CMLFormer: A Dual Decoder Transformer with Switching Point Learning for Code-Mixed Language Modeling

NEW YORK UNIVERSITY

Aditeya Baral† Allen George Ajith† Roshan Nayak§ Mrityunjay Abhijeet Bhanja§

†Courant Institute of Mathematical Sciences, §Tandon School of Engineering

Motivation

Code-mixed languages contain frequent and unstructured mid-sentence language switches; disrupts grammatical and semantic structure

Existing multilingual models are not natural code-mixers; fail at representing the inherent characteristics of code-mixing

Standard pre-training objectives lack supervision for language transitions and often overlook structural dynamics critical to CM understanding

Effective modeling of CM text requires **linguistically-aware designs** and **targeted objectives** focused on switching behavior and multilingual structure

Pre-training CMLFormer$_{base}$

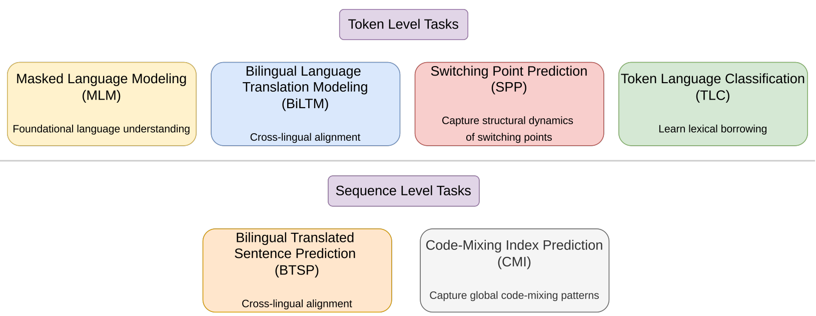

Figure 1. **Overview of CMLFormer's pre-training objectives**. Token-level tasks (top) capture local language dynamics such as foundational semantics, cross-lingual token alignment, language identity, and transition boundaries. Sequence-level tasks (bottom) model broader code-mixing phenomena, including sentence-level equivalence and global language-mixing complexity.

Setup

- Pre-trained on augmented L3Cube-HingCorpus
- Custom WordPiece tokenizer with shared vocabulary across code-mixed, base and mixing languages
- CMLFormer's encoder size matched with BERT$_{base}$ for fair comparison
- Multi-task optimization through joint pre-training

Fine-tuning CMLFormer$_{base}$

- Evaluated on HASOC 2021 (code-mixed hate-speech detection)
- Benchmarked against HingBERT (BERT$_{base}$ pre-trained with MLM)
- Full fine-tuning of CMLFormer encoder with classification head

Overview

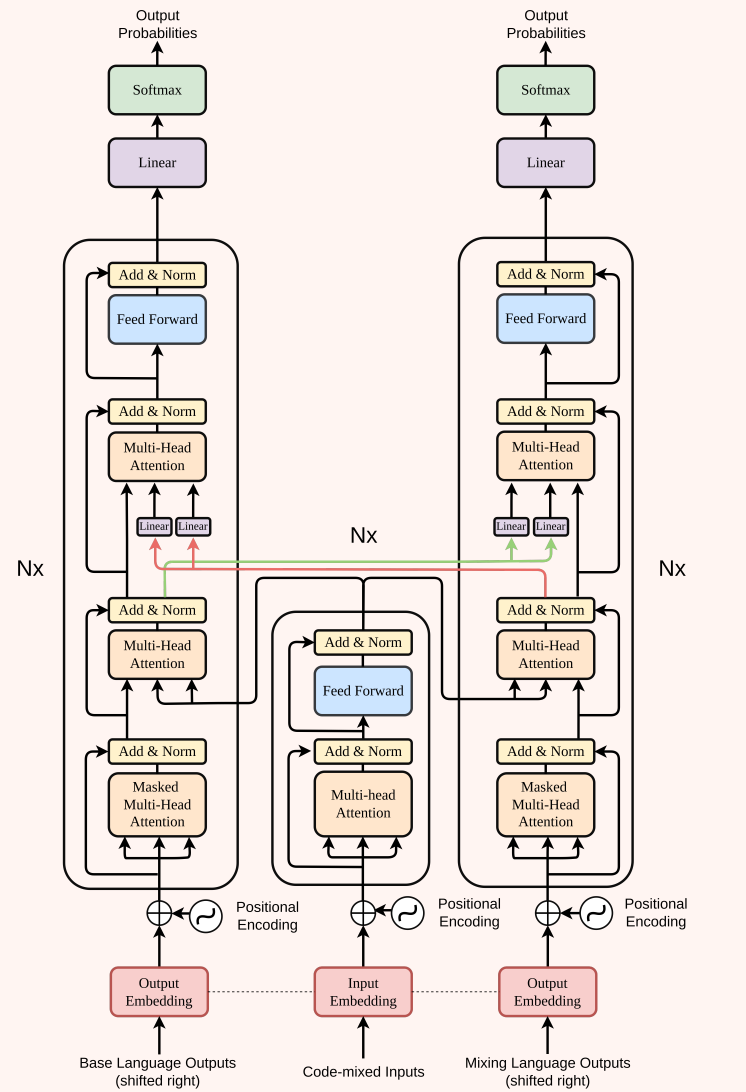

Figure 2. **The architecture of our proposed approach CMLFormer**: The outputs from each encoder-decoder attention sub-layer (arrows shown in green and red) are passed as input to the decoder-decoder cross-attention sub-layer. The decoders exhibit **full synchronous coupling** since each requires the hidden states from the other to compute its own hidden states. After pre-training, the encoder is detached and fine-tuned on downstream tasks.

Fine-tuning Results

Model	MLM	BiLTM	SPP	BTSP	TLC	CMI	Precision	Recall	Accuracy	F1
BERT$_{base}$	✓						0.189	0.367	0.496	0.249
CMLFormer$_{base}$	✓	✓					**0.327**	**0.633**	**0.504**	**0.431**
	✓	✓	✓				0.223	0.433	0.498	0.295
	✓	✓	✓	✓			0.086	0.167	0.490	0.113
	✓	✓	✓	✓	✓		0.120	0.233	0.492	0.159
	✓	✓	✓	✓	✓	✓	0.155	0.300	0.494	0.204

Table 1. **Results on HASOC 2021 with different pre-training objectives.** CMLFormer outperforms BERT$_{base}$ across all metrics when BiLTM and SPP pre-training strategies are applied. A ✓indicates the pre-training strategy applied, green indicates a gain in performance and **bold** indicates the best performance on that metric.

Learning Switching Point Dynamics

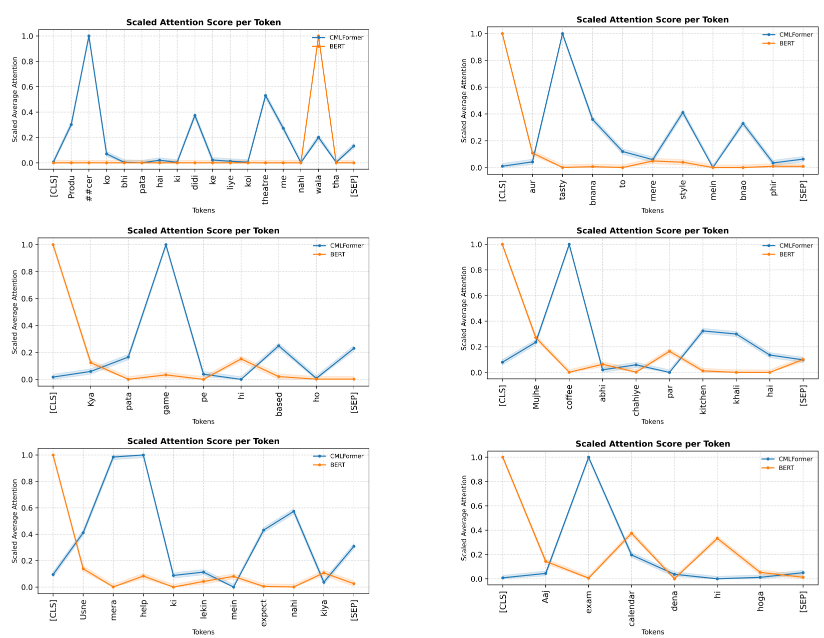

Figure 3. **Average Attention Score per Token**. CMLFormer consistently identifies and attends to language transitions around switching points, and is agnostic to the nature and number of transitions; BERT$_{base}$ fails to identify transitions and attends to trivial tokens.