Final Report: **Language Detection (Kyrgyz vs Other Languages)**

### **1. Project Objective**

**The goal of this project was to create a text classification model capable of identifying whether a given phrase is written in Kyrgyz or belongs to any other language.  
 We performed fine-tuning on the Whisper Small Kyrgyz model from Hugging Face, using a custom multilingual dataset containing simple phrases in Kyrgyz, Russian, and English.**

### **2. Dataset Description**

* **Total samples: 40,000**
* **Languages:**
  + **Kyrgyz 20000**
  + **Russian 10000**
  + **English 10000**

**Structure:**

|  |  |
| --- | --- |
| **Text** | **Label** |
| **"Салам"** | **1** |
| **"Hello, how are you?"** | **0** |
| **"Как дела?"** | **0** |

**Label Explanation:**

* **1 → Kyrgyz**
* **0 → Other languages**

**We have trained the model for 20 minutes with 1 epoch because when we tried to make more epochs it gave the retrained model.**

### **4. Training Results**

**After fine-tuning, the model achieved perfect classification results on validation data:**

|  |  |
| --- | --- |
| **Metric** | **Value** |
| **Training Loss** | **0.0032** |
| **Validation Loss** | **0.0000** |
| **Accuracy** | **1.0000** |
| **F1 Score** | **1.0000** |

**Note: Since the dataset contained very simple and distinct phrases, the model quickly achieved 100% accuracy on training and validation sets.**

### **Obtaining a model**

### **3.1 Fine-Tuning and Model Adaptation**

### **3.1.1 Environment & Tools**

### **Platform: Google Colab with a T4 GPU (12 GB VRAM), Python 3.11**

### **Key libraries:**

### **transformers (Hugging Face) for XLM-Roberta**

### **datasets & pandas for data preparation**

### **scikit-learn for train/validation split and metrics**

### **torch (PyTorch) for training**

### **gradio for interactive model demo**

### **3.1.2 Dataset Preparation & Balancing**

### **Selected 4 000 examples for each language from the multilingual CSV:**

### **Kyrgyz → 4 000**

### **Russian → 4 000**

### **English → 4 000**

### **Merged Russian & English into a single Other class; kept Kyrgyz as its own class.**

### **Performed a stratified 85 / 15 train/validation split on the combined dataset.**

### **3.1.3 Model Architecture & Hyperparameters**

### **Base model: xlm-roberta-base**

### **Task: Binary classification (num\_labels=2)**

### **Training settings:**

### **Epochs: 2**

### **Batch size per device: 8**

### **Learning rate: 2 × 10⁻⁵**

### **Mixed-precision (fp16=True) to save memory**

### **Save checkpoint every 500 steps**

### **3.1.4 Fine-Tuning Workflow**

### **Loaded xlm-roberta-base and its tokenizer from Hugging Face.**

### **Tokenized all examples in batches (max length = 128) with the datasets.map() API.**

### **Wrapped tokenized inputs in a custom PyTorch Dataset that returns (input\_ids, attention\_mask, labels).**

### **Initialized a Hugging Face Trainer with metric-based checkpointing (best F1-macro).**

### **Used trainer.train(resume\_from\_checkpoint=True) to avoid losing progress on session restarts.**

### **3.1.5 Results of My Fine-Tuning**

### **Training time: ≈ 3 minutes on T4 GPU**

### **Final metrics on validation set:**

### **Training Loss: ≈ 0.19**

### **Validation Loss: ≈ 1.7 × 10⁻⁵**

### **Validation Accuracy: 1.00**

### **Validation F1-macro: 1.00**

### **Checkpoint saved in KGvsOther-finetuned/ (model weights + tokenizer files).**

### **3.1.6 Interactive Demo**

### **Built a simple Gradio interface that lets users paste any text and instantly see:**

### **Probability scores for Kyrgyz vs Other**

### **Final predicted label and confidence**

### **The live demo link is generated automatically from Colab (share=True), enabling quick validation of the model in action.**