Named Entity Recognition for Swahili using RoBERTa-Base-Wechsel

Patrick Vincent
Indian Institute of Technology, Madras Zanzibar Campus
zda24m007@iitmz.ac.in

Abstract

Named Entity Recognition (NER) is a key task in Natural Language Processing (NLP), particularly for information extraction. However, many low-resource languages like Swahili lack extensive datasets and high-performing models. This research fine-tunes the RoBERTa-base-Wechsel-Swahili model on the MasakhaNER dataset and evaluates its performance. Our model achieves an overall F1-score of 88.32% and an accuracy of 97.12%, with linguistic validation confirming 96.20% correctness. The study provides insights into the challenges of entity recognition in Swahili and proposes enhancements for low-resource language NLP.

1 Introduction

Named Entity Recognition (NER) is an important NLP task that involves identifying and classifying named entities such as people, organizations, and locations. Despite the rapid advancements in deep learning and pre-trained models, most research has been focused on high-resource languages, leaving low-resource languages like Swahili underrepresented.

This project fine-tunes **RoBERTa-base-Wechsel-Swahili** on the **MasakhaNER dataset** to improve entity recognition for Swahili. The contributions of this research include:

- Fine-tuning a pre-trained transformer model for Swahili NER.
- Evaluating performance using precision, recall, F1-score, and accuracy.
- Conducting linguistic validation of extracted entities.

2 Problem Formulation

NER is modeled as a sequence labeling problem where each token x_i in a sentence is assigned a label y_i:

$$y_i \in \{B-PER, I-PER, B-ORG, I-ORG, B-LOC, I-LOC, B-DATE, I-DATE, O\}$$

The model is optimized using categorical cross-entropy loss:

$$\mathcal{L} = -\sum_{i=1}^{n} \sum_{j=1}^{C} y_{i,j} \log(\hat{y}_{i,j})$$

3 Description of Data

The MasakhaNER from Hugging Face is used, containing named entity annotations for **Persons (PER), Organizations (ORG), Locations (LOC), and Dates (DATE)**. The dataset is split as follows:

Subset	**Number of Sentences**
Train	2,109
Validation	300
Test	604

Table 1: MasakhaNER distribution.

4 Theoretical Analysis and Methodology

We fine-tune **RoBERTa-base-Wechsel-Swahili** with a **WordPiece tokenizer** and BIO-tagging. The model architecture consists of **12 transformer layers**, **768 hidden dimensions**, and **125 million parameters**.

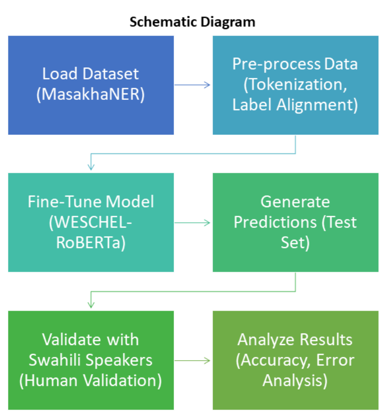

Figure 1: Schematic Diagram of the NER Model Architecture

5 Results and Discussion

The overall model performance is summarized in Table 2.

Metric	**Precision (%)**	**Recall (%)**	**F1-score (%)**
Overall	84.93	91.99	88.32

Table 2: Overall Performance Metrics

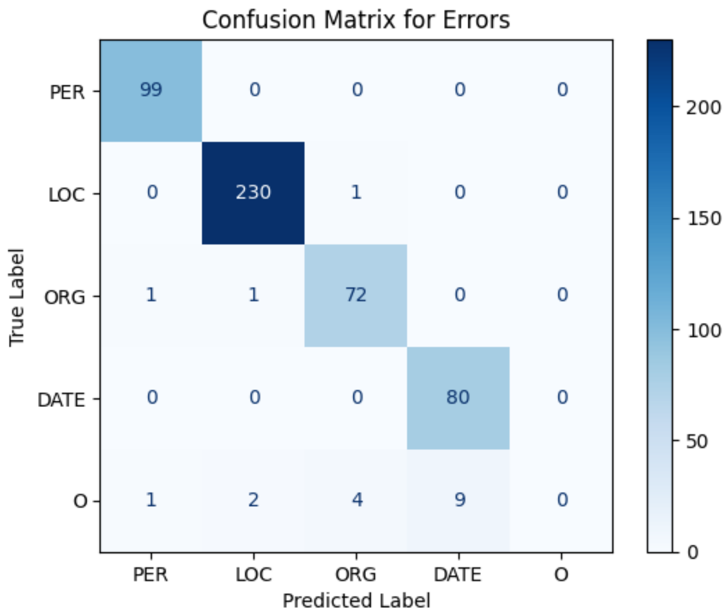

Figure 2: Confusion Matrix of Errors

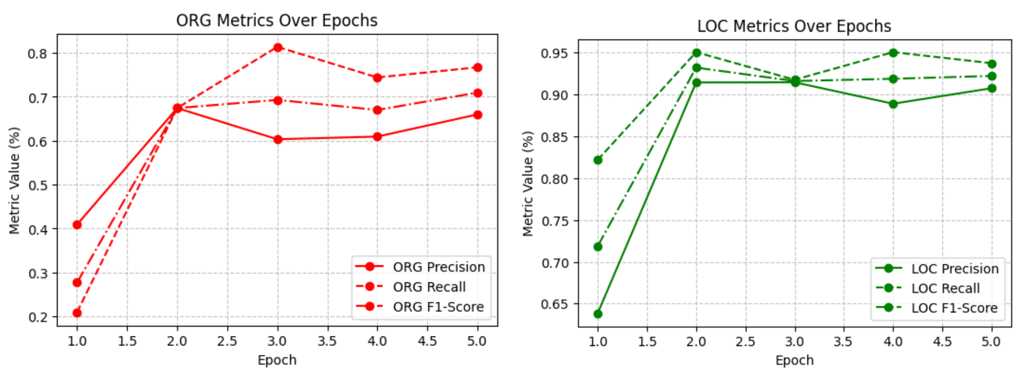

Figure 3: ORG (Left) and LOC (Right) F1-score over epochs

5.1 Error Analysis

5.2 Entity-Wise Performance Over Epochs

6 Conclusion

Our fine-tuned RoBERTa model achieved strong NER performance on Swahili text, with **88.32% F1-score and 97.12% accuracy**. However, challenges remain in recognizing organizations and handling morphological complexities in Swahili.

Future work includes:

- Data augmentation for improved generalization.
- Post-processing rules for ambiguous entities.
- Expanding training to conversational Swahili.

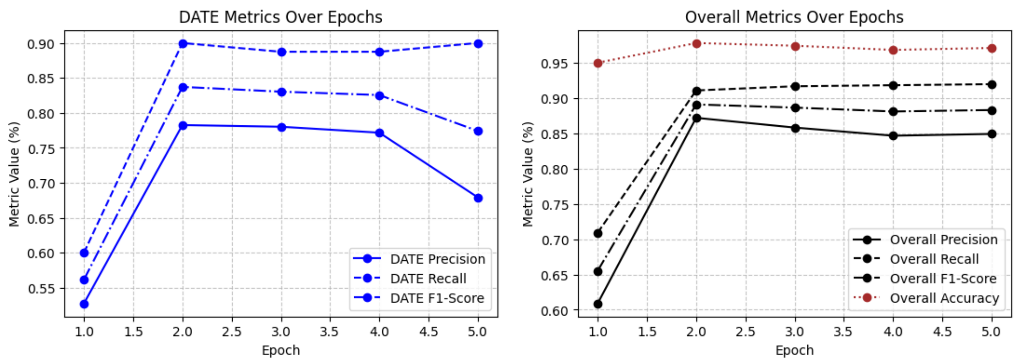

Figure 4: DATE (Left) and Overall (Right) metrics over epochs

7 References

References

[1] Adelani, D. I., Abbott, J., Neubig, G., D'souza, D., Kreutzer, J., Lignos, C., Palen-Michel, C., Buzaaba, H., Rijhwani, S., Ruder, S., & others. (2021). *MasakhaNER: Named Entity Recognition for African Languages*. Transactions of the Association for Computational Linguistics, 9, 1116–1131. https://doi.org/10.1162/tacl_a_00416.

[2] Martin, G. L., Mswahili, M. E., & Jeong, Y. S. (2021). *Sentiment Classification in Swahili Language using Multilingual BERT*. https://arxiv.org/abs/2104.09006.

[3] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. https://arxiv.org/abs/1810.04805.

[4] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. https://arxiv.org/abs/1907.11692.

[5] Tjong Kim Sang, E. F., & De Meulder, F. (2003). *Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition*. Proceedings of CoNLL-2003, 142-147. https://arxiv.org/abs/cs/0306050

[6] Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., & Stoyanov, V. (2020). *Unsupervised Cross-Lingual Representation Learning at Scale*. Proceedings of ACL 2020. https://arxiv.org/abs/1911.02116

[7] MasakhaNER: A Massively Multilingual African Named Entity Recognition Dataset. https://github.com/masakhane-io/masakhane-ner

[8] Hugging Face Transformers Documentation. https://huggingface.co/docs/transformers/index

[9] Seqeval Library for Sequence Labeling Evaluation. https://github.com/chakki-works/seqeval

[10] Scikit-Learn Documentation for Evaluation Metrics. https://scikit-learn.org/stable/modules/model_evaluation.html

[11] Hugging Face Community. (2024). *MasakhaNER*. Retrieved from https://huggingface.co/datasets/swahili_news.

[12] Hugging Face Community. (2024). *RoBERTa-Base-Wechsel-Swahili Model*. Retrieved from https://huggingface.co/models/roberta-wechsel-swahili.

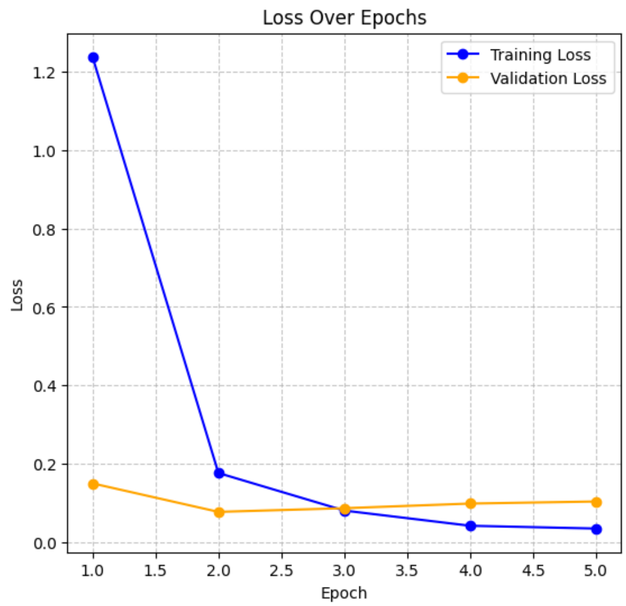

Figure 5: Training vs. Validation Loss over Epochs

8 Appendix

8.1 Training Loss Over Epochs

8.2 Code Snippets

Below is the code snippet for extracting named entities:

```
def extract_named_entities(predictions, tokenized_test_set):
    entities = []
    label_list = tokenized_test_set.features["ner_tags"].feature.names
    for pred, example in zip(predictions, tokenized_test_set):
        tokens = example["tokens"]
        word_ids = example["input_ids"]
        current_entity = []
        entity_type = None
        for idx, (p, word_id) in enumerate(zip(pred, word_ids)):
            if word_id is None or word_id >= len(tokens):
                continue
            label = label_list[p]
            if label != "O":
                if not current_entity:
                    entity_type = label.split("-")[-1]
                    current_entity.append(tokens[word_id])
                else:
                    current_entity.append(tokens[word_id])
            else:
                if current_entity:
                    entities.append(("".join(current_entity), entity_type))
                    current_entity = []
        if current_entity:
            entities.append(("".join(current_entity), entity_type))
    return entities[:500]
```

All code and experimental configurations used in this study are available at GitHub Repository.