Lost in Translation? Benchmarking LLMs' Response to English-Arabic Code-Switching

Chiemeka Nwakama*, **Taha Alnasser**†, **Lily Li**‡, **Ibrahim Ismail-Adebiyi**§
Saint Lingual

Abstract

Despite impressive recent advancements, most LLMs still primarily handle monolingual input, struggling to adapt when speakers fluidly alternate languages within a single sentence, an everyday phenomenon known as intra-sentential code-switching. This limitation is particularly pronounced for typologically distinct language pairs, such as English and Arabic, for which high-quality, human-verified evaluation resources remain scarce. To address this gap, we introduce a carefully curated English-Arabic code-mixed dataset generated from monolingual parallel sentences and rigorously validated by native-speaking annotators. Leveraging this novel dataset, we empirically evaluate the translation performance of three prominent language models, MBart-50, Phi-3.5-mini-I, and GPT-4.1, using a comprehensive multi-metric framework (BLEU, Cosine Similarity, and BERTScore). Our findings reveal critical insights into the strengths and shortcomings of current models when confronted with realistic intra-sentential code-switching scenarios, shedding light on their ability to navigate linguistic boundaries, preserve semantic meaning, and maintain grammatical coherence. This research underscores the importance of accommodating multilingual and multicultural linguistic practices within language technologies, highlighting clear pathways toward enhancing model robustness in global, multilingual contexts.

1 Introduction

Despite major strides in natural language processing, many language models remain optimized for monolingual input. This presents a disconnect from real-world linguistic behavior, where multilingual speakers frequently switch between languages, often within a single sentence. Such code-switching introduces complexities in syntax, semantics, and vocabulary that current systems are not well-equipped to handle. This project explores how state-of-the-art models respond to these challenges by focusing on English-Arabic code switching, a relatively underexplored pairing. We constructed a code-mixed dataset featuring intra-sentential switches between English and Arabic, along with human-verified translations in both languages. Using this resource, we evaluated the translation capabilities of three prominent models: MBart-50, Phi-3.5-mini-I, and GPT-4.1. The research we did on the models highlighted their strengths and shortcomings in multilingual, code-switched scenarios.

1.1 Motivation

Over half of the world's population speaks more than one language, and for many, code-switching is a natural and frequent part of everyday communication, particularly in multilingual or multicultural environments (Aryal et al., 2022). People code-switch for a variety of reasons: to better express themselves when one language falls short, to signal group identity or solidarity, or to reflect a shift in tone or attitude. While these switches are often seamless and intuitive for multilingual speakers, they can appear unpredictable or confusing to monolingual observers–and, by extension, to language models not trained to handle such input.

If AI systems and LLMs are to serve a truly global population, they must be able to understand and generate code-switched language. Without this capability, these systems risk excluding or misrepresenting a significant por-

*nwaka013@umn.edu
†alnas023@umn.edu
‡li002398@umn.edu
§ismai128@umn.edu

tion of real-world language use. Importantly, code-switching is not random; it follows syntactic, semantic, and sociolinguistic patterns that can, in principle, be learned.

By benchmarking LLMs on their ability to process code-switched language, particularly between typologically distinct languages like English and Arabic, we can begin to evaluate whether these models truly grasp the nuances of multilingual communication. Such evaluation is essential for understanding whether models can preserve meaning across language boundaries, adapt to shifting grammatical structure mid-sentence, and produce coherent outputs that reflect the complexity of real human speech.

1.2 Background

A growing body of research highlights the unique challenges posed by Arabic-English code-switching, especially in tasks involving language identification, speech recognition, and translation. One key area of focus is intra-word language identification, which is particularly common in Arabic-English contexts. The paper Language Identification of Intra-Word Code-Switching for Arabic–English presents a dataset sourced from Twitter, Facebook, and WhatsApp and applies models such as SegRNN to tackle this phenomenon (Sabty et al., 2021). The authors report high performance, 94.84% F1 for tagging and 99.17% for segmentation, demonstrating the value of context-aware approaches. Moreover, embedding techniques like FastText and the Multilingual Universal Sentence Encoder significantly improved mixed-word detection (F1-score of 81.45%). These findings underscore the importance of content-aware evaluation methods that go beyond surface-level metrics when assessing model understanding of code-mixed language.

In addition to textual data, spoken interactions are a critical domain where code-switching naturally occurs and presents further challenges for NLP systems. The ZAEBUC-Spoken corpus, introduced in ZAEBUC-Spoken: A Multilingual Multidialectal Arabic-English Speech Corpus, contains 12 hours of Arabic-English Zoom conversations, with 19% featuring code-switching (Hamed et al., 2024). Analysis shows a notable drop in NLP performance when models process code-mixed speech. These shortcomings emphasize both the need to understand why LLMs struggle with mixed inputs and the lack of robust, diverse data in this space. Our work responds to this need by applying a diverse set of evaluation metrics to a newly developed English-Arabic code-mixed dataset, aiming to expose specific limitations and guide future research.

Similar challenges are echoed in the Mixat dataset, which contains 15 hours of Emirati-English speech with 36% code-switched utterances (Ali and Aldarmaki, 2024). The study reports that models such as Whisper, MMS, and ArTST perform poorly on these inputs, with WER and CER often exceeding 90%. These results reinforce the difficulty LLMs have in processing mixed-language content and highlight the pressing need for deeper, more nuanced evaluation strategies. In response, our work leverages multiple benchmarking metrics to better capture model performance on code-switched text, offering more insight into where current systems fall short.

Finally, the ArzEn-ST corpus, a three-way parallel dataset consisting of monolingual English, monolingual Egyptian Arabic, and code-switched transcripts, contributes another valuable resource for evaluating multilingual systems (Hamed et al., 2022). This dataset extends the ArzEn corpus and enables structured evaluations of model behavior in complex linguistic contexts. Our project builds upon this foundation by developing a new English-Arabic code-mixed dataset based on ArzEn data and systematically evaluating LLM performance using a diverse set of content-sensitive metrics. Collectively, these studies and our own contributions highlight the necessity of developing richer evaluation methods and datasets to meet the challenges posed by code-switching in multilingual NLP.

2 Approach

To assess how well language models handle code-mixed input, we evaluated their performance on translation tasks. Translation serves as a meaningful measure for language understanding; if a model can accurately translate code-mixed content, it suggests that the model

is capable of identifying which parts of a sentence belong to which language, interpreting the semantics correctly, and producing coherent outputs that preserve meaning across linguistic boundaries.

Given the complexity of translation, where multiple valid outputs can express the same meaning, we recognized that no single metric could fully capture performance. Therefore, we adopted a multi-metric evaluation strategy to gain a more comprehensive understanding. Using multiple complementary metrics allowed us to cross-check model behavior from different perspectives and identify any consistent patterns in performance.

2.1 Data Creation

To evaluate the translation capabilities of LLMs on code-mixed inputs, we first needed a dataset that included both code-mixed samples and their corresponding monolingual ground truth translations. However, no such high-quality resource existed, particularly for English-Arabic intra-sentential code-mixing. Those that did exist were AI-generated and not human-verified, putting into question the validity and quality of the datasets. We hypothesized that it was possible to construct a high-quality dataset by generating code-mixed samples from aligned English-Arabic sentence pairs and validating the outputs through native speaker review.

Our focus was specifically on intra-sentential code-mixing, as it presents more complex challenges for translation. Other forms, such as tag-switching, which involves isolated interjections or discourse markers, and inter-sentential switching, where entire sentences are in different languages, were excluded. Tag-switching has minimal impact on overall sentence meaning, and inter-sentential switching resembles standard translation tasks, which current models handle relatively well. Intra-sentential mixing, on the other hand, introduces subtle grammatical and semantic shifts that are more demanding to model and translate.

To begin, we selected the ArzEn-MultiGenre dataset, a gold-standard, manually aligned corpus of English-Arabic sentence pairs (Hamed et al., 2020). We filtered the data for language purity and sentence length, retaining only samples that were monolingual and longer than six words to ensure meaningful opportunities for code-mixing. We also removed duplicate entries, resulting in a clean base dataset of English and Arabic sentence pairs suitable for transformation.

Next, we entered the prompting phase. We engineered prompts to generate intra-sentential code-mixed sentences from each English-Arabic pair using GPT-4.1. To ensure the outputs were natural and contextually appropriate, native Arabic speakers manually reviewed random samples throughout the generation process. We designed prompts to produce both English-dominant and Arabic-dominant variants to capture a range of language ratios and evaluate how models perform under different conditions. The prompt can be found in Figure B.

In the post-processing stage, we tagged and split the outputs by language switch points. Each final entry in the dataset includes the original English sentence (monolingual), the original Arabic sentence (monolingual), an English-Arabic code-mixed version, and an Arabic-English code-mixed version.

This scalable pipeline allowed us to produce a robust and diverse set of high-quality code-mixed samples with clear ground truth references, enabling reliable evaluation of translation performance in realistic multilingual scenarios.

2.2 Model Evaluation

To assess how well language models handle code-mixed input, we evaluated their performance on translation tasks. Translation serves as a meaningful measure for language understanding; if a model can accurately translate code-mixed content, it suggests that the model is capable of identifying which parts of a sentence belong to which language, interpreting the semantics correctly, and producing coherent outputs that preserve meaning across linguistic boundaries.

Given the complexity of translation, where multiple valid outputs can express the same meaning, we recognized that no single metric could fully capture performance. Therefore, we adopted a multi-metric evaluation strategy to gain a more comprehensive understanding. Using multiple complementary metrics al-

lowed us to cross-check model behavior from different perspectives and identify any consistent patterns in performance.

2.2.1 BLEU Score

The BLEU (Bilingual Evaluation Understudy) score is a metric developed for evaluating the quality of machine-translated text by comparing it to one or more human-generated reference translations. It works by calculating the precision of n-grams in the output that match those in the reference translation (Reiter, 2018). To avoid inflated scores for repeating words, it utilizes modified precision, which limits the count of each n-gram to the maximum number of times it appears in any reference translation. Additionally, it applies a brevity penalty for translations that are shorter than the reference. The final score ranges from zero to one, with higher scores indicating a closer match to reference translations.

BLEU is mostly a surface-level evaluation, so it still has some limitations. Since it focuses on n-gram matches, it tends to struggle to account for semantic meaning and grammatical correctness. This means that if the translation uses synonyms or a different phrasing it may be penalized for it. The validity of BLEU is also highly dependent on the reference translations; the more accurate and diverse the reference translations are, the more accurate BLEU is.

2.2.2 Cosine Similarity

Cosine similarity is a metric used to measure how similar two vectors are based on the angle between them (Sravanthi and Srinivasu, 2017). Its strength lies in comparing the similarity of two texts after they have been converted into vector representations. Because it focuses on the direction (angle) rather than the magnitude, cosine similarity is better at assessing content similarities rather than surface-level features like word count. Moreover, since it works with vectorized representations of language, it can be generalized across multilingual data, assuming appropriate vectors are available. However, cosine similarity only considers the direction of the vectors and not their semantic context. If two sentences use different words with similar meanings, cosine similarity may fail to detect their relatedness unless the embeddings adequately capture the semantic relationships. It also does not account for syntax, grammar, or the order of words, as it ignores sentence structure entirely and does not consider linguistic structure or coherence. Additionally, if a vector contains only zero entries, cosine similarity is undefined or defaults to zero.

2.2.3 BERT Score

BERTScore leverages contextual embeddings from transformer models such as BERT, RoBERTa, or XLM-R, enabling it to capture meaning beyond surface-level comparisons (Zhang et al., 2019). It is adept at recognizing synonyms and paraphrasing, thus providing a better measure of semantic similarity. BERTScore performs token-level matching rather than relying on whole sentences or n-gram overlaps. It also offers strong multilingual support, particularly when used with multilingual models, making it suitable for cross-lingual and code-mixed evaluation. However, BERTScore is computationally expensive. Its scores are less interpretable compared to metrics like BLEU; it can be difficult to determine which specific words contributed to a low score or to identify particular errors. The method also depends heavily on pretrained models, so its performance may decline if the underlying model was not trained on relevant data, especially for specific or low-resource languages. While BERTScore effectively captures semantic meaning, it does not account for grammatical correctness and fluency; thus, a sentence that is semantically accurate but grammatically incorrect might still receive a high score.

3 Results and Analysis

3.1 Evaluation Metrics and Success Criteria

We measured success using a multi-metric framework that allowed us to evaluate translation quality from several angles: surface-level matching, vector similarity, and semantic understanding. Instead of relying on just a single metric, which would likely fall short in capturing the complexity of code-mixed language understanding, the evaluation used BLEU score, Cosine Similarity, and BERTScore. Our first

metric was BLEU Score assessed surface-level n-gram overlap with reference translations. Second metric used was Cosine Similarity. Cosine Similarity measured semantic similarity in embedding space. Lastly, BERTScore evaluated token-level semantic similarity using contextual embeddings, which was particularly useful for code-mixed inputs. Success of a given LLM was first determined by the averages of each of the three metrics which all have a range from 0 to 1 where 0 means not similar rather semantically or in measure depending on the metric, and 1 meaning perfectly identical semantically or in general from LLM translation when compared to the ground truth. We consider a BERTScore of < 0.60 to indicate a poor or failed translation, whereas scores 0.75 are taken as good, reliable semantic fidelity. Likewise, BLEU scores below 0.30 are unacceptable fluency and content overlap, while BLEU 0.75 denotes near–human-level translation quality. Finally, cosine similarity values below 0.50 reflect weak or irrelevant output, but values 0.75 represents good semantic equivalence with .9 being excellent nearly identical meaning between the two sentences.We accessed each average for a given metric for each LLM to give us a good idea of success. BLEU provided insight into surface-level word overlap with ground truth translations, Cosine Similarity allowed us to examine broader semantic alignment in vector space. When looking for specific failure examples, we chose to base it primarily on BERTScore, since based on our manual inspection where we randomly sampled 50 examples from all three metrics for each LLM for scores. From manual inspection, BERTScore was the closest to how a human would analyze and score how successful a translation was based on what a given BERTScore represents for that given translation. It is more sensitive to cases where the model preserved or missed the intended meaning, even if the surface wording differed. Given that code-mixing in language involves informal expressions and flexible syntax, BERTScore overall was better aligned with human judgment of semantic adequacy and proved more effective in highlighting genuine translation errors.

3.2 Research Questions

The core research questions were centered around assessing how effectively modern LLMs understand code-mixed languages, specifically sentences containing a blend of Arabic and English. Translation tasks and linguistic metrics were employed primarily as diagnostic tools to gauge the LLMs' comprehension of such code-mixed input. Specifically, we aimed to determine whether current LLMs could accurately interpret the meaning of code-mixed text through their ability to translate sentences into coherent and semantically correct outputs. Additionally, we were interested in exploring whether language dominance (Arabic-dominant versus English-dominant code-mixing) affected the models' comprehension performance. We also wanted to understand whether prominent models such as MBart-50, GPT-4.1, and Phi 3.5-mini-I differed significantly in their capacity to cognitively process code-mixed language. Lastly, we aimed to evaluate the extent to which standard translation quality metrics like BLEU, Cosine Similarity, and BERTScore effectively reflect an LLM's actual understanding of code-mixed linguistic input.

3.3 Experimental Setup

To evaluate how effectively language models handle code-mixed Arabic-English input, we conducted a series of translation experiments designed to systematically vary the proportion of each language within the input. Translation was chosen as the task because it requires not only recognizing which parts of a sentence belong to which language, but also understanding the combined semantics and generating coherent output in a single target language. This makes translation a rigorous test of multilingual and code-mixed language understanding. We selected three models for evaluation: MBart-50, Phi-3.5-mini-I, and GPT-4.1. Each model was tasked with translating code-mixed inputs into either fully Arabic or fully English. To structure the evaluation, we categorized input samples into bins based on the percentage of Arabic or English present in each code-mixed sentence. For Arabic translation tasks, inputs were divided into three bins: 0%–33% Arabic (5 samples), 34%–66% Arabic (500

samples), and 67%–100% Arabic (4,685 samples). Similarly, for English translation tasks, the inputs were grouped as follows: 0%–33% English (1,071 samples), 34%–66% English (3,485 samples), and 67%–100% English (671 samples). This binning allowed us to systematically examine model performance across varying degrees of code-mixing. Each translated output was then evaluated against ground truth references using three complementary metrics: BLEU score, to measure surface-level n-gram overlap; Cosine Similarity, to assess sentence-level semantic alignment via vector embeddings; and BERTScore, to capture token-level semantic correspondence using contextualized embeddings. This multi-metric evaluation framework provided a robust basis for comparing model outputs across different input types, capturing both syntactic fidelity and semantic adequacy. Together, these experiments offered detailed insights into how well each model copes with varying levels of language mixing during translation.

3.4 Quantitative and Qualitative Results

Using this experimental setup, the evaluation highlights clear, consistent trends across both English and Arabic translation tasks, and reveals that the greatest challenges arise when inputs are heavily dominated by the non-target language, more so than in the balanced 34%-66% bins, which one might intuitively expect to be hardest. In the English translation experiment described in Table 1, GPT-4.1's BLEU score increases steadily from 0.1732 in the 0-33 percent English bin to 0.1973 in the 34-66 percent bin and 0.2025 in the 67–100 percent bin. Phi-3.5-mini.I follow the same pattern, rising from 0.1097 to 0.1478 to 0.1926, while MBart-50 seemingly went up and then regressed with scores of 0.0753, 0.0775, and 0.0709. Cosine similarity for GPT-4.1 climbs from 0.7960 to 0.8045 and then to 0.8075. Phi-3.5-mini.I moved more modestly from 0.6718 to 0.7051 to 0.7219, and MBart-50 stayed flat around 0.5837, 0.5890, and 0.5803. On BERTScore, GPT-4.1 received an average of 0.9287 in the minority-English bin, improved to 0.9328 in the balanced bin, and held at 0.9326 in the majority-English bin, outperforming both Phi-3.5-mini.I and MBart-50 at every bin. In the Arabic translation experiment shown in Table 2, GPT-4.1's BLEU score starts at 0.1786 when Arabic comprises only 0-33 percent of the source, just below MBart-50's 0.1872, then jumps to 0.2024 in the 34-66 percent bin and to 0.2274 in the 67-100 percent bin. Its cosine similarity follows suit, rising from 0.6426 to 0.7652 and then to 0.7794 as the Arabic proportion grows. MBart-50 begins slightly higher at 0.6522 but then declines, and Phi-3.5-mini.I remains substantially lower throughout. Finally, GPT-4.1's BERTScore dips slightly from 0.8441 in the low-Arabic bin to 0.8346 in the balanced bin before recovering to 0.8430 in the high-Arabic bin, consistently outpacing the other models under all language-mix conditions. Overall, GPT-4.1 not only outperforms the other models. Unsurprisingly, all models performed poorly, far below the even established .3 threshold for poor BLEU since BLEU, unlike the other two measures, is very literal in how it scores, looking for exact matches when realistically, even for human translations, they will vary from speaker to speaker. For BERTScore, GPT-4.1's performed consistently across varying degrees of English–Arabic code-mixing reveals remarkable stability: when translating into English, with scores around .93 and scores around .84 for translating to Arabic regardless of the percentage of code-mixing. This consistency demonstrates that GPT-4.1 preserves deep contextual similarity regardless of code-switch density; at the same time, the lower BERTScore values in the Arabic-translation tasks indicate that producing fluent, semantically faithful Arabic remains a more demanding challenge for the LLM. For cosine similarity for the English translation task, GPT-4.1 remained at a high score of around 8. for all codemix bins while the Arabic translation, although MBart-50 slightly outperforms GPT-4.1 in the 0–33 percent Arabic bin (0.6522 versus 0.6426) this is one off and overall GPT 4.1 once outperformed the other two LLMs by a significant margin.

3.5 Successes, Failure Cases, and Error Sources

The study achieved partial success. We were able to confirm that modern models like GPT-4.1 can handle low to moderate levels of code-

mixing with reasonable accuracy and maintain semantic coherence in translation. Our metric framework successfully highlighted both surface-level and deeper semantic errors, allowing us to dissect performance differences across models and code-mix levels. However, we also uncovered significant limitations. All models struggled when the input contained a high degree of code-mixing, particularly when embedded languages were used in unconventional or idiomatic ways. The percentage-based token analysis we used for language classification also had shortcomings, as it sometimes misclassified the language proportion due to differences in token lengths or output verbosity. These failures largely stem from the models' lack of exposure to diverse code-mixed examples during training and the absence of code-switching-specific mechanisms in their architectures, which misclassified the language proportion due to differences in token lengths or output verbosity. These failures largely stem from the models' lack of exposure to diverse code-mixed examples during training and the absence of code-switching-specific mechanisms in their architectures.

Analysis of individual examples reveals specific instances where GPT-4.1 outperforms baseline systems, as well as cases where it falls short. For instance, in Table 3, the case of ID 700, both GPT-4.1 and MBart produce a fluent, fully-Arabic translation of the original code-switched sentence. However, MBart omits an initial pronoun drifting further from the reference and resulting in a lower BERTScore (0.67 versus GPT-4.1's >0.71). Similarly, with ID 953 (Arabic 34–66%), MBart retains the noun "Omnia" in arabic, thereby preserving unwanted code-switching, while GPT-4.1 renders the sentence entirely in Arabic, avoiding this error. In the case of ID 1495 (Arabic 0–33%), Phi-3.5 generates only a summarized fragment, prematurely ending the translation and omitting key pragmatic markers. Both GPT-4.1 and MBart succeed in preserving these elements, indicating that some errors are specific to the capacity constraints of smaller models. Conversely, there are examples where baseline systems succeed while GPT-4.1 fails. With ID 2130 (Arabic 34–66%), GPT-4.1 misinterprets the idiomatic phrase "أصبحت ضيف الله" ("I became God's guest"), erroneously treating "ضيف الله" as a personal name and hallucinating a new discourse context. This leads to a notably low BERTScore (0.52), and the error is not mirrored in MBart's outputs. In another case (ID 2247, Arabic 67–100%), GPT-4.1 fails to translate the English phrase "there are beneficial crop seeds...," likely due to incorrect detection of language boundaries within a highly mixed sentence. MBart, on the other hand, substitutes a plausible Arabic equivalent, demonstrating a relative advantage in this scenario. A notable proportion of low-scoring examples (approximately 8% of those with BERTScore <0.65 across all systems) are attributable not to translation mistakes but to mis-aligned references. ID 700 exemplifies this, where the provided reference sentence corresponds to a different source, artificially capping achievable metric scores even for perfect translations. Many apparent "unique" errors across models, therefore, reflect evaluation artefacts rather than substantive performance differences. As a result, a more rigorously aligned reference set would likely improve GPT-4.1's macro-level scores more significantly, as its outputs tend to more closely approximate literal, word-level references.

Three persistent error classes remain and are not fully addressed by current approaches. First, reference noise and metric fragility, stemming from mis-aligned sentence pairs, artificially depress scores for all models, but our methodology, focused on model improvement rather than data curation, cannot mitigate errors originating from the evaluation ground truth. Second, boundary-detection failures persist under conditions of heavy code-switching (34–100% mixing), where sentence-level decoding objectives cause GPT-4.1 to sometimes leave English fragments untranslated, and MBart to misorder constituents. Such errors are best addressed by training with more granular, token-level language identification or by integrating constrained decoding mechanisms that enforce target-language vocabulary compliance. Third, smaller models such as Phi-3.5-mini exhibit classic degeneration phenomena such as early stopping, where decoding ends prematurely, or semantic drift,

primarily due to limited context windows and vocabulary size. Remedies for these issues would require scaling up model capacity or targeted fine-tuning, as they are not solvable by evaluation-time prompts or decoding strategies.

4 Discussion

4.1 Dataset Construction and Replicability

Our study is highly replicable due to the comprehensive dataset we developed and the transparent, step-by-step methodology we employed. The dataset consists of English-Arabic code-mixed samples generated via a prompt-based approach, using aligned monolingual pairs from the ArzEn-MultiGenre corpus as the foundation. This method, while tailored to English-Arabic, is generalizable to other language pairs, provided sufficiently high-quality bilingual resources and native speakers are available for manual validation. Both English-dominant and Arabic-dominant code-mixed variants were produced for balanced coverage. Each final entry in the dataset includes aligned monolingual sentences in both languages, their respective code-mixed forms, and explicit language segmentation, enabling controlled and fine-grained evaluation.

All code for data generation, prompt engineering, validation interface, and evaluation is openly documented and provided, making it straightforward for other researchers to reproduce the dataset or extend it to additional language pairs or domains. Our evaluation methods, including BLEU, Cosine Similarity, and BERTScore, follow standard practices, and our configuration is provided in sufficient detail for direct comparison or replication. It is important to note that, while our pipeline is transparent and replicable, results on models such as MBart-50, Phi-3.5-mini-I, and GPT-4.1 may vary slightly across runs due to the inherent nondeterministic outputs of LLMs. However, the overall trends should remain robust, and the methodology provides a foundation for further benchmarking.

4.2 Ethical Considerations

From an ethical standpoint, our work leverages real conversational examples from an existing dataset where user consent was previously obtained, minimizing privacy-related risks. Moreover, because our current contribution is confined to dataset creation and model evaluation without real-world system deployment, direct societal risks or harms remain minimal at this stage. However, ethics in NLP also extend to inclusivity and representational fairness in multilingual systems. Our work explicitly addresses this aspect by developing and evaluating resources and models that better reflect how multilingual speakers communicate naturally, particularly those engaging in English-Arabic code-switching, an underrepresented language pairing. Such efforts help to ensure that LLMs can serve diverse populations more equitably, avoiding biases and exclusion of communities frequently engaging in code-switched communication.

4.3 Limitations and Future Directions

Our primary limitation is the reliance on AI-generated examples derived from existing parallel sentences, which may inadvertently introduce subtle inconsistencies or unnatural phrasing, despite manual validation steps. Expanded and continuous human verification would greatly enhance the data's naturalness, authenticity, and overall reliability. In addition, we evaluated model performance on only three well-known LLMs (MBart-50, Phi-3.5-mini-I, GPT-4.1), limiting the generalizability of our conclusions. To bolster these findings, future research should incorporate a wider range of models, benchmarks, and performance metrics, as well as explore extended conversational contexts beyond single-sentence translations to reflect more richly realistic multilingual discourse. Developing larger-scale, rigorously human-validated datasets, refining data generation and annotation protocols, incorporating diverse sociolinguistic contexts, and expanding this approach to additional language pairs will further strengthen code-switched NLP research and support robust, inclusive language technologies.

5 Conclusion

Our study presents a systematic evaluation of leading language models, GPT-4.1,

MBart-50, and Phi-3.5-mini-I, on the complex task of translating English-Arabic code-mixed text. By leveraging a novel, expertly validated dataset and a multi-metric assessment framework (BLEU, Cosine Similarity, and BERTScore), we reveal both strengths and persistent weaknesses in current multilingual models. While advanced models like GPT-4.1 demonstrate notable semantic stability and maintain contextual fidelity across varying levels of code-mixing, all models exhibit clear limitations when faced with heavy or unconventional code-switching, boundary detection challenges, and the nuances of idiomatic language. These findings are further complicated by reference data misalignment and the inherent rigidity of traditional evaluation metrics. Our research highlights a critical need for more robust datasets, improved code-mixed language handling in training, and refined metrics that better align with human judgment. Addressing these challenges will be essential to advancing language technologies that are inclusive of real-world multilingual and multicultural LLM practices.

Acknowledgments

We acknowledge the use of OpenAI's ChatGPT in the preparation of this manuscript. ChatGPT was used to assist with proofreading of the text and to provide guidance and suggestions for programming code (OpenAI, 2025). All final content was critically reviewed and verified by the authors.

References

Gustavo Aguilar, Sudipta Kar, and Thamar Solorio. 2020. Lince: A centralized benchmark for linguistic code-switching evaluation.

Meta AI. 2025. Mbart-50 (may, 2025 version). Multilingual sequence-to-sequence translation model.

Maryam Al Ali and Hanan Aldarmaki. 2024. Mixat: A data set of bilingual emirati-english speech.

Saurav K. Aryal, Howard Prioleau, and Gloria Washington. 2022. Sentiment classification of code-switched text using pre-trained multilingual embeddings and segmentation.

Injy Hamed, Fadhl Eryani, David Palfreyman, and Nizar Habash. 2024. Zaebuc-spoken: A multilingual multidialectal arabic-english speech corpus.

Injy Hamed, Nizar Habash, Slim Abdennadher, and Ngoc Thang Vu. 2022. Arzen-st: A three-way speech translation corpus for code-switched egyptian arabic - english.

Injy Hamed, Ngoc Thang Vu, and Slim Abdennadher. 2020. ArzEn: A speech corpus for code-switched Egyptian Arabic-English. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4237–4246, Marseille, France. European Language Resources Association.

Holy Lovenia, Samuel Cahyawijaya, Genta Indra Winata, Peng Xu, Xu Yan, Zihan Liu, Rita Frieske, Tiezheng Yu, Wenliang Dai, Elham J. Barezi, Qifeng Chen, Xiaojuan Ma, Bertram E. Shi, and Pascale Fung. 2022. Ascend: A spontaneous chinese-english dataset for code-switching in multi-turn conversation.

Microsoft. 2025. Phi-3.5-mini.i (may, 2025 version).

OpenAI. 2025. Chatgpt (may, 2025 version). Large language model. Accessed: 2025-05-08.

Ehud Reiter. 2018. A structured review of the validity of bleu. *Computational Linguistics*, 44(3):393–401.

Caroline Sabty, Islam Mesabah, Özlem Çetinoğlu, and Slim Abdennadher. 2021. Language identification of intra-word code-switching for arabic–english. *Array*, 12:100104.

Pantulkar Sravanthi and B Srinivasu. 2017. Semantic similarity between sentences. *International Research Journal of Engineering and Technology (IRJET)*, 4(1):156–161.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

A Tables

Metric	Model	En 0–33	En 34–66	En 67–100
BLEU	MBart-50	0.0753	0.0775	0.0709
	Phi-3.5-mini.I	0.1097	0.1478	0.1926
	GPT-4.1	0.1732	0.1973	0.2025
Cosine Sim.	MBart-50	0.5837	0.5890	0.5803
	Phi-3.5-mini.I	0.6718	0.7051	0.7219
	GPT-4.1	0.7960	0.8045	0.8075
BERTScore	MBart-50	0.8891	0.8906	0.8882
	Phi-3.5-mini.I	0.9061	0.9144	0.9204
	GPT-4.1	0.9287	0.9328	0.9326

Table 1: Mean BLEU, Cosine similarity and BERTScore by English code-mixing category (percentage of English content).

Metric	Model	Ar 0–33	Ar 34–66	Ar 67–100
BLEU	MBart-50	0.1872	0.1656	0.1462
	Phi-3.5-mini.I	0.0253	0.0484	0.0391
	GPT-4.1	0.1786	0.2024	0.2274
Cosine Sim.	MBart-50	0.6522	0.7110	0.7062
	Phi-3.5-mini.I	0.5447	0.5671	0.5326
	GPT-4.1	0.6426	0.7652	0.7794
BERTScore	MBart-50	0.8255	0.7892	0.7772
	Phi-3.5-mini.I	0.7162	0.6328	0.6095
	GPT-4.1	0.8441	0.8346	0.8430

Table 2: Mean BLEU, Cosine similarity and BERTScore by Arabic code-mixing category (percentage of Arabic content).

Model	Mix Cat.	Ex. ID (BERT)	Code-mixed source	Model translation	Reference (ground truth)
MBart-50	Arabic 0–33%	700 (0.6746)	أنا عارف what they're doing وهـ prove it to you.	أعرف همّا بيعملوا إيه و هـ أثبتلك.	أنا شوفت البت بعيني. أنا عرفت مين اللي بتنده
	Arabic 34–66%	953 (0.5258)	إإيه قلة الذوق اللي انت كنت so rude to Omnia ؟ بيها	لم يكن على ما يرام إنك Omnia. تكون قليل الذوق كده مع	إيه قلة الذوق اللي إنت عاملت آلست دي بيها؟
	Arabic 67–100%	2037 (0.4388)	مش عارفة إزاي الدنيا بتشغلنا بجد اللي people إن إحنا نشوف الـ بنجبهم.	لقد كانت طويلة من ساعة ما شفتك آخر مرة.	مش عارفة إزاي الدنيا بتشغلنا بجد إن إحنا نشوف الناس اللي بنجبهم.
GPT-4.1	Arabic 0–33%	700 (0.7133)	أنا عارف what they're doing وهـ prove it to you.	أنا عارف همّا بيعملوا إيه و هـثبتلك.	أنا شوفت البت بعيني. أنا عرفت مين اللي بتنده
	Arabic 34–66%	2130 (0.5209)	اصبحت ضيف الله و I'm sure ضيف الله never gets harmed.	انت فين؟ خليك واخد بالك. Reda,	اصبحت ضيف الله وضيف الله لا يضام
	Arabic 67–100%	2247 (0.4444)	من good seeds والبتالي فيه من bad seeds الزرع المفيد وفيه الزرع المؤذي.	وبالتالي، there are beneficial crop seeds and there are harmful crop seeds.	والبتالي فيه بزور الزرع المفيد وفيه بزور الزرع المؤذي.
Phi-3.5	Arabic 0–33%	1495 (0.6392)	أه I do that بس I do that alone.	أعتني بنفسي فقط	أه بعمل كدا بس بعمل كدا لوحدي.
(mini.I)	Arabic 34–66%	3532 (0.2932)	سنانه ما كانتش الـ ordinary pyramid-shaped teeth بتوع معظم الـ sharks.	لم يكن سن	سنانه ما كانتش زي سنان معظم اسماك القرش على شكل هرم.
	Arabic 67–100%	4862 (0.1877)	مسحت العرق اللي كان مغطي وشي ... إلا لما سمعتهم بينادوا على الـ بتاع دار المسنين. warden	أن	مسحت العرق إللي كان مغطي وشي ... إلا لما سمعتهم بينادو على مدير دار المسنين.

Table 3: Three lowest-scoring translations (by BERTScore) for each model and mixing category.

B Prompts

You are a linguist specializing in Arabic-English code-switching.
Given an Egyptian Arabic sentence and its English translation,
generate two intra-sentential code-mixed outputs that reflect
commonly-used code-switched speech:

1. Arabic-dominant with English insertions
2. English-dominant with Arabic insertions

Input:
Arabic: الدكتور قال إن الحالة اتحسنت بعد العملية، بس لسه محتاجة متابعة.
English: The doctor said the condition improved after the surgery, but it still needs monitoring.

Important:
Only output the two sentences, no labels, prefixes, or explanations.
Exact format:
<Arabic -> English code-mixed sentence>
<English -> Arabic code-mixed sentence>

You are a professional translator.

Task:
Translate Input 1 from code-switched Arabic-English to fluent English.
Translate Input 2 from code-switched English-Arabic to fluent Egyptian Arabic.

Only return the two translations, one per line, in the same order.

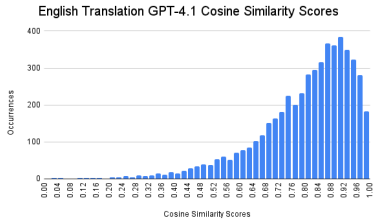
Example:
Input 1: هو was talking too fast
Input 2: I told ه مش هينفع كده

Desired output:
He was talking too fast.
قلتله مش هينفع كده

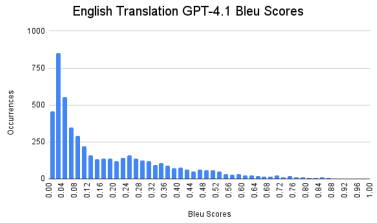
Now translate:
Input 1: {arabic_eng_code_switched}
Input 2: {english_arab_code_switched}

C Graphs

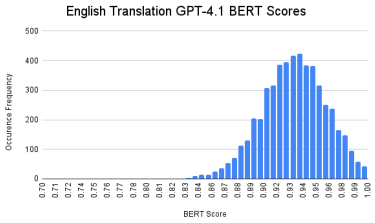
C.1 GPT-4.1



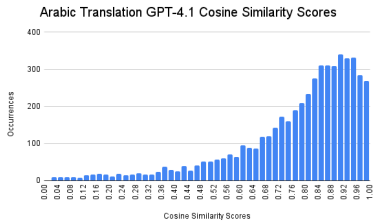
(a) Cosine Similarity



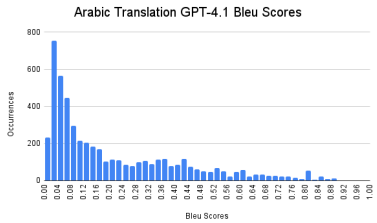
(b) BLEU Score



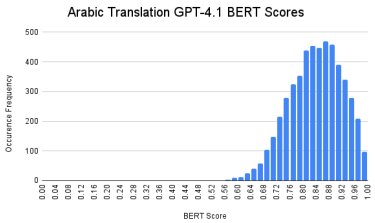
(c) BERT Score



(d) Cosine Similarity



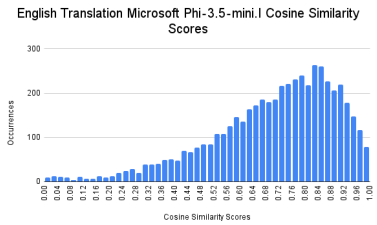
(e) BLEU Score



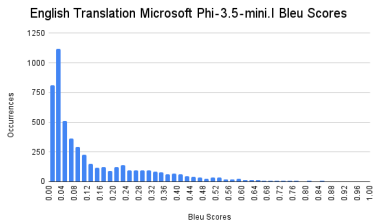
(f) BERT Score

Figure 1: GPT-4.1 translation metrics, English (top) vs. Arabic (bottom).

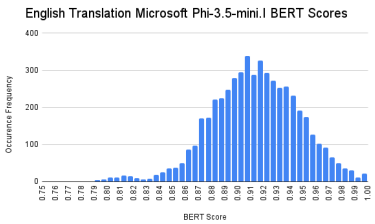
C.2 Microsoft Phi-3.5-mini.I



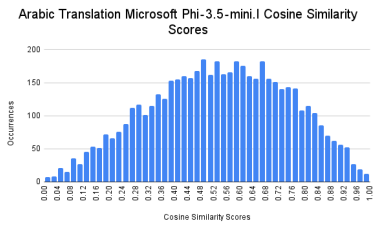
(a) Cosine



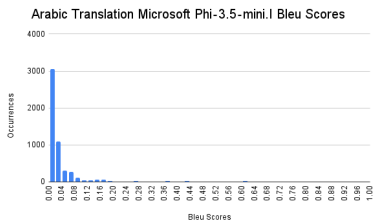
(b) BLEU



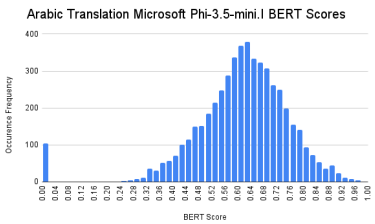
(c) BERT



(d) Cosine



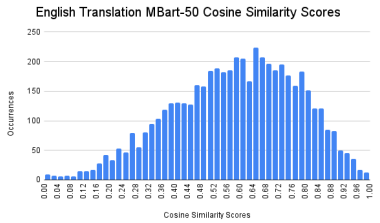
(e) BLEU



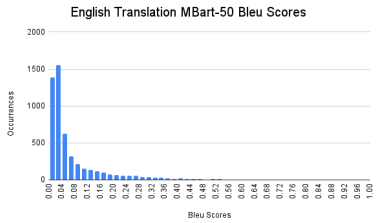
(f) BERT

Figure 2: Microsoft Phi-3.5-mini.I translation metrics.

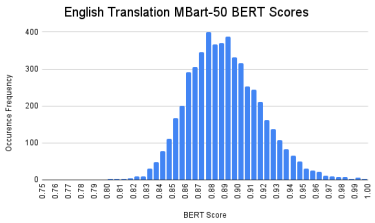
C.3 MBart-50



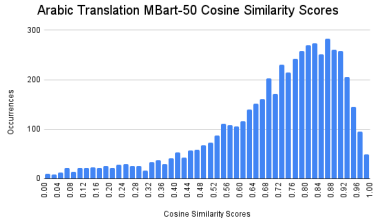
(a) Cosine



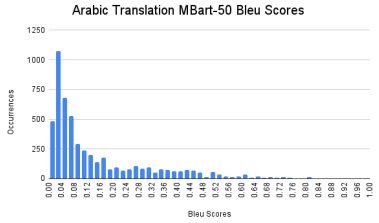
(b) BLEU



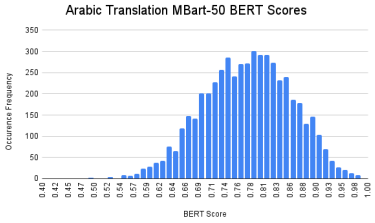
(c) BERT



(d) Cosine



(e) BLEU



(f) BERT

Figure 3: MBart-50 translation metrics.