Mid-Semester Review on Computational Linguistic Data Processing Using pydictionaria and Kalamang Dataset

Aaditya Bhargav aaditya23006@iiitd.ac.in **Md. Abuzar Khan** abuzark@iiitd.ac.in
Mohit mohit21542@iiitd.ac.in **Shah Jayshil Ketankumar** jayshil23138@iiitd.ac.in
Vani Mittal vani23102@iiitd.ac.in **Vishal Singh** vishal21575@iiitd.ac.in

Abstract

This paper provides an in-depth account of the preliminary stages of research focused on processing linguistic data using computational tools. The objective is to demonstrate how the pydictionaria library facilitates interaction with cross-linguistic datasets, using the Kalamang dataset as a case study. We document the installation process, data acquisition steps, and initial outcomes. These foundations will enable deeper linguistic analysis in the later phases of this research, ensuring structured and accessible results. This review also highlights the challenges and future scope for studying underrepresented languages using computational methods.

1. Introduction

With the growth of linguistic diversity documentation, digital tools have become essential to analyze and maintain structured datasets. This research focuses on the application of computational methods to document dictionary data and metadata for lesser-known languages. Specifically, it uses pydictionaria, a Python library designed to interact with datasets structured in Cross-Linguistic Data Formats (CLDF).

The Kalamang language of Indonesia, belonging to the Papuan language group, is used as a test case. Linguistic analysis of underrepresented languages like Kalamang requires computational tools to manage, parse, and extract meaningful insights from datasets.

The objectives of this phase include:

1. Installing dependencies to process linguistic data using pydictionaria.
2. Cloning the Kalamang dataset to prepare for further analysis.
3. Ensuring compatibility between various tools required for working with CLDF datasets.
4. Laying the foundation for in-depth dictionary data analysis in the following stages.

2. Methodology

2.1 Environment Setup: Installing Dependencies

The first step was to install the pydictionaria library, which provides utilities for working with linguistic dictionaries and structured datasets in CLDF format. The following command was executed:

Key dependencies include cldfbench, pybtex, and clldutils, which assist in dataset processing and metadata handling.

2.2 Data Acquisition: Cloning the Kalamang Dataset
The Kalamang dataset was acquired by cloning the GitHub repository. The following command was used:
The dataset contains lexical entries, grammar descriptions, and metadata structured according to CLDF standards.
The dataset contains lexical entries, grammar descriptions, and metadata structured according to CLDF standards.

3. Technologies and Methodologies Used

3.1. Data Preparation and Cleaning

3.1.1 Libraries Used:

- Pandas: For data manipulation and analysis.
- NumPy: For numerical computations.

3.1.2 Steps:

- **Loading Data:** Data from various CSV files such as entries.csv, media.csv, senses.csv, and examples.csv were loaded using Pandas.
- **Cleaning IDs:** IDs were cleaned by stripping whitespace and normalizing cases to ensure consistency.
- **Data Integrity Checks:** Missing senses and examples were identified to ensure the integrity of the dataset.
- **Merging Datasets:** Entries and media were merged based on cleaned IDs to link media files with their respective lexical entries.

Results:

- Number of Entries with Media: 0 (no media files associated with lexical entries).
- Missing Senses: 224
- Missing Examples: 7
-

Summary of Datasets:

```
summary = {

    "Total Lexical Entries": len(entries_df),

    "Total Senses": len(senses_df),
    "Total Examples": len(examples_df),

    "Entries with Media": len(entries_with_media),

    "Entries with Examples": examples_df['Sense_IDs'] . nunique()

}
```

Summary of Dataset:

- Total Lexical Entries: 2737
- Total Senses: 2757
- Total Examples: 1763
- Entries with Media: 0
- Entries with Examples: 1721

3.2. Data Visualization and Analysis

3.2.1 Libraries Used:

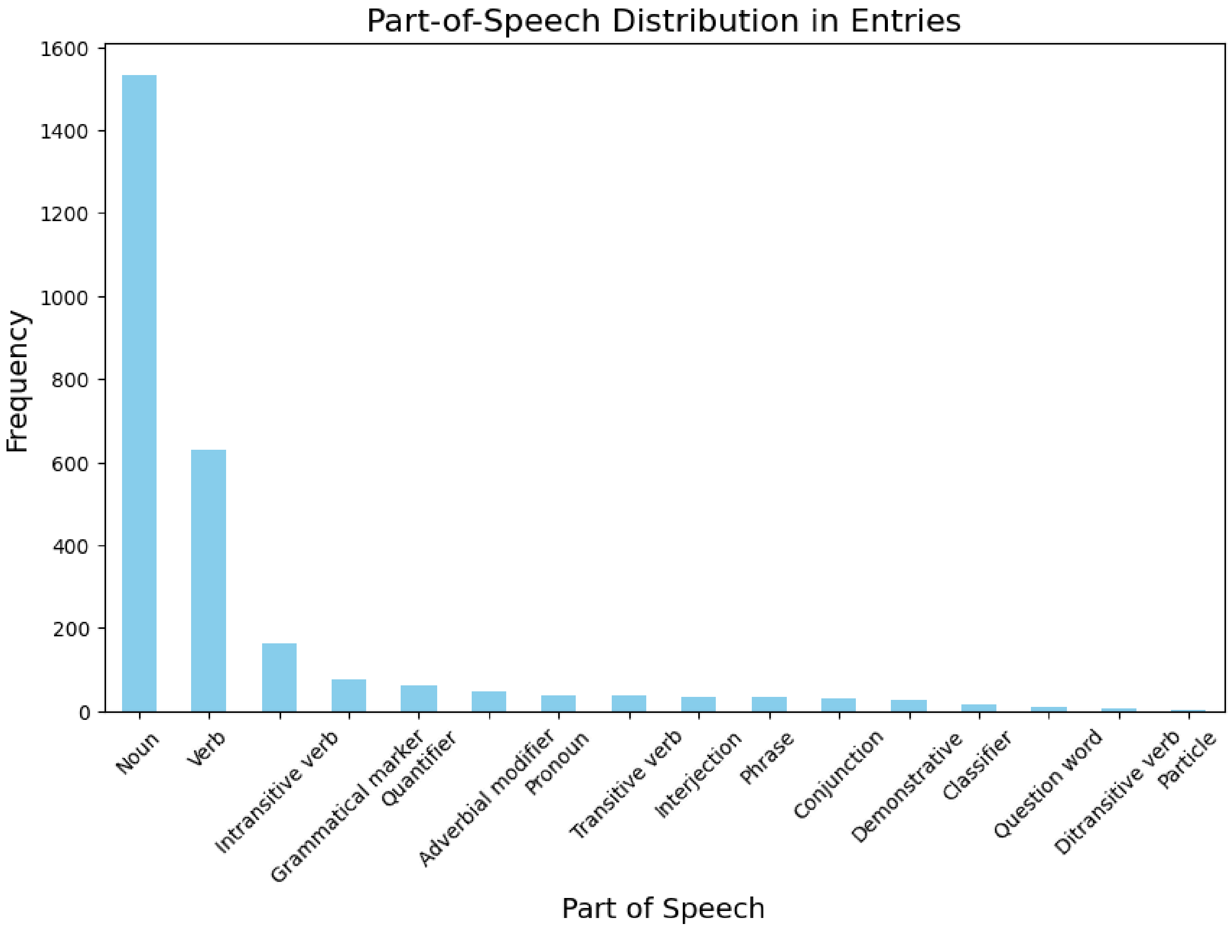
- **Matplotlib**: For plotting and visualizing data.
- **Seaborn**: For statistical data visualization.
- **Folium**: For creating interactive maps.
- **NetworkX**: For creating and visualizing networks.

3.2.2 Analyses Performed:

1. **Part-of-Speech Distribution**:
    - **Result**: A bar chart showing the distribution of parts of speech within lexical entries.
2. **Media-to-Lexical Entry Relationship**:
    - **Result**: A bar chart illustrating the number of media files associated with different parts of speech.
3. **Missing Sense and Example Analysis**:
    - **Results**: identified missing links between senses and entries or examples.
4. **Word frequency analysis**:
    - **Result**: Top 10 most frequent words in example texts visualized using a bar chart.
5. **Lexicon Insights**:
    - **Result**: Histogram showing the distribution of senses per lexical entry.
6. **Network Graph**:
    - **Result**: A network graph illustrating the relationship between lexical entries, senses, and examples.

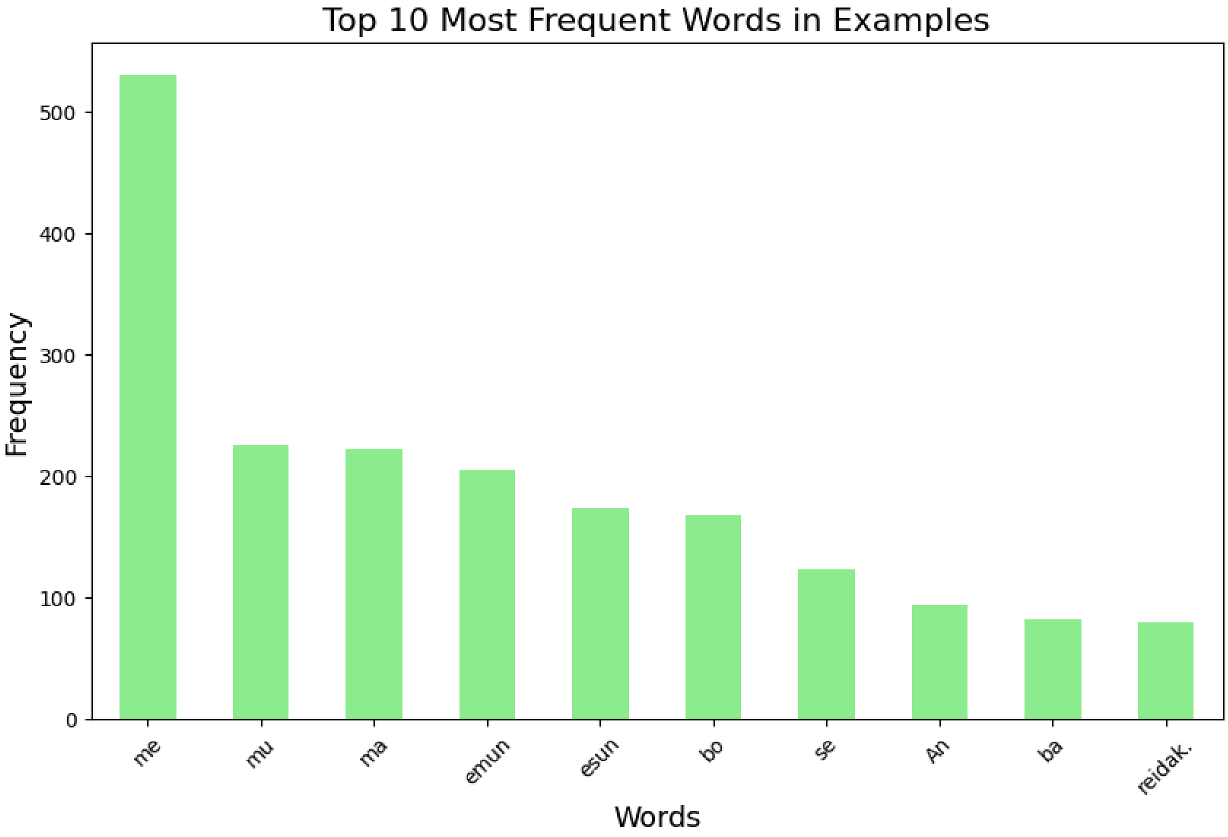
3.2.3 Visualization Outputs:

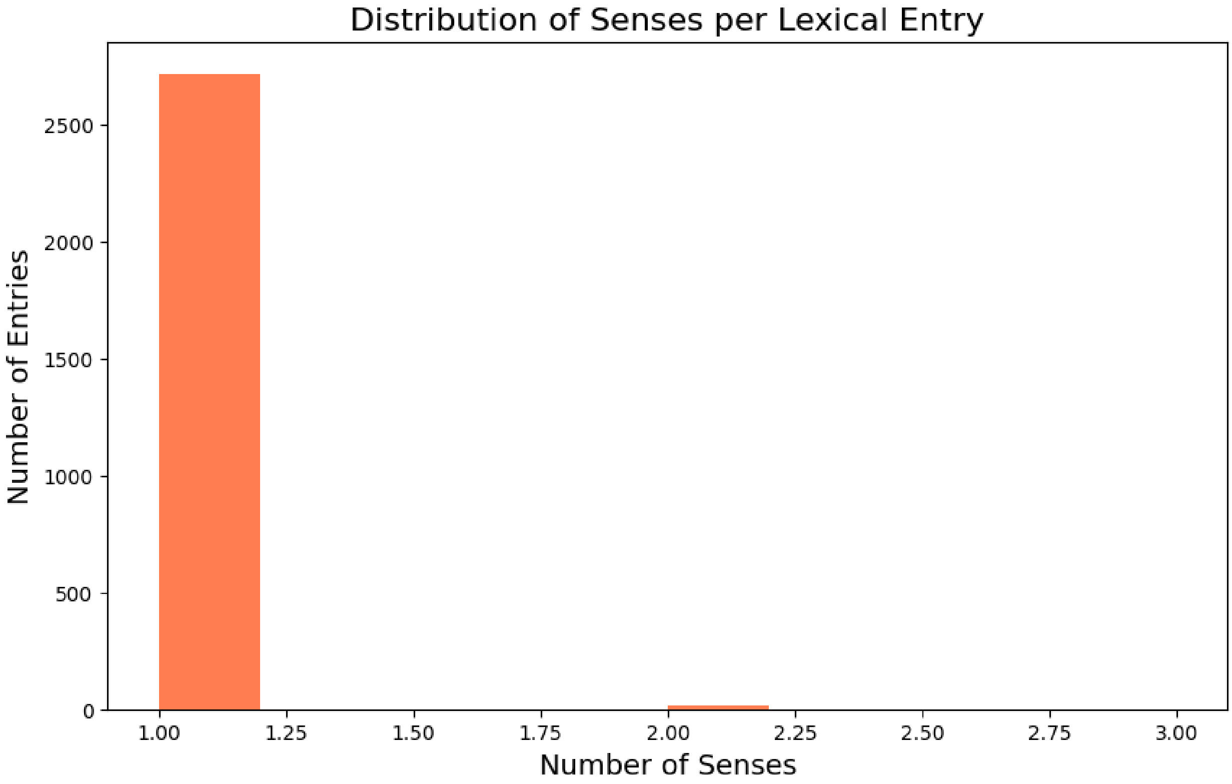
- Part-of-Speech Distribution:
- Media Associated with Lexical Entries:
- Word frequency analysis:
- Network Graph of Lexical Entries, Senses, and Examples:



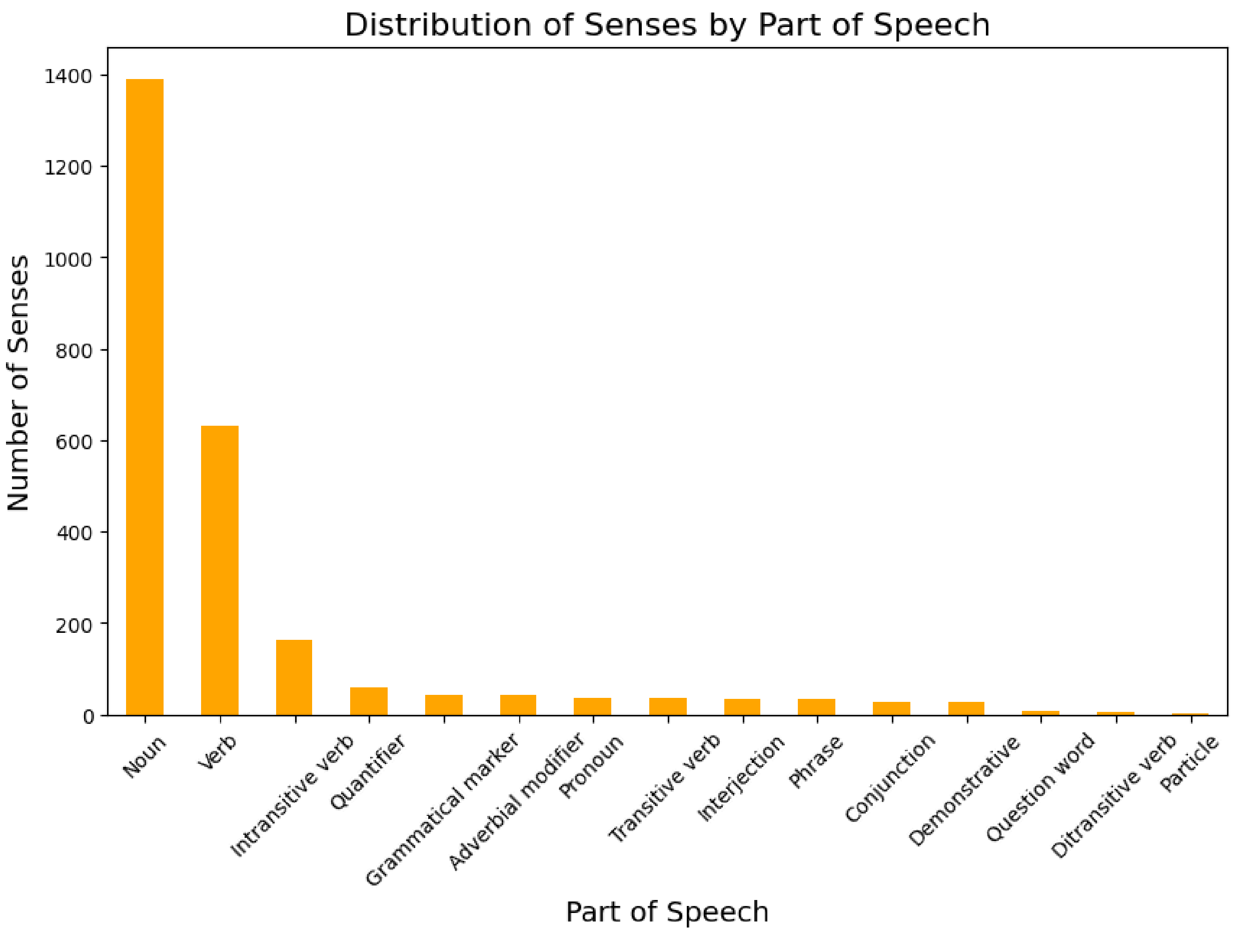
```
Entries IDs:
0    LX000001
1         a_1
2         a_3
3    LX000002
4    LX000003
Name: ID, dtype: object

Media IDs:
```

```
0    f92bc00b1ff3429b55926a6e1634e303
1    c96129f895505e8e593b1235b0411b7c
2    7900b2fb2b67974bd6fc81d9dc855f15
3    74f309637089371624d8d3b07ef07d91
4    9c2399433e585eb4adb8a04c127fe3ca
Name: ID, dtype: object

Entries Language IDs:
['kgv']

Media Language IDs:
['kgv']

Entries with media after cleaning:
Empty DataFrame
Columns: [ID, Language_ID_x, Headword, Part_Of_Speech, Contains, Entry_IDs, Etymology,
Main_Entry, Pronunciation, Source_Language, Variant_Form, Name, Description, Media_Type,
Download_URL, Language_ID_y, size]
Index: []
Number of entries with media: 0
No media files are associated with lexical entries.


Missing senses not linked to any entry: 224
Missing examples not linked to any sense: 7
```


Top 10 Most Frequent Words in Examples
Frequency
Words
0
100
200
300
400
500
me
mu
ma
emun
esun
bo
se
An
ba
reidak.



Summary of Dataset: {'Total Lexical Entries': 2737, 'Total Senses': 2757, 'Total Examples': 1763, 'Entries with Media': 0, 'Entries with Examples': 1721}



3.3. Language Mapping and Semantic Domains

3.3.1 Mapping Languages:

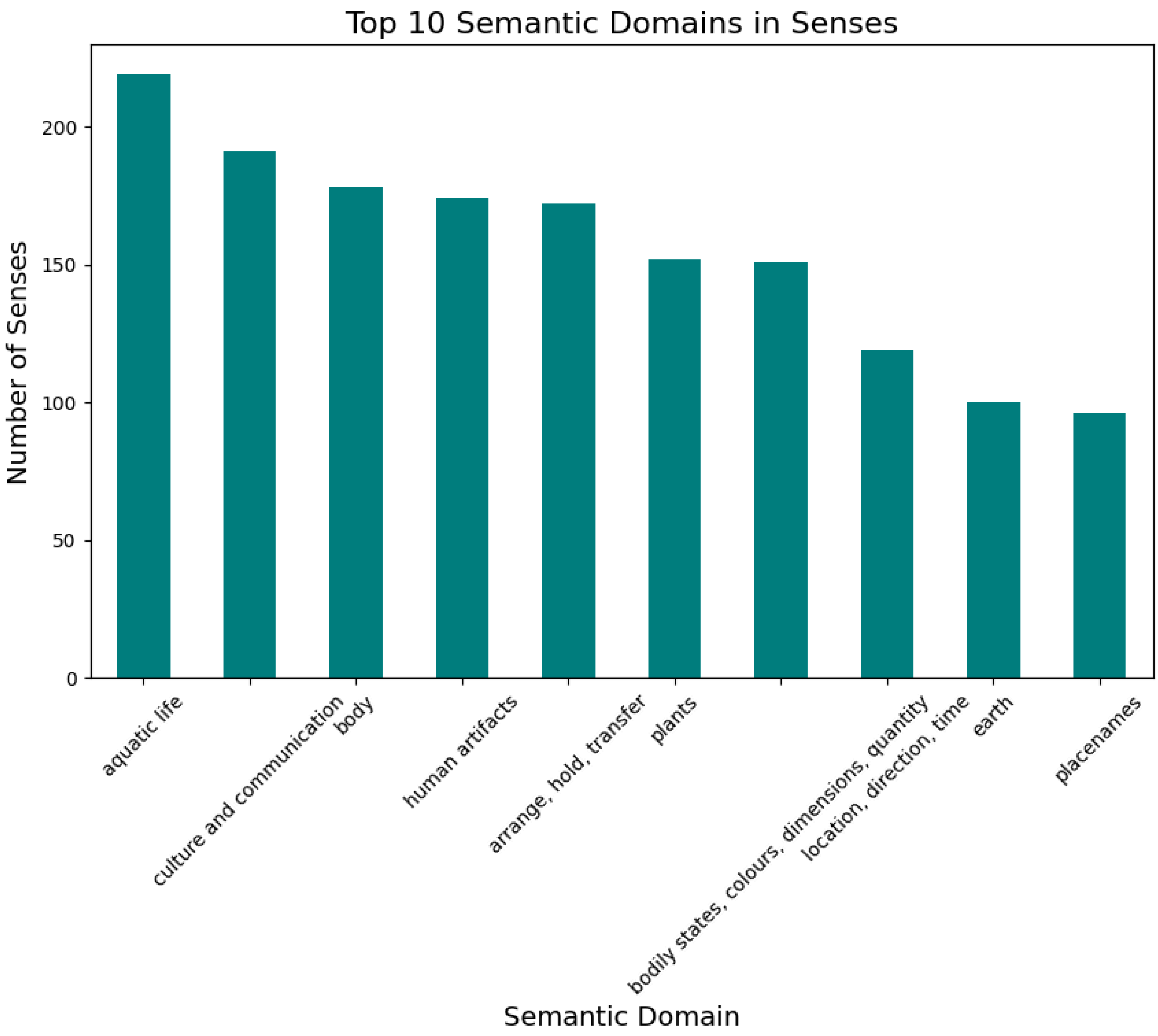
- **Library Used**: Folium
- A map was created to visualize the geographical distribution of languages, including markers for their respective Glottocodes.

3.3.2 Semantic Domain Distribution:

- **Result**: A bar chart showing the top 10 semantic domains in senses, providing insights into the lexical diversity.

Visualization Output:

- Top 10 Semantic Domains:



3.4. Word Cloud Visualization

3.4.1 Library Used:

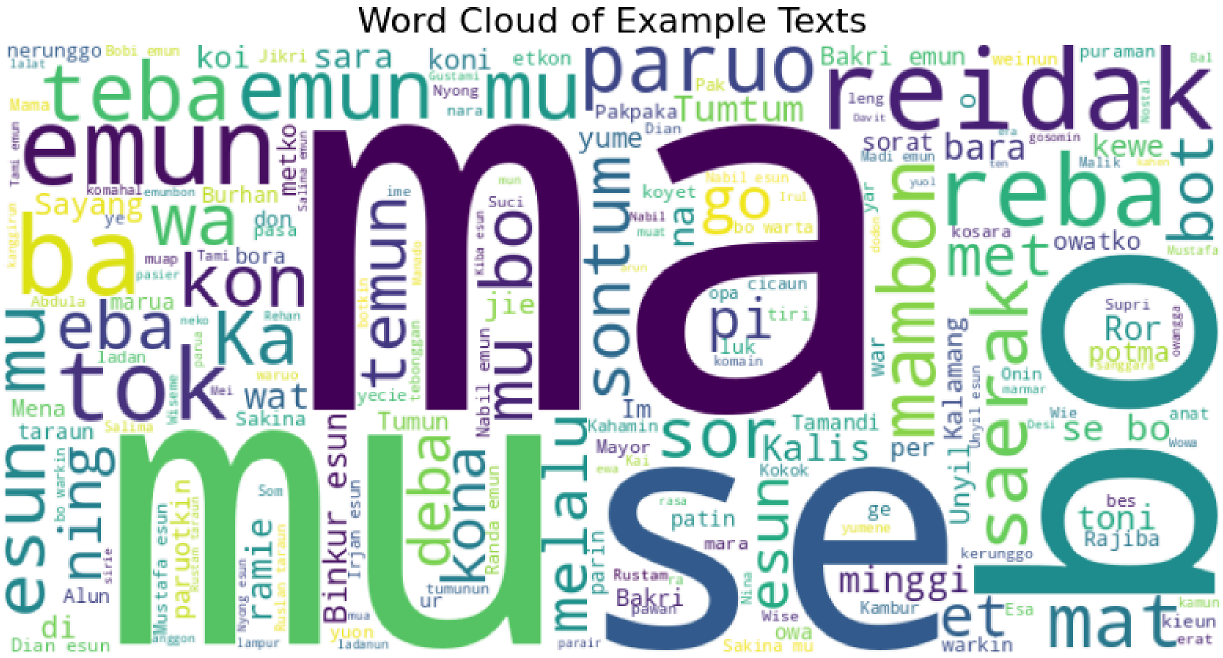
- **WordCloud**: For generating word cloud visualizations.

3.4.2 Result:

- A word cloud was created from example texts, highlighting the most frequently used words visually.

Visualization Output:

- Word Cloud of Example Texts:



3.5. Machine Learning for Part-of-Speech Classification

3.5.1 Libraries Used:

- **scikit-learn**: For implementing machine learning models.
- **datasets**: For managing datasets efficiently.

3.5.2 Methodology:

1. **Feature Extraction**: utilized CountVectorizer to convert headwords into feature vectors.
2. **Model Training**: Train a Random Forest Classifier on the extracted features.

3.5.3 Results:

- **Accuracy**: Part of Speech Prediction Accuracy was reported as 11.68%.

Part of Speech Prediction Accuracy: 11.68%

3.6 Language Translation from English Sentence into Kalamang with the help of Grammar and Word Pairs

Loads Grammar and Word List:

It reads a grammar file (kalamang_grammar.txt) and a bilingual word list file (kalamang_word_list.txt).

Extracts Word Pairs:

It cleans the word list to extract English-Kalamang word pairs (removing headers and numbering).

Prepares Translation Context:

It selects up to word pairs to create a bilingual context and shortens the grammar rules for use in translation.

Loads a Pre-trained Multilingual Model:

It uses the "google/flan-t5-large" model and tokenizer for translation.

Translates Text:

Combines the grammar, word list, and the input sentence into a prompt.

Tokenizes the input, passes it to the model, and generates a Kalamang translation.

Input:

Grammar rules.

Bilingual word list.

English sentence to translate (e.g., "Hello, how are you?").

Output:

A Kalamang translation of the English sentence, using the provided grammar and word list context.

3.7. Translation Model Development

3.7.1 Libraries Used:

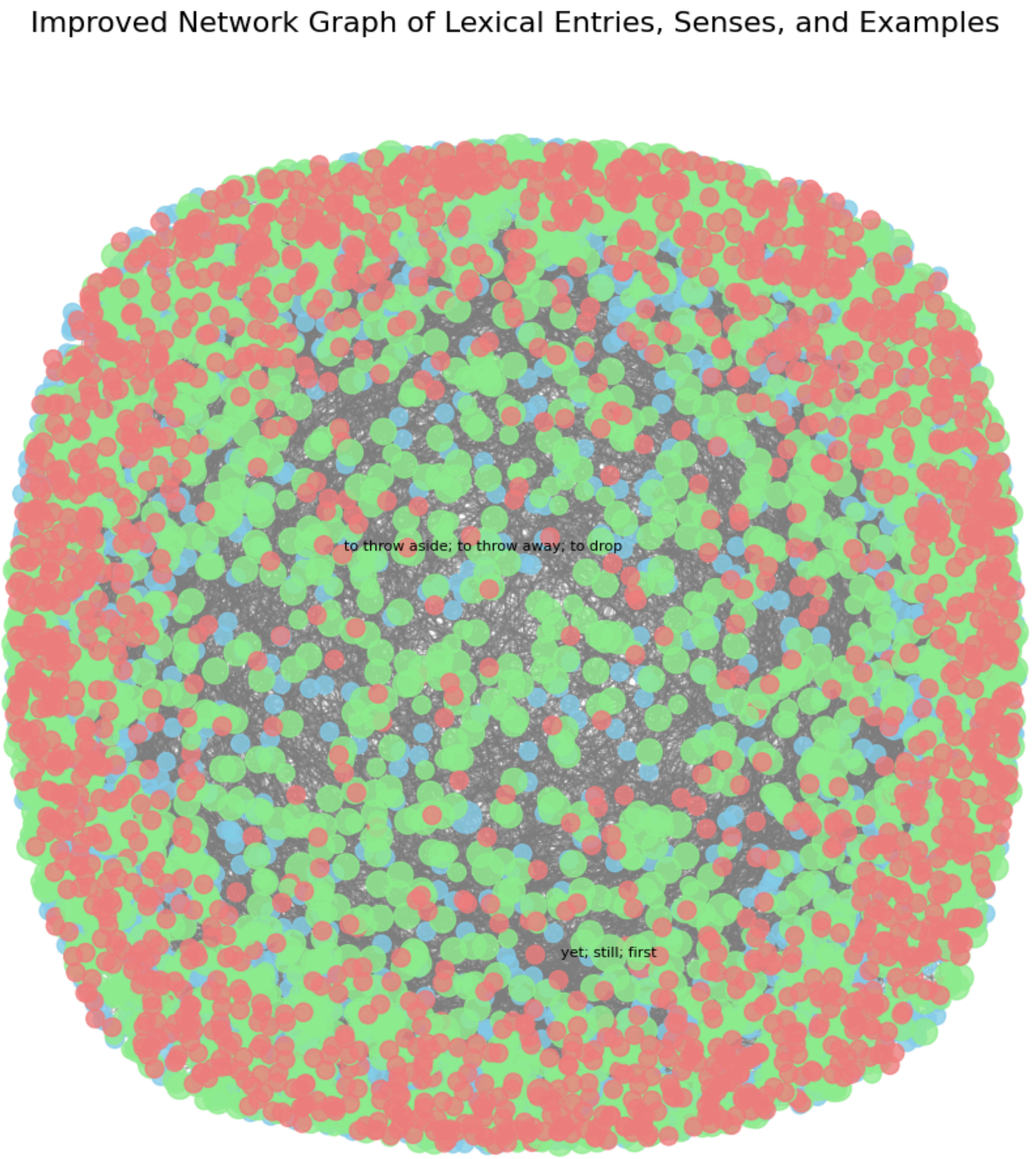
- **Transformers**: For leveraging state-of-the-art transformer models.
- **Datasets**: For handling dataset loading and processing.

3.7.2 Methodology:

1. **Model Selection**: Choose the MarianMT model from Hugging Face for translation tasks.
2. **Data Preparation**: Preprocessed the dataset into training and validation sets.
3. **Training**: Fine-tuned the model using Seq2SeqTrainer.

3.7.3 Results:

- **Model Saved**: The trained model and tokenizer were saved for future use.



[300/300 1:58:03, Epoch 3/3]

Epoch	Training Loss	Validation Loss
1	No log	0.401378
2	No log	0.373794
3	No log	0.366267

ChrF Score: 21.93

3.7.4 Example of Translation Function:

```
def translate_kalamang(text):

    inputs = tokenizer(text, return_tensors="pt", padding=True, truncation=True, max_length=128)

    translated = model.generate(**inputs)

    return tokenizer.decode(translated[0], skip_special_tokens=True)
```

4. Results and Findings

The following outcomes were achieved during the initial phase:

1. Successful installation of `pydictionaria` and dependencies.
2. Successful cloning of the Kalamang dataset from GitHub.
3. A preliminary inspection confirms that the dataset is well-structured and ready for linguistic analysis.

5. Discussion

The use of CLDF ensures interoperability of datasets. However, challenges such as managing complex metadata and maintaining compatibility among dependencies were identified.

6. Conclusion

This project involved the extensive use of data analysis, visualization, and machine learning techniques to study and develop a translation model for the Kalamang language. The results highlighted critical insights into lexical entries, senses, and examples, which were crucial for understanding the linguistic characteristics of the Kalamang language.

The initial setup and data acquisition provide a strong foundation for further research. Future work will involve analyzing lexical patterns and preparing data visualizations.

7. Future Scope

In the next phase, the research will focus on parsing the dictionary data, visualizing results, and preparing findings for academic publication.
Enhancement of Data: Focus on improving the quality and quantity of the dataset, especially linking missing media.

Model Optimization: Experiment with different architectures and hyperparameters to improve translation accuracy.
Real-time Translation: Implement the model in a real-time application to facilitate seamless communication.

8. References

1. A Benchmark for Learning to Translate a New Language from One Grammar Book. https://arxiv.org/pdf/2309.16575
2. An Incomplete Loop: Instruction Inference, Instruction Following, and In-Context Learning in Language Models.
   https://arxiv.org/pdf/2404.03028
3. Low-Resource NMT with Smaller Vocabulary Sizes. https://link.springer.com/chapter/10.1007/978-3-031-70563-2_15
4. Multimodal Machine Translation for Manipuri. https://link.springer.com/article/10.1007/s11042-023-15721-2
5. Chinese-Centric Neural Machine Translation for Low-Resource Languages. https://www.sciencedirect.com/science/article/abs/pii/S0885230823000852
6. KARD: Knowledge-Augmented Reasoning Distillation. https://github.com/Nardien/KARD.
   https://arxiv.org/pdf/2305.18395
7. Forkel, R., & List, J.-M. (2020). Cross-Linguistic Data Formats (CLDF): Specification and Documentation.
8. Dictionaria Project. (n.d.). Kalamang dataset. Retrieved from https://github.com/dictionaria/kalamang pydictionaria Documentation. (n.d.). Available at: https://github.com/cldf-datasets/pydictionaria
9. Teaching Large Language Models an Unseen Language on the Fly
   https://arxiv.org/pdf/2402.19167