International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Exploring Marathi-English Code-Mixing: Comprehensive Analysis of NLP Applications (QA and NER)

Dhiraj Amin*[1], Dr. Sharvari Govilkar[2], Sagar Kulkarni[3], Dr. Madhura Vyawahare[4], Shubhangi Chavan[5], Pooja Pandey[6]

Submitted: 27/12/2023 **Revised**: 03/02/2024 **Accepted**: 11/02/2024

Abstract: Code-mixing, the linguistic practice of blending elements from multiple languages, is a common phenomenon that reflects the linguistic and cultural context of speakers. This research investigates Marathi-English code-mixing, with a focus on natural language processing (NLP) applications such as question answering (QA) and named entity recognition (NER). A sophisticated Marathi-English code-mixed QA system is proposed, which can comprehend and respond to questions that span multiple languages. The effectiveness of the system is evaluated using real and synthetic code-mixed QA datasets, revealing promising results, with the MuRIL model achieving an exact match (EM) score of 0.41 and 0.62 on real and synthetic datasets, respectively. The same model, when fine-tuned for code-mixed NER on the MahaRoBERTa code-mixed NER dataset, achieves an impressive F1 score of 73.92, outperforming other models in accurately labeling named entities in code-mixed text. This research advances code-mixed language processing by addressing issues in multilingual communication contexts.

Keywords: *Code-Mixed, Language Model, Marathi BERT, Natural Language Processing, Named Entity Recognition, Question Answering*

1. Introduction

In multilingual environments, code-mixing is a linguistic phenomenon where speakers proficient in multiple languages seamlessly blend elements from two or more languages within a single expression. This practice involves integrating vocabulary, grammar, or phrases from different languages to enhance communication effectively. Code-mixing occurs as individuals draw upon their language skills to convey meaning, addressing concepts that might lack a precise equivalent in one language by incorporating words from another. In Figure 1, a Marathi speaker may opt for "amazing" instead of the Marathi term "आश्चर्यकारक" (Āścaryakāraka) when referring to a bowler. This choice enables them to convey ideas more precisely by selecting words with specific meanings. Code-mixing enhances conversations by skilfully blending languages to cater to individual communication preferences.

Natural Language Processing (NLP) stands out as a trailblazing influence in technological advancement, featuring a broad spectrum of applications. These include tasks such as question answering and named entity recognition, extending to document summarization, sentiment analysis, speech recognition, and text generation. The versatility of NLP significantly contributes to the evolution of communication, automation, and information management across diverse domains

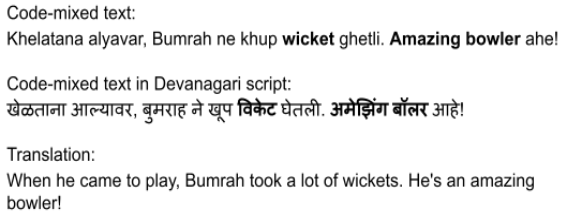

Fig. 1. Example of Marathi-English Code-mixed text in Latin and Devanagari script

A sophisticated linguistic technology known as a code-mixed Question Answering System (QAS) has the capability to understand and respond to questions that involve a combination of multiple languages. This system employs natural language processing techniques to adeptly interpret questions and contexts that may span different languages. Consequently, it can furnish accurate and coherent answers, effectively bridging language barriers and accommodating multilingual communication scenarios. This enhances accessibility and comprehension across

[1] *Department of Computer Engineering, Pillai College of Engineering, Navi Mumbai, Maharashtra, India*
ORCID ID: 0000-0002-7931-0815
[2] *Department of Computer Engineering, Pillai College of Engineering, Navi Mumbai, Maharashtra, India*
ORCID ID: 0000-0003-2722-0065
[3] *Department of Computer Engineering, Pillai College of Engineering, Navi Mumbai, Maharashtra, India*
ORCID ID: 0009-0008-5701-9059
[4] *Department of Computer Engineering, SVKM's MPSTME, Mumbai Maharashtra, India*
ORCID ID: 0000-0002-8981-7636
[5] *Department of Computer Engineering, Pillai College of Engineering, Navi Mumbai, Maharashtra, India*
ORCID ID: 0009-0006-3643-9885
[6] *Department of Computer Science and Engineering, Prestige Institute of Engineering Management and Research, Indore, Madhya Pradesh, India*
ORCID ID: 0009-0008-6229-2906
** Corresponding Author Email: amindhiraj@mes.ac.in*

diverse linguistic contexts. In scenarios using mixed-code syntax, users can submit queries. For example, a Marathi speaker might pose the question, "heart attack cinema chae director kon ahe?" or "who directed the movie heart attack?" In this case, the terms "heart attack," "director," and "cinema" are all in English. Similarly, the same question can be expressed using English phrases in the Devanagari script: "हार्ट अटॅक सिनेमाचा डायरेक्टर कोण आहे?".

Named Entity Recognition (NER) in a code-mixed context involves the identification and categorization of entities, such as names of individuals, locations, organizations, dates, and more, within sentences that blend multiple languages. This process is essential for extracting meaningful information from code-mixed text, contributing to a more profound comprehension of the content. For example, in the sentence "माझा फ्रिएन्द गूगल मध्ये काम करतो" ("My friend works in Google"), NER would pinpoint "गूगल" (Google) as an organization entity, illustrating its role in grasping mixed-language context. This capability significantly improves the accuracy of entity recognition in multilingual communication scenarios.

2. Related Work

The amount of research focused on code-mixing in Indian languages is comparatively limited when compared to other linguistic research areas, particularly in English and European languages. This imbalance can be attributed to the extensive linguistic diversity in India, where a multitude of languages and dialects are spoken. In addition to the 22 official languages acknowledged by the Indian Constitution, there exist hundreds of dialects and regional languages. However, recent times have seen a growing interest in code-mixing in Indian languages, motivating researchers to delve more comprehensively into this subject. The proposed research specifically targets textual question answering systems and named entity recognition in code-mixed (Marathi-English) versions within Indian regional languages.

Code-mixed question answering systems can be developed using multiple approaches. One approach is to translate the code-mixed questions into either the primary or secondary language, and then use a much more accurate monolingual question answering system to extract answers. Another approach is to not translate the code-mixed text, and instead use a multilingual or code-mixed pre-trained model and fine-tune it on question answering tasks on a code-mixed dataset. This approach is more challenging, but it can potentially lead to more accurate results.

Authors employ a variety of approaches to build code-mixed question answering systems. Singh et al. [1] developed a system designed for retrieving mixed-script information in code-mixed Hindi-English tweets. They employed Vector Space Models to gauge semantic similarity in their approach. The system achieved a Mean Average Precision of 0.0315, indicating its success in retrieving relevant tweets in the Code-Mixed context. Incorporating deep learning, word embedding, and TF-IDF, Kumar et al. [2] enhanced query expansion and classification algorithms for mixed-script IR outperformed a baseline model with a significant 20.44% improvement in MRR and 15.61% rise in MAP. By leveraging bilingual dictionaries to translate questions from Hinglish and Tenglish to English, Chandu et al.'s [3] proposal for a web-based Factoid QA system for those languages overcomes the lack of linguistic resources. The system's efficiency is demonstrated by evaluation findings, which show an MRR of 0.37 for Hinglish and 0.32 for Tenglish. Gupta et al. [4] developed a Hindi-English code-mixed question answering (CMQA) system that replaces Hindi named entities with English words, proposed bilinear attention and answer-type focused neural framework achieving a code-mixed question evaluation score with an EM of 40.50% and an F1 of 53.73%. Ambiguity occasionally led to incorrect predictions by the system. A flexible deep neural network strategy for multilingual question answering was presented by Gupta et al. [5]. By aligning question words from both Hindi and English, their technique enables the model to acquire a common representation of the question. The answer extraction layer then receives this shared representation along with the attention-based snippet representation and pulls the answer span from the snippet. This model achieved an F1 score of 44.97 and an Exact Match score of 39.44 on a benchmark Question Answering dataset that includes multiple languages. An online system for mixed language questions and answers was proposed by Thara S et al. [6]. The system converts user queries in Hindi, Telugu, or Tamil into English to facilitate streamlined processing. Deep learning algorithms like RNN and HAN are used for question classification and answer extraction, achieving an accuracy of 80.667%.

Code-mixed named entity recognition systems can be constructed through different strategies. One method focusses on translating code-mixed text, incorporating named entities, into either the primary or secondary language. Following the translation, a highly accurate monolingual named entity recognition system can be employed to extract entities. Another approach avoids translation altogether and utilizes a multilingual or code-mixed pre-trained model. This model is then fine-tuned on named entity recognition tasks using a code-mixed dataset. While this method presents greater challenges, it holds the potential to deliver more precise results in identifying named entities within code-mixed contexts present in the text.

Authors utilize diverse methods when constructing code-mixed named entity recognition system. Dowlagar et al. [7]

tackles the complexity of Named Entity Recognition (NER) in code-mixed text, demonstrating a 6% enhancement over the baseline through the utilization of multilingual data and pre-trained mBERT. The study highlights the constraints of statistical models, such as CRF, and recommends investigating meta embeddings, language identification, or POS tagging to enhance code-mixed NER further. Singh et al. [8] investigates the complexities of automated text analysis for Named Entity Recognition (NER) within the domain of code-mixed Hindi-English content on Online Social Networks (OSNs). Through the introduction of LSTM and CRF models, a significant enhancement is demonstrated, surpassing the performance of currently available off-the-shelf NER tools by 33.18% (F1 score). The integration of a semi-supervised language identifier further refines the NER model, offering potential advancements for downstream NLP tasks in the context of code-mixed data. Mekki et al. [9]addresses the intricate task of Named Entity Recognition (NER) in multilingual and code-mixed contexts. Employing the XLM-RoBERTa Transformer, the proposed system, integrating a CRF-based layer and span classification, exhibits substantial enhancements in the Multilingual Complex Named Entity Recognition (MultiCoNER) shared task. Following self-training, the system achieves notable F1-scores of 72.49% in the multilingual track and 79.21% in the code-mixed track, showcasing its effectiveness across diverse linguistic challenges. Srirangam et al. [10]delves into the complexities of Named Entity Recognition (NER) in Telugu-English code-mixed social media content, a crucial facet of Natural Language Processing. Employing Conditional Random Fields (CRF), Decision Trees, and Bidirectional LSTMs, the authors achieve competitive F1-scores of 0.96, 0.94, and 0.95, respectively. They introduce a unique annotated corpus in code-mixed Telugu-English, contributing to the comprehension of NER in the multilingual social media landscape. Sabty et al. [11] created the inaugural annotated Arabic-English Code-Mixed (CM) corpus for NER, comprising 6,525 sentences from various sources. Annotations adhere to the Named Entity Annotation guidelines. The study evaluates diverse pre-trained word embeddings, both classical (Word2Vec, FastText, GloVe) and contextual (ELMo, BERT, FLAIR), coupled with a BiLSTM-CRF model. The most successful model, combining Pooled embeddings and FastText, achieves a notable F1-score of 77.69%. The findings underscore the efficacy of contextual embeddings and the significance of addressing code-mixing in NER tasks on social media data. To create a meta-embedding, Ruba Priyadharshini et al. [12]used pre-trained embedding, sub-word embedding, and languages that were closely related to the languages found in the code-mixed corpus. Following that, the code-mixed sentence was encoded using the Transformer, and Named Entities were predicted using the Conditional Random Field in the code-mixed text. This methodology allows Named Entity prediction in code-mixed corpora written in both native and Roman script, in contrast to traditional Named Entity recognition approaches that usually target monolingual text.

3. Marathi-English (Minglish) Code-Mixed Question Answering System

The primary goal of the code-mixed Marathi-English question-answering system is to deliver coherent and precise natural language responses to fact-based inquiries formulated in the code-mixed Devanagari script. Given a natural language question (CMQ) in code-mixed Marathi-English text, the system must extract relevant information from various sources, including Marathi and code-mixed Marathi-English text. The system must identify the start and end positions of the answer within the passage for a specific code-mixed question. Figure 2 provides an example showcasing a code-mixed question, the context, and the corresponding output answer generated by the system. The system also supports code-mixed Marathi-English text written in Latin script by using IndicXlit [13] is a transformer-based multilingual transliteration model to transliterate into Devanagari script.

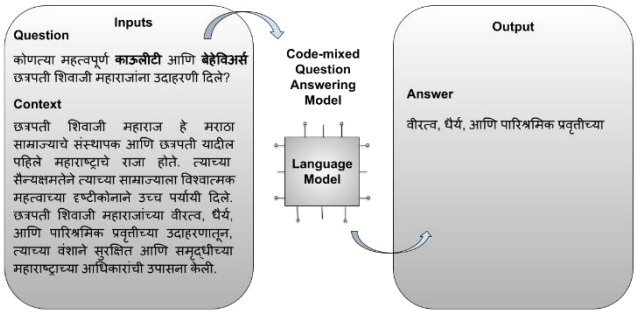

Fig. 2. Inputs and output in a Code-Mixed Question Answering System

3.1. Datasets for Code-Mixed Question Answering

The real Marathi-English Code-Mixed QA dataset (MrEnCMQA) and the synthetic Marathi-English Code-Mixed SQuAD QA dataset (MrEnCMSQuADQA) have been carefully created by leveraging the proposed streamlined codemix text generation algorithm[14]. Existing Marathi questions[15] have been skillfully transformed into Marathi-English code-mixed questions. This innovative approach has resulted in a real code-mixed QA dataset, consisting of 2499 questions that were expertly generated using question answering annotation tools, with the active involvement of crowdsource workers as. The dataset has been strategically partitioned into 1874 training question-answer pairs and 625 test question-answer pairs.

The MrEnCMSQuADQA dataset is of the same size as the pre-existing MrSQuADQA dataset [15]However, in the code-mixed dataset, the questions have been transformed into code-mixed format. Notably, while the questions have

been converted to this code-mixed format, the passage content and the answers within the dataset remain consistently in Marathi. There are 30,162 unique answers, 46,960 unique questions, and 17,337 unique contexts, in the training subset of the dataset. There are 1,922 unique contexts, 5,805 unique questions, and 4,409 unique answers in the testing subset of the dataset.

3.2. Proposed approach

Code-mixed question answering system provides users the ability to search in Marathi-English code-mixed language. Users get answers which are extracted from Marathi passages. Code-Mixed question answering is accomplished by adopting the retriever-reader approach, which first processes questions and then retrieves relevant paragraphs/contexts from different sources. The most relevant paragraph containing the answer is then extracted by ranking the relevant paragraphs. The answer reader module which is the core part for providing the answer and the focus of this research work extracts the relevant portion of text that answers the question from the passage with the highest ranking. A code-mixed question and the answer retriever module's most relevant paragraph are the two inputs that the answer reader module needs. The answer reader module generates a potential answer for the input question by extracting the answer span.

Language models like BERT can be fine-tuned to perform question answering tasks, which involves fine-tuning the existing BERT [16] model to extract the answer span for the question provided. The crucial step of fine-tuning the MuRIL [17] transformer to extract the response span for a given code-mixed question is depicted in Figure 3. MuRIL, a BERT variation, is a multilingual language model created by Google that supports 17 Indian languages besides English. MuRIL was trained on masked language modelling tasks with a maximum sequence length of 512, following a BERT base architecture. Comparing multilingual models like MuRIL to their monolingual counterparts, the latter are less capable of handling code-mixing problems. Multilingual models can capture mixed information from more than one language. This is because they have vector representations of different language words in the same space. This helps them to understand sentences which have code-mixed words. To create code-mixed answer span extractor model thus MuRIL is fine-tuned separately using real MrEnCMQA dataset and synthetic MrEnCMSQuADQA dataset.

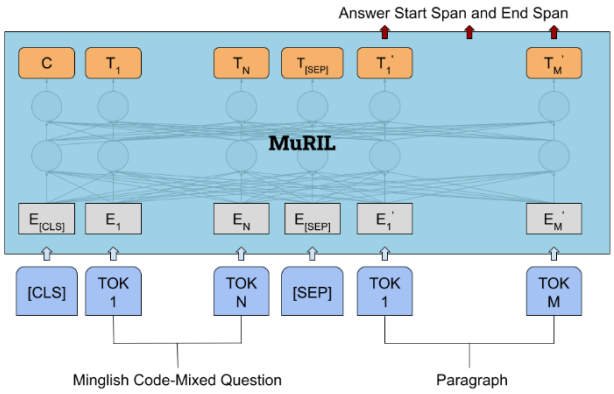

Fig 3. Fine-tuning MuRIL transformer for Code-Mixed Question Answering System

3.3. Results and Analysis

By determining the beginning and ending indices of the sequence inside the passage, the model uses an input question and text to anticipate the most accurate answer. Many of the predicted sequences might contain words that aren't in the real answers, or they might have words in them that sound similar to words in the real answers. Metrics like the F1 score, Exact Match (EM) score, and BERT score [18] are used to measure the effectiveness of extractive question answering models. These indicators aid in evaluating the model's effectiveness.

Table 1. Performance Comparison of Marathi-English Code-Mixed Question Answering Systems

Criteria	***Real MrEnCMQA Dataset***	***MrEnCMSQuADQA Dataset***
No. of Question Answer in Dataset	2,499 real questions and answers	52,897 synthetic questions and answers
Model	MuRIL	MuRIL
EM Score	0.41	0.62
BERT Score	0.88	0.92
F1 Score	0.65	0.72

Exact Match (EM) signifies a complete match between predicted and actual answers, requiring character-by-character identity. Even a single character difference results in an EM score of zero. F1 Score compares overlapping words in the predicted and actual answers, utilizing True Positive (TP) for shared words, False Positive (FP) for extra predicted words, and False Negative (FN) for missing words in the prediction. While F1 and EM lack semantic consideration, BERTScore uses contextual BERT embeddings for cosine similarity, enabling meaningful

semantic comparison between predictions and ground truth.

The Code-Mixed Answer Reader model, specialized in extracting answer spans from relevant passages, is fine-tuned through two datasets: the real Marathi-English codemix Question Answering (MrEnCMQA) dataset and the Marathi-English codemix SQuAD Question Answering (MrEnCMSQuADQA) dataset. This fine-tuning process aims to enhance the model's efficacy in question answering tasks.

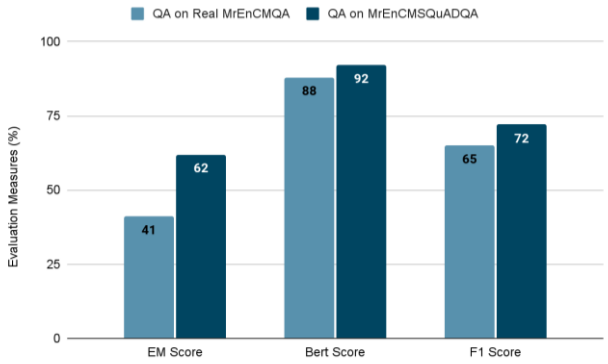

Fig 4. Comparison of code-mixed question answering system performance on real and synthetic dataset

The fine-tuned MuRIL model yields potential starting and ending indices for predicted answers across all model variations. This model was trained using a batch size of 15 and a sequence length of 512. The training process made use of high-performance GPUs available through Google Colab Pro. On the real code-mixed QA dataset known as MrSQuADQA, the MuRIL model was fine-tuned over the course of 8 epochs. This fine-tuning resulted in an Exact Match (EM) score of 0.41, a BERTScore of 0.88, and an F1 score of 0.65, as indicated in Table 1. The same model was then fine-tuned for 4 epochs on a synthetic code-mixed MrEnCMSQuADQA dataset. Surprisingly, despite being a synthetic dataset, this fine-tuning led to superior outcomes with an EM score of 0.62, a BERTScore of 0.92, and an F1 score of 0.72. This improvement can be attributed to the larger scale of the code-mixed MrEnCMSQuADQA dataset. A comparison of the code-mixed question-answering system's performance on real and synthetic datasets is shown in Figure 4. Here the models are evaluated on EM score, BERT score and F1 score.

4. Marathi-English (Minglish) Code-Mixed Named Entity Recognition

Natural language processing (NLP) relies heavily on Named Entity Recognition (NER) since it is essential to the recognition and classification of named entities. These entities include people, places, things, and groups that are contained in a particular text. The usual method employed by a named entity recognizer involves a two-step process. Initially, it meticulously identifies named entities, treating them as individual words or clusters of adjacent words within sentences. Following this identification, the entities are then sorted into their predetermined classes. At the core of Named Entity Recognition (NER), the classification of text at an individual word level is of utmost importance. To illustrate this, let us examine the sentence, "Sachin Tendulkar is an international cricketer from India." In this sentence, "Sachin Tendulkar" is identified as a person, a clear example of a named entity. Similarly, "India" is also recognized as a country.

The increasing prevalence of code-mixing, or the use of multiple languages within a single utterance, in online communication underscores the need for a high-quality code-mixed Named Entity Recognition (NER) dataset. The availability of such a dataset is essential for developing a reliable NER model capable of accurately recognizing named entities in code-mixed text. By providing a diverse range of code-mixed text examples, the dataset would facilitate the model's learning process and enhance its ability to identify named entities in such contexts effectively.

4.1. Datasets for Code-Mixed NER

In the context of the Marathi language, a comprehensive Named Entity Recognition (NER) dataset has been meticulously crafted [19], comprising a substantial 25,000 manually tagged sentences. These sentences are presented in both IOB and non-IOB formats, allowing for diverse representation and analysis. The dataset encompasses a range of named entity types, catering to the specific nuances of the Marathi language. These types include Location (NEL), Time (NETI), Measure (NEM), Dates (NED), Designation (ED), and Organizations (NEO). This dataset is thoughtfully segmented into 21,500 training sentences with a total of 27,300 tagged words, 2,000 test sentences marked with 2,472 tagged words, and 1,500 validation sentences carrying 1,847 tagged words.

To pave the way for Marathi-English Code-Mixed Named Entity Recognition (NER) (MrEnCMNER) dataset, a parallel corpus is created. This involves the translation of the existing Marathi NER dataset into English. Subsequently, harnessing the potential of the proposed code-mixed text generation algorithm [14] a synthetic Marathi-English code-mixed NER dataset is carefully generated.

4.2. Proposed approach

For tasks like Named Entity Recognition (NER), pre-trained language models such as BERT and RoBERTa [20] are commonly fine-tuned. However, in scenarios where tasks require handling code-mixed languages but lack dedicated pre-trained code-mixed models, employing multilingual language models is a viable approach. Multilingual models like mBERT [21], Google Muril, and XLM-RoBERTa [22]can be effectively fine-tuned to leverage their

capabilities. Furthermore, when dealing with code-mixed text written in native or embedded language scripts, such as Devanagari for Marathi-English, utilizing regional language models during the fine-tuning phase can provide significant advantages.

The Code-Mixed Marathi-English (Minglish) Named Entity Recognition (NER) model is constructed using a transfer learning method, which consists of two primary phases. The first phase involves pre-training an extensive neural network in an unsupervised mode. The second phase involves fine-tuning this neural network for the task at hand. For BERT (Bidirectional Encoder Representations from Transformers) and its variants, the fundamental architecture is adapted by incorporating a token classification head onto the uppermost layer. This modification empowers the model to produce predictions for individual tokens instead of the entire sequence. Within this framework, NER is predominantly treated as a token classification task. Figure 5 represents basic process of fine-tuning MahaRoBERTa [23] on NER task.

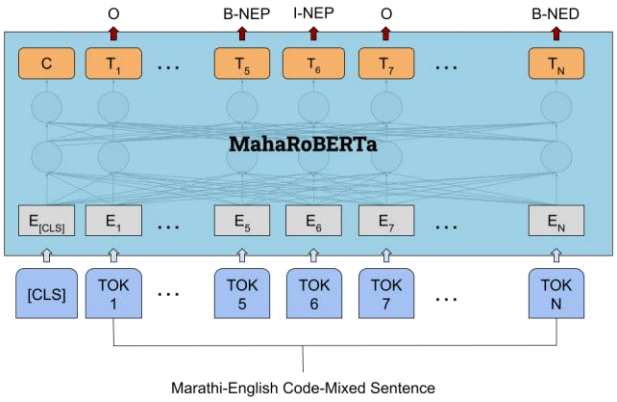

Fig 5. Fine-tuning MahaRoBERTa transformer for Named Entity Recognition

To perform comparative analysis, the fine-tuning process was executed on various models, including mBERT, Google MuRIL [17], XLM-RoBERTa , MahaBERT [23], and MahaRoBERTa [23] , with the aim of developing code-mixed Named Entity Recognition (NER) models. Fine-tuning process is conducted on the MrEnCMNER dataset.

4.3. Results and Analysis

Sequence labeling is a type of classification in which specific items in a series of data, like words or tokens in a sentence, are given labels. The objective is to assign a specific label to each element in the sequence based on its characteristics or traits. Named Entity Recognition (NER) is a type of sequence labelling categorization that places labels, such as names, dates, and locations, on specific items within a text sequence. It is crucial to consider how well a named entity recognition (NER) model identifies entities in order to thoroughly assess its performance.

Precision: It is the ratio of correctly identified entities to the total anticipated entities. Precision helps evaluate the model's false positive rate for each entity type. For example, if a model identifies "person" entities, precision indicates the percentage of "person" entities that were correctly identified out of all the entities that were predicted to be "person".

Recall: Recall measures a model's ability to identify named entities in a text. It indicates the effectiveness of the model in finding existing entities without missing any. Recall is useful for evaluating the rate of false negatives. For example, if the focus is on location entities, recall would reveal the ratio of correctly identified location entities to all actual location entities. This metric helps assess the false negative rate.

F1-Score: The F1 score combines recall and precision, providing a fair evaluation of a model's ability to identify named entities accurately. It comprehensively assesses a model's recognition of named entities, considering both false positives and false negatives.

All the fine-tuned models are evaluated based on key metrics, including F1 score, precision, and recall. These combined metrics provide a comprehensive understanding of how each model excels in accurately identifying named entities and mitigating errors in a code-mixed context.

As mentioned in Table 2 the mBERT model achieves a balanced F1 score, indicating an overall good trade-off between precision and recall. This implies that while it manages to identify named entities correctly, it maintains a fair balance between avoiding false positives (high precision) and capturing a significant proportion of actual entities (high recall).

Table 2. Comparison of different transformer models fine-tuned over code-mixed MrEnCMNER dataset

Model	***F1***	***Precision***	***Recall***
mBERT	71.66	73.62	70.12
MuRIL	70.01	80.00	62.24
XLM-RoBERTa	73.78	74.50	73.07
MahaBERT	69.72	79.48	62.09
MahaRoBERTa	73.92	75.27	72.63

MuRIL exhibits a high precision score of 80.00, which implies that it's cautious in labeling named entities, leading to fewer false positives. However, this high precision comes at the expense of recall, as it captures only 62.24% of the actual entities. XLM-RoBERTa exhibits a good mix between recall and precision. Indicating a well-rounded performance, Its high recall (73.07%) suggests it's adept at

identifying a significant portion of actual named entities, while its precision (74.50%) ensures the precision-recall trade-off remains well-maintained. MahaBERT displays a higher precision score (79.48%), indicating fewer false positives. However, like MuRIL, this precision comes with a compromise on recall (62.09%), which means it might miss a portion of actual named entities. MahaRoBERTa emerged as the top performer with an impressive F1 score of 73.92. Its precision and recall scores are well-balanced, indicating that it excels in both labeling named entities accurately (precision) and capturing a substantial portion of actual named entities (recall).

MahaRoBERTa emerged as the frontrunner, boasting a remarkable F1 score of 73.92. The equilibrium between its precision and recall scores underscores its adeptness in accurately labeling named entities (precision) while comprehensively capturing genuine named entities (recall). This achievement accentuates its exceptional competency. This establishes it as the most adept model for code-mixed Named Entity Recognition on the MrEnCMNER dataset. The RoBERTa architecture's well-established prowess in classification tasks further solidifies its commendable performance in the realm of NER. Additionally, the multilingual model MuRIL achieves the highest precision score of 80.00, indicating a minimal occurrence of false positives. On the other hand, XLM-RoBERTa demonstrates a robust recall value of 73.07, indicating its proficiency in identifying a significant portion of actual entities. Figure 6 complements these findings, offering a visual comparison of diverse transformer models fine-tuned using the codemix MrEnCMNER dataset.

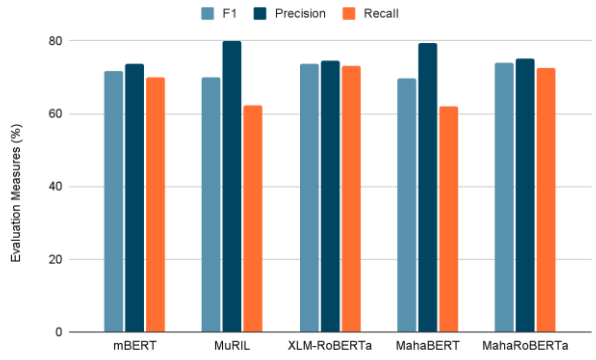

Fig 6. Comparison of different multilingual and monolingual models for code-mixed Named Entity Recognition

5. Conclusion

The research demonstrates the effectiveness of Question Answering (QA) and Named Entity Recognition (NER) systems in the context of Marathi-English code-mixed language. The research has innovatively introduced a pioneering dataset for Marathi-English code-mixed QA and NER, marking a noteworthy milestone in the field. The proposed QA system, leveraging MuRIL, demonstrates promising results with an Exact Match (EM) score of 0.41 and 0.62 on real and synthetic datasets, respectively. Additionally, the code-mixed NER model, based on MahaRoBERTa, outperforms counterparts with an impressive F1 score of 73.92, showcasing its proficiency in accurately labeling named entities in code-mixed text. In future work, there is an intent to develop and train language models, particularly large language models, utilizing code-mixed datasets to achieve superior performance in multilingual contexts.

References

[1] S. Singh, M. Anand Kumar, and K. P. Soman, “CEN@Amrita: Information retrieval on CodeMixed Hindi English tweets using vector space models,” in CEUR Workshop Proceedings, 2016.

[2] D. S. Sharma et al., “Improving Document Ranking using Query Expansion and Classification Techniques for Mixed Script Information Retrieval,” 2016.

[3] K. R. Chandu, M. Chinnakotla, A. W. Black, and M. Shrivastava, “WebShodh: A code mixed factoid question answering system for web,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2017. doi: 10.1007/978-3-319-65813-1_9.

[4] D. Gupta, P. Lenka, A. Ekbal, and P. Bhattacharyya, “Uncovering code-mixed challenges: A framework for linguistically driven question generation and neural based question answering,” in CoNLL 2018 - 22nd Conference on Computational Natural Language Learning, Proceedings, 2018. doi: 10.18653/v1/k18-1012.

[5] D. Gupta, A. Ekbal, and P. Bhattacharyya, “A deep neural network framework for English Hindi question answering,” ACM Transactions on Asian and Low-Resource Language Information Processing, vol. 19, no. 2, 2019, doi: 10.1145/3359988.

[6] S. Thara, E. Sampath, B. Venkata Sitarami Reddy, M. Vidhya Sai Bhagavan, and M. Phanindra Reddy, “Code mixed question answering Challenge using deep learning methods,” in Proceedings of the 5th International Conference on Communication and Electronics Systems, ICCES 2020, 2020. doi: 10.1109/ICCES48766.2020.09137971.

[7] S. Dowlagar and R. Mamidi, “CMNEROne at SemEval-2022 Task 11: Code-Mixed Named Entity Recognition by leveraging multilingual data,” in SemEval 2022 - 16th International Workshop on Semantic Evaluation, Proceedings of the Workshop,

2022. doi: 10.18653/v1/2022.semeval-1.214.

[8] K. Singh, I. Sen, and P. Kumaraguru, "Language identification and named entity recognition in hinglish code mixed tweets," in ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Student Research Workshop, 2018. doi: 10.18653/v1/p18-3008.

[9] A. El Mekki, A. El Mahdaouy, M. Akallouch, I. Berrada, and A. Khoumsi, "UM6P-CS at SemEval-2022 Task 11: Enhancing Multilingual and Code-Mixed Complex Named Entity Recognition via Pseudo Labels using Multilingual Transformer," in SemEval 2022 - 16th International Workshop on Semantic Evaluation, Proceedings of the Workshop, 2022. doi: 10.18653/v1/2022.semeval-1.207.

[10] V. K. Srirangam, A. A. Reddy, V. Singh, and M. Shrivastava, "Corpus Creation and Analysis for Named Entity Recognition in Telugu-English Code-Mixed Social Media Data," in Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop, F. Alva-Manchego, E. Choi, and D. Khashabi, Eds., Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 183–189. doi: 10.18653/v1/P19-2025.

[11] C. Sabty, A. Sherif, M. Elmahdy, and S. Abdennadher, "Techniques for Named Entity Recognition on Arabic-English Code-Mixed Data," International Journal of Robotic Computing, 2019, doi: 10.35708/tai1868-126245.

[12] R. Priyadharshini, B. R. Chakravarthi, M. Vegupatti, and J. P. McCrae, "Named Entity Recognition for Code-Mixed Indian Corpus using Meta Embedding," in 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), 2020, pp. 68–72. doi: 10.1109/ICACCS48705.2020.9074379.

[13] Y. Madhani et al., "Aksharantar: Towards building open transliteration tools for the next billion users," ArXiv, vol. abs/2205.03018, 2022.

[14] D. Amin et al., "Marathi-English Code-mixed Text Generation," ArXiv, vol. abs/2309.16202, 2023.

[15] D. Amin, S. Govilkar, and S. Kulkarni, "Question answering using deep learning in low resource Indian language Marathi," ArXiv, vol. abs/2309.15779, 2023.

[16] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional Transformers for language understanding," Oct. 2018.

[17] S. Khanuja et al., "MuRIL: Multilingual representations for Indian languages," Mar. 2021.

[18] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "BERTScore: Evaluating Text Generation with BERT," Apr. 2019.

[19] P. Patil, A. Ranade, M. Sabane, O. Litake, and R. Joshi, "L3Cube-MahaNER: A Marathi Named Entity Recognition Dataset and BERT models," ArXiv, vol. abs/2204.06029, 2022.

[20] Y. Liu et al., "RoBERTa: A robustly optimized BERT pretraining approach," Jul. 2019.

[21] T. Pires, E. Schlinger, and D. Garrette, "How Multilingual is Multilingual BERT?," in Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, A. Korhonen, D. Traum, and L. Màrquez, Eds., Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 4996–5001. doi: 10.18653/v1/P19-1493.

[22] A. Conneau et al., "Unsupervised cross-lingual representation learning at scale," Nov. 2019.

[23] R. Joshi, "L3Cube-MahaCorpus and MahaBERT: Marathi monolingual corpus, Marathi BERT language models, and resources," Feb. 2022.