been studied extensively in psycho-linguistics and socio-linguistics [1, 2, 3], it is still in its early stages when it comes to the analysis of the text being automated. This is where automated Language Identification comes in to aid the task of having technology in place to automate the processing of the data to prepare it for data analysis due to the large amount of Code Switched data available. However, it can not be processed timely / efficiently because the current techniques are primarily by hand. It has become necessary to develop tools that will automate this process.

In this paper, we conduct 2 Experiments a single model for LID to classify code-switch data language labels and a novel task that can distinguish between language pairs. Language Pair Identification is a novel approach to analyzing a code-switched sentence and telling what specific language pair is within the sentence. This can be key to an approach of data where the pair is unknown and be utilized to analyze the sentence with the proper language pair-model. For both tasks, we utilize current state-of-the-art transformer models to baseline their performance on both tasks and to see their applicability to these tasks.

The following section reviews relevant research on the LID task and details the proposed Language Pair Identification task. This paper seeks to help advance and contribute to the NLP and code switched data analysis by evaluating the performance of current state-of-the-art transformer models and methods across LinCE [4] provided datasets. We end with a presentation of our results and a brief discussion of limitations and future work.

II. Relevant Works

While there is sufficient work on the sentiment analysis task with code-switched text [5, 6], the following relevant work discussed will be limited to the Language Identification Task.

A. Language Identification (LID) Task

Language Identification (LID) is critical in handling code-switched data since it is the initial step in determining whether a system can effectively process code-switched data. By correctly classifying the language associated with individual

(*) means contact author
Also thanks to be added upon acceptance.

2769-5654/23/$31.00 ©2023 IEEE
DOI 10.1109/CSCI62032.2023.00036

tokens, LID enables higher-level applications that require general language understanding to process code-switched text.

While throughout the years, there have been a lot of iterations and similar data sets based on code-switched data, specifically LID. The constant changes in datasets used are due to the field being in its early stages. It needed more structure to properly support the advancements in code-switched analysis to have models that can have generalized performance ready for real-world use.

Recently LinCE [4] has been able to help solve this by providing a centralized benchmark and a strong corpa for LID, Parts-of-Speech (POS) Tagging, Named Entity Recognition (NER), Sentiment Analysis (SA), and Machine Translation (MT) with an extensive and growing amount of Language Pair datasets that are appropriately labeled and balanced. With them, they have allowed research to find best-performing models for specific tasks and allow methods to be applied across different language pairs to demonstrate if the methods work in general or are language specific.

Specifically for LID, LinCE [4] has been able to elevate the task in that they were able to provide four datasets that are comparable to each other and as balanced as a dataset of this nature could get. The corpora follow the CALCS LID label scheme, which includes labels for lang1, lang2, mixed (text containing both languages partially), ambiguous (text in either one or the other language), fw (a language different than lang1 and lang2), ne (named entities), other, and unk (unrecognizable words). LinCE datasets were essential in advancing the task since it allows models to show whether they have generalized performance. The datasets provided were all pre-existing but used better data splits to demonstrate a model's performance truly.

B. Contemporary Models and Techniques for LID

According to the LinCE leaderboards, the current best-performing models for the LID task all use some pre-trained BERT(Bidirectional Encoder Representations) [7] finetuning with and its other models based on the BERT structure such as XLM-Roberta [8]. This technique has yielded the best performance compared to previous techniques of developing deep learning model networks. This is because BERT has been proven to have outstanding generalized performance on the whole gamut of natural language tasks due to its ability to develop a deep understanding of the text provided to it and its pattern recognition within the text. According to LinCE leaderboards, the best-performing model is XLM-Roberta. This section will not cover it due to its anonymity, so we cannot cover how they did it. We will use the two well-documented models on the leaderboards, which include Char2Subword [9] and Much Gracias [10].

1) Char2Subword [9]: is ranked two on the LID task, where they were able to propose a char2subword module that would expand mBert (multilingual BERT) ability to generate word embeddings without the restriction of models fixed vocabulary. This was done by replacing BERT's subword embedding table with a technique that would be able to take in words at a character level instead of an word level allowing for the embeddings to be vastly more robust in that it can account for misspellings, punctuation's, and more word level variance that is mitigated by character level tokenization. While char2subword is not only for the task of LID, it did prove beneficial by being able to generate the defacto best-performing model in the LID task that is documented with an overall weighted F1 of 95.48 across all datasets of Spanish-English (SPA-ENG), Hindi-English (HIN-ENG), Nepali-English (NEP-ENG), and Modern Standard Arabic-Egyptian Arabic (MSA-EA). Its per language pair performance is seen in the table below.

Language Pair	Weighted F1
Overall	95.48
SPA-ENG	98.33
HIN-ENG	96.23
NEP-ENG	96.19
MSA-EA	91.19

TABLE I: Char2Subword [9] Weighted F1 Performance

In table I, it showcases the performance of the char2subword model in the Language Identification (LID) task has been exceptional, positioning it as the top-performing model. With an impressive overall weighted F1 score of 95.48 across all datasets, including Spanish-English (SPA-ENG), Hindi-English (HIN-ENG), Nepali-English (NEP-ENG), and Modern Standard Arabic-Egyptian Arabic (MSA-EA), char2subword has demonstrated its efficacy in accurately identifying languages in code-switched text.

2) Much Gracias [10]: is the next highest-ranked documented model in the Top 7 on the leaderboard. Much Gracias also provides a novel approach, comprehensively evaluating various models for semi-supervised language identification in English-Spanish code-switched data. The models investigated include word uni-grams, word n-grams, character n-grams, Viterbi Decoding, Latent Dirichlet Allocation, Support Vector Machine, and Logistic Regression. Overall, the study demonstrates promising results across most models, highlighting their potential for this task. Among the evaluated models, the Viterbi decoding model stands out as the top performer, achieving a weighted F1 score of 95.76% on the validation data and 92.23% on the test data (RQ1). Their Viterbi decoding approach, involved tackling the problem of code-switching by modeling, seeing it as a Hidden Markov Model (HMM). They recognized that a sentence can be viewed as a Markov chain with hidden states representing the two languages involved in code-switching. To assign language labels (states) to each word (observation), they employed the Viterbi decoding algorithm [11]. They utilized an implementation of the Viterbi algorithm by Eginhard. They conducted a grid search on the development set to optimize their model.

While the performance is not state of the art, it offers a valuable takeaway highlighting the advantages of more

straightforward and faster approaches, such as the models examined in this study. Their approaches prove advantageous when top performance is not crucial. They enable researchers to avoid the labor-intensive human annotation process and the extensive training time required for fine-tuning large transformer models on supervised data.

III. Proposed Task: Language Pair Identification

This section offers a comprehensive exposition of our novel proposed task known as Language Pair Identification. Providing an introduction to the task, the rationale behind its development, datasets utilized, and its potential applications, especially in the realm of natural language processing and code-switched text analysis.

A. Introduction to Language Pair Identification (LPI)

Language Pair Identification (LPI) is a natural language processing task we propose that involves determining the specific language pairs used in a text that exhibits code-switching. Multiple languages or dialects are seamlessly integrated within code-switched text, and LPI aims to identify which language pairs are involved in the written content. The foundation for this task is rooted in the study presented in [12]. In this research, a language identification model was employed to discern the language of a text. Subsequently, this information was leveraged to select a language-specific model, leading to a performance enhancement of 4% universally in sentiment analysis for African languages. Given the parallels between Code-Switched and African Language NLP research, both characterized by being low-resource challenges, LPI is poised to assist in various facets related to Code-Switched corpora and classification performance.

B. Rationale for LPI

One might wonder why there is an emphasis on identifying language pairs in code-switched text. The reasons are multifold. While numerous multilingual models exist in the realm of NLP, their performance often needs to improve compared to models explicitly trained on individual languages, particularly for intricate tasks like sentiment analysis. By zeroing in on the exact language pairs in code-switched text, we can harness the power of models tailored to these specific languages, enhancing the accuracy and effectiveness of analysis. Furthermore, LPI stems from a need to automate and streamline traditionally manual, time-consuming, and error-prone processes. An example is how the datasets from LinCE [4] needed manual categorization.

C. Datasets for LPI

Initially, for the LPI task, we will primarily draw upon the previously mentioned LinCE datasets. These datasets serve as the foundation for our task, providing robust examples of code-switched text from which our model can learn. In the future, as the corpus of code-switched data expands, we plan to incorporate examples that go beyond those involving English in the code-switching.

To enhance the scope and ensure the model's ability to distinguish between code-switched and monolingual text, a segment of English text will be integrated. The English only text will be sourced from Sentiment140 [13] dataset which includes over 1.6 million tweets labeled by sentiment. Sentiment140 was selected due to its extensive volume of data and its resemblance to the origins of the LinCE datasets. This inclusion is crucial for LPI since it will showcase a model's ability to discern between traditional monolingual text and the complexities of code-switched content.

Furthermore, it is worth noting that our utilization of the LinCE datasets will be specifically from its LID task. However, instead of maintaining the original LID token labels, these will be removed to represent the respective language pairs from which each dataset originates.

D. Applications of LPI

The potential applications of LPI are wide-ranging and pivotal for advancing code-switched text analysis. LPI can lead to integrating multiple models trained distinctly on identified language pairs. This would circumvent the limitations posed by broader multilingual models. Moreover, LPI can significantly expedite and automate the creation and categorization of datasets akin to the ones from LinCE. Doing so paves the way for a more robust and refined data infrastructure, which in turn can significantly elevate the quality of analysis. In essence, LPI does not just offer a solution to a niche problem; it can significantly enhance how we approach and understand code-switched text.

IV. Methodology

The following subsections will cover the Datasets, Experiments, Pre-processing, Modeling, and Evaluation where all Experiments were completed on a labmda laptop or a Google Collaboratory Notebook. It is highly recommended to complete the Experiments. An GPU with large amounts of VRAM will vastly speed up the training/testing times of the Experiments.

A. Datasets

This section will describe the four datasets; Spanish-English (SPA-ENG), Hindi-English (HIN-ENG), Nepali-English (NEP-ENG), and Modern Standard Arabic-Egyptian Arabic (MSA-EA) used for the LID task. All the datasets follow the format of CALCS LID label scheme, which is comprised of lang1, lang2, ambiguous (can be one or the other language), mixed (combination of both languages), fw (a language that is different from lang1 and lang2), ne (named entities), other, and unk (unrecognizable words). The datasets comprise the majority of lang1, lang2, ne, and other labels. An additional dataset is covered for the Language Pair Identification task, which is the English dataset. While it is not part of the LID Task, the dataset is English sentences from Twitter which is needed for Experiment 1.

1) Spanish-English: The Spanish-English dataset comes from the 2016 CALCS workshop[14] where it is comprised of Twitter data containing 32,651 posts totaling up to 390,953 tokens. LinCE slightly modified the dataset to improve the data splits to further balance the dataset.

2) Hindi-English: The Hindi-English dataset comes from [15], comprising Twitter and Facebook data containing 7,421 posts totaling up to 146,722 tokens. LinCE also modified this dataset due to the combination of Twitter and Facebook data, leading to the Facebook data being considerably longer than the Twitter data, which was modified to account for that.

3) Nepali-English: The Nepali-English dataset comes from the 2016 CALCS workshop [16], comprising of Twitter data containing 13,011 posts totaling up to 188,784 tokens. LinCE modified the dataset to generate a dev/validation split since the original dataset only includes train and test splits.

4) Modern Standard Arabic-Egyptian Arabic: Modern Standard Arabic-Egyptian Arabic dataset comes from the 2016 CALCS workshop [14] where it comprises Twitter data containing 11,243 posts totaling up to 227,354 tokens. LinCE slightly modified the dataset to have improved label distribution between the splits.

5) Sentiment140: Sentiment140 is from [13] comprising all Twitter posts that was sourced similar to the LinCE datasets, in it being social media text. This dataset contains over 1.6 million posts, vastly more than the LinCE datasets. To ensure the balance, we randomly pick the equivalent amount of tweets as the other four datasets' sentence count combined for each data split of train/validation/test.

B. Experiments

This section will detail the two steps of our proposed algorithm approaches in the LID task: the codeswitched language pair identification and the conventional LID Task. These models should be implemented to work together.

C. Pre-processing

The Pre-Processing of both tasks utilized Huggingface's library [17] for tokenizing and finetuning. They differ in preprocessing since Language Pair Identification is sentence-level classification, and Single Model LID TASK is token-level classification.

1) Language Pair Identification: In the process of preparing the dataset for our study, we sourced our primary data from the LinCE LID dataset. This dataset, in its original format, consists of individual tokens representative of fragmented sentence components. To construct meaningful and coherent sentences, we adopted a systematic approach of sequentially aggregating these tokens. Each token was concatenated using a space delimiter, effectively reconstructing the original sentence structures.

To facilitate the training phase and enable our model to discern between different language combinations, a critical preprocessing step was introduced. Each reconstituted sentence was then labeled based on its originating dataset. For example, sentences that were derived from the NEP-ENG dataset were duly assigned the 'NEP-ENG' label. Such a labeling schema serves to provide a clear mapping of each sentence to its linguistic origin, which is crucial for the LPI task.

Recognizing the importance of challenging the model's discriminatory capacities, we further enriched our dataset. An English-only dataset was integrated into our corpus. The rationale behind this integration was two-fold: firstly, to augment the volume and diversity of our training data, and secondly, to ensure that our model possesses the capability to effectively distinguish between monolingual English sentences and those that manifest code-switching phenomena. This enhancement aims to equip our model for a more realistic linguistic environment, where it may encounter a broad spectrum of sentence structures and linguistic combinations.

2) Single Model LID TASK: In the tokenization phase, a challenge often encountered is the alignment between tokens and their corresponding labels. Specifically, certain words, when tokenized, may yield multiple subword units, resulting in a potential misalignment between the number of tokens produced by the tokenizer and their associated labels. This misalignment, if left unchecked, can introduce significant noise into the data and compromise the integrity of the learning process.

To address this challenge, we adopted a strategic approach during token processing. Instead of indiscriminately labeling all sub-tokens derived from a single word, only the primary or first token was assigned a label. Subsequent sub-tokens, which are essentially fragments of the original word, were deliberately ignored for the purpose of labeling. This approach ensures a one-to-one correspondence between words and their labels, thereby preserving the integrity of the labeling process.

Once this meticulous tokenization and labeling process was completed, the tokenized datasets were systematically merged to produce a cohesive and uniform dataset. Upon successful combination and rigorous quality checks, the data was deemed ready and was subsequently fed into the training model. This iterative and thorough preprocessing pipeline was instrumental in ensuring the fidelity and robustness of the model's subsequent learning phase.

D. Modeling and Evaluation

For modeling, for both Experiments, we trained two models, including xlm-roberta-large [8] and BERT-base-multilingual-cased [7]. These models were chosen due to their downstream task-leading ability and ability to understand multilingual text at a high level that other models are far from when it comes to that. They were each finetuned using Huggingface's library [17]. The model was trained using the following hyperparameters: A learning rate of 2e-5, a batch size of 64 samples both for training and validation and the training process spanned over 5 epochs. Additionally, to ensure the model remains generalized and doesn't overfit, we incorporated a weight decay of 0.01. These parameters were chosen through an

Models were evaluated with the weighted F1 metric. With the ability of a model to predict each label type. There is also

per language performance reported and overall performance across all four datasets. Since for Experiment 1, the research is novel, there is no baseline to compare to. However, for Experiment 2, we compared the scores to the char2subword, LinCE baseline performance, and Much Gracias [10] Viterbi model to see whether the single model is viable compared to language-specific models.

V. Results

A. Language Pair Identification Task Results

models	precision	recall	f1-score	accuracy
bert-base-multilingual-cased	**0.94**	**0.94**	**0.94**	**0.94**
xlm-roberta-large	0.94	0.94	0.94	0.94

TABLE II: Overall Model performance on Language Pair Identification. The Metrics are weighted average when applicable

In table II, the high performance of both the bert-base-multilingual-cased and xlm-roberta-large models is evident. These results underscore that both models have successfully handled the Language Pair Identification Task, setting a remarkably high baseline for future research in this domain. The consistently high scores across various metrics demonstrate the potential of these architectures in establishing robust standards for identifying and distinguishing between different language pairs. Establishing such a strong baseline is paramount, given the inherent complexities and nuances associated with code-switched data. This achievement validates the capabilities of the presented models and challenges future endeavors to match or surpass this established benchmark.

	precision	recall	f1-score	support
Spanish-English	0.95	0.91	0.93	8289
Hindi-English	.81	0.78	0.80	1854
MSA-EA	1.00	1.00	1.00	1663
Nepali-English	0.94	0.92	0.93	3228
English	.93	0.97	0.95	15034

TABLE III: Bert-base-multilingual-cased Per category Metrics

In table III, gives further insight into the overall performance numbers of the best performing model bert-base-multilingual-cased. Despite the model's commendable overall performance, it is evident that it encounters challenges when processing the Hindi-English language pair. This facet certainly warrants a more in-depth analysis and understanding. Furthermore, the seemingly impeccable performance exhibited by the model for the MSA-EA pair can be somewhat misleading. A closer inspection reveals that the MSA-EA pair is unique because it does not encompass English. This factor might have contributed to the inflated accuracy for this specific pair. As the corpus expands, it becomes imperative to introduce monolingual data from MSA or EA to ensure a more holistic training regime, thereby minimizing potential biases or overfitting linked to the absence of English in this pair.

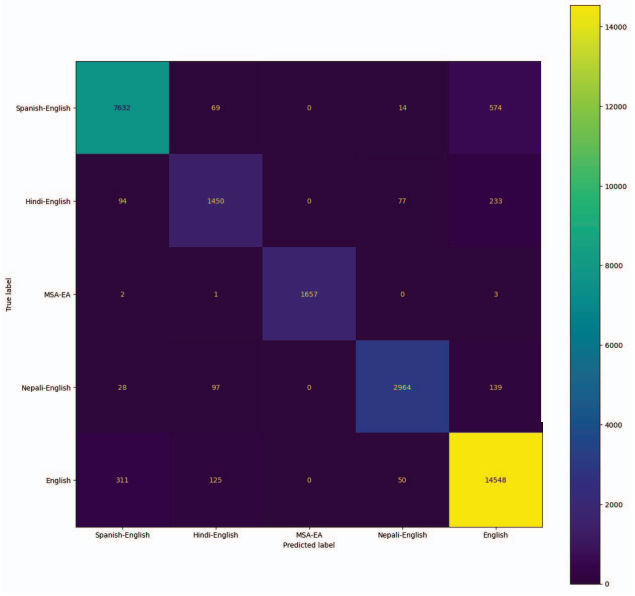

Fig. 1: Confusion Matrix for BERT-base-multilingual-cased

In table II, table III, and figure 1; shows that mBERT and XLM-Roberta are more than capable of comprehending code-switched language. While there is no baseline, 94% accuracy is high considering that the classes include English, so it can differentiate between plain English and code-switched English. This performance will only improve with the expansion of languages that do not include, as seen in MSA-EA where it could be classified perfectly. This phenomenon is seen in the confusion matrix showing that there is room for more languages, and the model has generalized performance in deciphering which code-switched pairs a sentence belongs to.

B. Language Identification Task Results

	SPA-ENG	HIN-ENG	NEP-ENG	MSA-EA	Overall
Char2subword mBERT	98.33	96.23	96.19	91.19	95.48
LinCE Organizers mBert	98.36	94.24	96.32	91.55	95.12
Much Gracias Viterbi model	92.23	NA	NA	NA	NA
bert-base-multilingual-cased	98.30	95.91	96.24	91.72	95.54
XLM-Roberta-Large	**98.63**	**96.97**	**96.70**	**93.37**	**96.41**

TABLE IV: Weighted F1 Performance for LID TASK of Token Classification Per Language Pair and Overall

Examining table IV reveals a significant advancement in our approach to the challenge. Our training regimen, which utilized a single XLM-Roberta model, succeeded in eclipsing the performance metrics of other methodologies. Notably, while other strategies on the LinCE dataset predominantly deployed language-specific models, our singular XLM-Roberta model surpassed them. Such an outcome underscores the ability of transformer models on code-switched data when presented with a diverse array of datasets. By accessing this diverse data, the model cultivates a profound comprehension of the intricacies of code-switched data and its corresponding labels. Consequently, it becomes adept at handling virtually any language pairing with commendable accuracy. To further

analyze our approach's performance, the subsequent sections examine the token-level performance.

XLM-Roberta-Large	precision	recall	f1-score	support
lang1	0.96	0.98	0.97	44675
lang2	0.97	0.96	0.97	30778
mixed	0.0	0.0	0.0	30
ambiguous	0.0	0.0	0.0	217
fw	0.0	0.0	0.0	31
ne	0.87	0.82	0.85	4892
unk	0.66	0.05	0.10	34
other	0.98	0.98	0.98	16431

TABLE V: Per Label Accuracy across all four languages across the validation set since the test data labels are not publicly available but the numbers are comparable

In table V, the model's performance at the token level remains commendable across most of the dataset. Interestingly, the instances where precision, recall, and f1-score columns report zeros indicate a lack of predictions for the categories "mixed," "ambiguous," and "fw." Several factors could contribute to this, but the limited occurrences of these categories within the dataset probably play a pivotal role. Additionally, there might be instances of term overlap wherein a word belongs to multiple categories, leading the model to favor a dominant label due to its higher prevalence, subsequently sidelining the less frequent ones. This underscores the potential inadequacies stemming from a dearth of data on these tokens. In summary, while there is a pressing need for expanded data and focused research on tokens categorized as mixed, ambiguous, fw, and unk, the current performance metrics underscore a promising utility in practical applications.

VI. Conclusion

In conclusion, we have demonstrated the efficacy of contemporary state-of-the-art transformer models in Token Level Language Identification (LID) and our newly introduced Language Pair Identification (LPI) task. For the LID task, the present state-of-the-art transformer has set a new benchmark by outperforming all existing methods on the LinCE Dataset. Hence, we advocate recognizing these transformer models as the gold standard for performance in the LID domain. Our endeavor into the Language Pair Identification task shed light on its potential to discern distinct language pairs effectively. The results from the LPI task provide insights into its potential integration into existing sentiment analysis methodologies and other tasks, facilitating the use of language pair-specific models even when the precise language pair remains unidentified. Overall, this study furthers the frontier of language processing techniques tailored for code-switched text, heralding enhanced analysis and interpretation of multilingual exchanges across a spectrum of applications.

VII. Limitations

One primary area for improvement in for our task LPI was the dependency on labeled Token Level Language Identification (LID) data. This reliance restricted the model's scope to a specific set of only four language pairs. Since MSA-EA does not have any monolingual overlap with other datasets, it is easier for the model to handle categorizing it so ensuring that data is balanced and diverse across all language pairs is imperative to prevent any biases in the model's predictions and achieve a more nuanced understanding of code-switched texts. Moreover, there was a noticeable imbalance in the distribution of token categories. Predominantly, the categories like lang1, lang2, ne, and others were heavily represented, leading to an underrepresentation of categories like unk, mixed, ambiguous, and fw. This skewed distribution consequently influenced the model's predictive behavior, often causing it to overlook or underpredict the less represented categories. Nevertheless, it is essential to highlight that despite these constraints, the datasets employed in our research successfully depicted the model's versatility across distinct language pairs. This adaptability underscores the model's potential to be effectively applied to a more expansive array of code-switched texts in future endeavors.

VIII. Future Work

Future work involves implementing an algorithm to utilize Language Pair Identification to create fusion models of language pair-specific models or to incorporate the Language Pair as a feature in a general model to let the model increase language pair-specific performance theoretically.

Additionally, Future work involves expanding the datasets, focusing on real-world settings, and exposure to more languages. Language-conscious labeling is necessary for code-switched data with multiple labels. New datasets containing over 3+ languages require language-specific labels to handle any number of languages in the text. Model hyperparameter tuning, including increased dropout, might improve token classification scores due to label distribution.

IX. Acknowledgement

This project was supported (in part) by the Office of Data Science Strategy of the National Institutes of Health under OTA 3OT2 OD032581-0151, a 2023 Amazon Research Award, and the Office of Naval Research sponsored Human Centered Artificial Intelligence Institute under grant #N00014-22-1-2714. The content is solely the responsibility of the authors and does not necessarily represent the official views of the fund.

References

[1] S. Poplack, D. Sankoff, and C. Miller, "The social correlates and linguistic processes of lexical borrowing and assimilation," 1988.

[2] S. Poplack, "Code switching: Linguistics. international encyclopedia of the social & behavioral sciences, ed. niel smelser and paul baltes," 2015.

[3] G. Sankoff, *Language use in multilingual societies: some alternative approaches*. Penguin Books, 1972.

[4] G. Aguilar, S. Kar, and T. Solorio, "LinCE: A Centralized Benchmark for Linguistic Code-switching

Evaluation," in *Proceedings of The 12th Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, May 2020, pp. 1803–1813. [Online]. Available: https://www.aclweb.org/anthology/2020.lrec-1.223

[5] S. K. Aryal, H. Prioleau, S. Aryal, and G. Washington, "Baselining performance for multilingual codeswitching sentiment classification," *Journal of Computing Sciences in Colleges*, vol. 39, no. 3, pp. 337–346, 2023.

[6] S. K. Aryal, H. Prioleau, and G. Washington, "Sentiment classification of code-switched text using pre-trained multilingual embeddings and segmentation," in *CS & IT Conference Proceedings*, vol. 12, no. 20. CS & IT Conference Proceedings, 2022.

[7] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018. [Online]. Available: http://arxiv.org/abs/1810.04805

[8] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," *CoRR*, vol. abs/1911.02116, 2019. [Online]. Available: http://arxiv.org/abs/1911.02116

[9] G. Aguilar, B. McCann, T. Niu, N. F. Rajani, N. S. Keskar, and T. Solorio, "Char2subword: Extending the subword embedding space from pre-trained models using robust character compositionality," *CoRR*, vol. abs/2010.12730, 2020. [Online]. Available: https://arxiv.org/abs/2010.12730

[10] D.-M. Iliescu, R. Grand, R. van der Goot, and S. Qirko, "Much gracias: Semi-supervised code-switch detection for spanish-english: How far can we get?" in *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*. Association for Computational Linguistics, 2021, p. 65.

[11] G. D. Forney, "The viterbi algorithm," *Proceedings of the IEEE*, vol. 61, no. 3, pp. 268–278, 1973.

[12] S. Aryal and H. Prioleau, "Howard university computer science at semeval-2023 task 12: A 2-step system design for multilingual sentiment classification with language identification," in *Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023)*, 2023, pp. 2153–2159.

[13] T. Sahni, C. Chandak, N. R. Chedeti, and M. Singh, "Efficient twitter sentiment classification using subjective distant supervision," in *2017 9th International Conference on Communication Systems and Networks (COMSNETS)*, 2017, pp. 548–553.

[14] G. Molina, F. AlGhamdi, M. Ghoneim, A. Hawwari, N. Rey-Villamizar, M. T. Diab, and T. Solorio, "Overview for the second shared task on language identification in code-switched data," *CoRR*, vol. abs/1909.13016, 2019. [Online]. Available: http://arxiv.org/abs/1909.13016

[15] D. Mave, S. Maharjan, and T. Solorio, "Language identification and analysis of code-switched social media text," in *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*. Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 51–61. [Online]. Available: https://aclanthology.org/W18-3206

[16] T. Solorio, E. Blair, S. Maharjan, S. Bethard, M. T. Diab, M. A. Ghoneim, A. Hawwari, F. A. Alghamdi, J. Hirschberg, A. Chang, and P. Fung, "Overview for the first shared task on language identification in code-switched data," in *CodeSwitch@EMNLP*, 2014.

[17] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, and J. Brew, "Huggingface's transformers: State-of-the-art natural language processing," *CoRR*, vol. abs/1910.03771, 2019. [Online]. Available: http://arxiv.org/abs/1910.03771