QUITA-mini

Student Name: Kıvanç Gördü

Subject: Current Approaches to Grammar Theory (KOL/STEG)

Department: Department of Linguistics

School: Palacký University Olomouc

Date: 27.04.2025

Github Profile: https://github.com/kivanc57

Project Overview

QUITA-mini is a text analysis tool that calculates various linguistic metrics from text data. Its name is inspired by the QUITA Online service. It processes a collection of text files, calculating metrics such as the *Type-Token Ratio (TTR)*, *entropy*, *average token length*, *hapax legomena percentages*, and other related linguistic indexes that are commonly used. The tool outputs these results in an Excel file, following a similar philosophy to QUITA.

Specifically, this program reads text files within a given directory and concatenates them to generate both a lemma frequency list and a token frequency list. It then calculates indexes one by one, mirroring the default computations of QUITA. Subsequently, each index and its corresponding value is written to an Excel sheet to present the results.

In my opinion, the greatest strength of this project lies in its computational efficiency and its remarkable speed compared to the original service. Generally, the Go programming language effortlessly outperforms Python and Perl, which are used to build the original service. Furthermore, some of the key advantages mentioned above stem from the fact that I developed this package with minimal reliance on external libraries and by adhering to a single programming language. Last but not least, this is also an open-source version that can be used by anyone independently on their local machine, without the need for an internet connection or any service fee.

By no means is this intended to compete with the service, but rather it serves as an experimental approach where I could leverage my programming skills while applying my academic knowledge and combine them in a project to make an impact.

More details about this project and the QUITA research article are available on the Github repository attached below:

https://github.com/kivanc57/quita_mini

Features

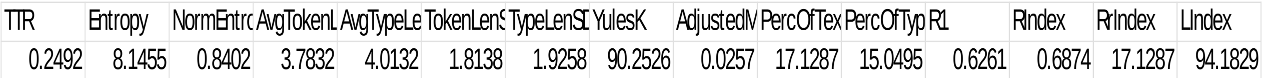
- **Text Analysis**: Tokenizes and processes text to calculate various linguistic metrics.
- **Metrics Calculated**:
    - Type-Token Ratio (TTR)
    - Entropy
    - Normalized Entropy
    - Average Token Length
    - Average Type Length
    - Token Length Standard Deviation
    - Type Length Standard Deviation
    - Yule's K
    - Adjusted Modulus
    - Percentage of Text Hapax
    - Percentage of Type Hapax
    - R1, R Index, Rr Index, and L Index
- **Excel Output**: Saves results in an Excel file for further analysis or visualization.

My Usage Case

I wrote the scripts with functionalities for file and data type operations, culminating in the final results presented in an Excel sheet. Throughout this process, my primary focus remained on efficiency, simplicity, and modularity.

I begin by reading any text file located within the specified directory. I then process each file's content, writing it byte by byte and merging it into a strings.Builder data type to efficiently handle potentially large datasets. All processed file content is then assigned and concatenated into a single string variable, *corpus*. This corpus variable essentially represents the amalgamation of all input files into one continuous text string. Following this concatenation, I generate both a *lemma frequency list* and a *token frequency list*, which serve as the foundational data structures for subsequent calculations. For tokenization, I use regular expression utilized from 'regexp' Go library, particularly **[^a-z]+** and replace every character with the findings on the string.

Subsequently, I pass these frequency lists (both *map* data types) to a suite of carefully crafted calculation functions, each designed to compute a specific linguistic index. These functions implement the majority of the linguistic indexes available in QUITA. The results of these calculations are then stored directly into a slice data type. This slice is subsequently used to populate the cells of the Excel spreadsheet, with the column headers derived from the names of the calculated linguistic indexes. Once the rows are populated with the calculated values, the process concludes with the creation of an Excel file named 'results.xlsx'. Below, a fragment of this Excel application is attached to show an illustration of the output format.



TTR	Entropy	NormEntro	AvgTokenL	AvgTypeLe	TokenLenS	TypeLenSD	YulesK	AdjustedM	PercOfTex	PercOfTyp	R1	RIndex	RrIndex	LIndex
0.2492	8.1455	0.8402	3.7832	4.0132	1.8138	1.9258	90.2526	0.0257	17.1287	15.0495	0.6261	0.6874	17.1287	94.1829

Installation

1. Clone this repository to your local machine.

2. Ensure that you have Go installed. If not, you can download and install it from here.

3. Install the necessary Go packages:

   `go get github.com/xuri/excelize/v2`

4. Install other dependencies by running:

   `go mod tidy`

Script Usage

1. Set the ‘absDirPath’ variable in ‘main.go’ to the path of the directory containing your text files.

2. Run the program using:

   `go run main.go`

3. After execution, the results will be saved to ‘results.xlsx’ in the current directory.

License

This project is licensed under the GNU 3.0 License - see the GNU GENERAL PUBLIC LICENSE file for details.

Acknowledgments

- This project uses the Excelize library for handling Excel files.