Reformulating MaskLID with ILP for Robust Code-Switching Detection

PLDAC

Dwi Prima Handayani Putri 21402854

Master 1 DAC
Academic Year: 2024–2025

Date: May 21st, 2025

A report submitted in partial fulfillment of the PLDAC subject at Sorbonne University

Abstract

Code-switching language identification (CS-LID) is a challenging task due to the limited availability of annotated data and the highly variable structure of multilingual sentences. MaskLID is a heuristic-based method that performs language segmentation by iteratively masking tokens based on confidence scores from a pretrained language identification model. However, its greedy nature and lack of global inference often result in suboptimal predictions.

In this work propose an optimization-based reformulation of MaskLID by modeling the segmentation task as an Integer Linear Programming (ILP) problem. The objective is to maximize the sum of logit scores under constraints such as the number of active languages and minimum segment length per language. The approach supports multiple language identifiers including GlotLID and OpenLID.

Experiments conducted on the Turkish–English code-switching dataset demonstrate that the optimization framework significantly improves Exact Match (EM) performance over the baseline MaskLID, while maintaining competitive Partial Match (PM) accuracy. These results highlight the benefits of integrating global optimization strategies for more consistent and linguistically meaningful code-switching detection.

Contents

1 Introduction

1.1 Background

Reference to a paper on CS, or even to the Wikipedia page

Code-switching is a common phenomenon in multilingual communication. Code-switching refers to the alternation between two or more languages within a single linguistic interaction. This switch can occur within a single word or within a sentence. Intra-word code-switching happens when a suffix from one language is added to the lexical base of another language, for example: “ce projet, j'ai liké de ouf.” An example of inter-word code-switching is: “That's comme ça that you thank me to have learned you english.” This phenomenon is interesting and also important in various fields such as natural language processing (NLP), online communication, informal dialogue, and social media contexts.

Reference to the paper

The only requirement is that the model outputs a logit vector for each output token, whose component (one per language) are proportional to the posterior probability of that language, given the token P(L|W).

MaskLID is a model used to detect code-switching. MaskLID leverages the predictions from language identification models such as FastText, OpenLID, or GlotLID as the main input for its masking strategy. It is designed to improve the performance of these LID models, which typically only return the dominant language label for a code-switched sentence. MaskLID applies a simple approach by masking segments ~~of~~ the dominant language in a sentence, allowing the identification of other language segments in subsequent iterations. The model works by computing the dominant language probability using softmax and masking word segments with high logit values for that language. This process is repeated iteratively in the following steps.

iteratively masking

Reference to the papers

1.2 Problem

Code-switching detection using MaskLID has several limitations. First, MaskLID lacks an explicit objective function as the basis for its inference process. The masking process is performed heuristically without involving an optimization mechanism to find the globally best solution. This can lead to inconsistent predictions, especially when words are linguistically ambiguous. Second, the classification decision for each token or word is made independently. MaskLID does not take into account the preceding or following words. As a result, language predictions within the same language segment can switch unnaturally. Third, there is no explicit constraint on the number of languages allowed within a sentence segment. In ambiguous cases, the model may produce unrealistic multilingual outputs. Fourth, MaskLID's segmentation does not guarantee language continuity. A language may appear in separate, fragmented pieces that are linguistically inconsistent. However, more contiguous segmentation is crucial in code-switching to maintain coherent language representation.

1.3 Proposed Approach

This project focuses on an approach to reformulate the inference process of MaskLID as an optimization problem. Two LID models are used: OpenLID and GlotLID. For faire comparison, the languages in GlotLID is limited to the same set as OpenLID. The approaches covered in this project include:

fair

- First, redefining the objective function based on the sum of language prediction logits for each token, which is globally maximized.
- Second, adding a constraint to limit the maximum number of languages that can appear in a single predicted sentence.
- Third, selecting the best label for each token by considering the objective function and the defined constraints through an Integer Linear Programming (ILP) formulation.
- Forth, this approach is tested on a mixed `Turkish-English` dataset as an initial validation stage, compared to the original MaskLID segmentation.

2 Data Set

The dataset used in this work is the Turkish-English dataset developed by Yirmibeşoğlu and Eryi as part of their work on CS LID for language pairs. The original dataset can be found at

You should also consider monolingual data and mix all test sets
This will be important to detect the false positive (non code-switched)
that are code-switched.

github.com/zeynepyirmibes/code-switching-tr-en. It is available in a token-level annotation format with labels of either Turkish or English assigned to each token. In this project, only the Turkish-English code-switched data is extracted, and sentences containing only mono-English or mono-Turkish are removed. The labels for each sentence are also converted into the appropriate language label format used by the LID models, namely `__label__tur_Latn` and `__label__eng_Latn`. This is necessary for calculating the number of exact matches and partial matches between the model's output and the original labels. The token-level dataset is already clean, so no additional preprocessing is required. The only step taken is filtering sentences, where only those longer than 40 bytes are used. The final dataset consists of 339 sentences.

Give one examples of the data set.

3 Problem Formulation and ILP Modeling

Integer Linear Programming (ILP) is used to reformulate the language segmentation task, which was originally performed heuristically by MaskLID. The goal is to select the best language label for each token in a sentence by taking into account the confidence scores from the language identifier, as well as a set of reasonable linguistic constraints.

3.1 Notation

Let:

explain

- W be the set of tokens in a given sentence,
- $L \subseteq \mathcal{L}$ be the set of candidate language labels, obtained from the top-α predictions across all tokens,
- b_{ij} be the logit score assigned by the model for token $i \in W$ and label $j \in L$,
- $x_{ij} \in \{0, 1\}$ be a binary decision variable: $x_{ij} = 1$ if token i is assigned label j, and 0 otherwise,
- $y_j \in \{0, 1\}$ be a binary variable indicating whether label j is used in the final prediction.

3.2 Objective Function

The objective of the optimization is to assign a language label to each token in such a way that the total logit scores across all assignments are maximized. These logit scores, b_{ij}, represent the confidence of the pretrained language identification model in assigning language j to token i. By maximizing their sum, the model selects the most confident overall segmentation under the given constraints.

The objective function is formulated as:

$$\max \sum_{i \in W} \sum_{j \in L} b_{ij} \cdot x_{ij}$$

where x_{ij} is a binary decision variable indicating whether token i is assigned language label j.

3.3 Constraints

The model is subject to the following constraints:

- **Unique label per token:**

  language label

  $$\sum_{j \in L} x_{ij} = 1 \quad \forall i \in W$$

- **Language activation constraint:**

  $$\sum_{i \in W} x_{ij} \leq |W| \cdot y_j \quad \forall j \in L$$

  This ensures that $y_j = 1$ only if label j is assigned to at least one token.

- **Maximum number of active languages:**

$$\sum_{j \in L} y_j \leq \alpha$$

Find another notation; \alpha is something else

- **Minimum byte-length per language:**

Find another notation

$$\sum_{i \in W} \mathrm{len}(i) \cdot x_{ij} \geq \mathrm{min\_len} \cdot y_j \quad \forall j \in L$$

This constraint filters out languages that only cover very short segments.

The ILP problem is solved using the `Gurobi Optimizer`. After optimization, the final language assignment is determined by selecting, for each token i, the label j such that $x_{ij} = 1$.

In fact we are only interested in the list of languages for each y_j = 1.

4 Evaluation Metrics

For the evaluation of the optimization results, two primary metrics are used: **Exact Match (EM)** and **Partial Match (PM)**.

4.1 Exact Match (EM)

Exact Match (EM) measures the percentage of sentences for which the predicted set of language labels exactly matches the ground truth labels. In other words, a prediction is counted as an EM only if all tokens are correctly labeled, resulting in an exact match of the label set at the sentence level.

$$\mathrm{EM} = \frac{\text{\# of sentences with all labels correctly predicted}}{\text{Total number of sentences}} \times 100$$

No: EM i= 1 when we system predicts exactly the languages observed in the sentence, no less, no more.

For this data set, we could also compute at the word level, but this would be a different metric.

4.2 Partial Match (PM)

Partial Match (PM) is used when a prediction does not achieve an exact match but still shares at least one language label with the ground truth. This means that some part of the code-switching was correctly identified, even though not all labels matched.

$$\mathrm{PM} = \frac{\text{\# of sentences with partial label overlap}}{\text{Total number of sentences}} \times 100$$

It would be clearer to have the EM also count for a PM. Also the FP rate (over identification of CS)/

Note that in this evaluation, EM and PM are defined at the sentence level and are mutually exclusive. A sentence is either counted as an Exact Match (if all labels match), a Partial Match (if some but not all labels match), or excluded from both (if there is no overlap with the gold labels). This evaluation provides a coarse-grained view of code-switching detection performance.

4.3 Implementation and Result

what is the value of top-alpha ? what is the value of min len

Both EM and PM are computed by comparing the predicted labels from the model (either baseline MaskLID or the proposed optimization method) with the ground truth labels extracted from the Turkish-English annotated corpus. In all experiments, we set the value of α, the maximum number of active languages per sentence, to 2. This reflects the assumption that each sentence in the dataset contains at most two languages, which is consistent with the nature of code-switching corpora such as Turkish–English. This constraint helps prevent over-segmentation and spurious language assignments that could arise if the model is allowed to assign arbitrary language labels. The metrics are reported as percentages.

Model	**Exact Match (EM) [%]**		**Partial Match (PM) [%]**	
	OpenLID	GlotLID	OpenLID	GlotLID
MaskLID (heuristic)	17.1%	21.8%	81.1%	76.7%
Optimization (ILP)	**39.82%**	**37.46%**	**57.23%**	**60.47%**

Table 1: Comparison of baseline MaskLID and optimization-based inference using OpenLID and GlotLID as the underlying language identifier.

This is not what we reported in the paper - about 27 for glotLID and 20 for openLID. What is the difference ?

4.4 Analysis

The evaluation results in Table 1 show a clear improvement in both Exact Match (EM) and Partial Match (PM) when using the proposed optimization-based inference compared to the original MaskLID heuristic.

For both GlotLID and OpenLID, the optimization approach improves EM significantly. In particular, OpenLID benefits the most, with EM increasing from 17.1% to 39.82%. This suggests that the optimization model is better at globally assigning consistent labels that fully match the gold annotation.

Interestingly, the Partial Match (PM) scores decrease in the optimized version. This is expected, since EM and PM are mutually exclusive in this evaluation. A higher EM typically implies fewer borderline cases that fall into the PM category. Despite the PM drop, the overall increase in fully correct predictions (EM) indicates a more reliable model behavior.

Comparing GlotLID and OpenLID, GlotLID yields slightly lower PM scores in the heuristic baseline but outperforms OpenLID in the optimized version in terms of PM. However, OpenLID slightly outperforms GlotLID in terms of EM, indicating a minor advantage in exact sentence-level labeling.

Can you add examples of cases where the heuristic approach fails, and the ILP methof works ?

Can you give statistics for the cases where the dominant language takes too many words
so that the remaining parts are too short ?

4.5 Limitations

Future work should include testing on other code-switched datasets from the LinCE benchmark, such as Nepali–English, Hindi–English, and similar language pairs. On the other hand, although the ILP-based approach significantly improves sentence-level accuracy, it does not yet enforce contiguity of language segments. Adding such structural constraints is a promising direction for future work.

4.6 Conslusion

Overall, these results confirm that formulating MaskLID as an ILP problem improves the quality of language segmentation in code-switched sentences. One notable advantage of this approach is that it does not require the user to specify the set of languages in advance. This makes the method more flexible and applicable to real-world multilingual data, where the specific language pair being mixed is not always known beforehand.