UNIVERSITY OF MORATUWA

Faculty of Engineering

Final Year Project
Progress Report
Advanced Interactive Sinhala Dictionary

Group 49
170210J W.A.M Hasaru
170243L: K.I.L.Isuranga
170388J: W.G.P.Mihiranga

Supervisor: Prof. Gihan Dias
Dr. Nisansa de Silva

Table of Content

1. Introduction

Sri Lanka is the only country in which most people speak the Sinhala language. Since the country is small and has less population, the Sinhala language is becoming an endangered language day by day. Every language has an equal value and rich literature attached to it. It is necessary to keep that language alive to carry the rich literature and knowledge written in that language.

As similar to every other language, the Sinhala language also has a rich vocabulary. One word can have multiple meanings concerning the context in which the word is used. Because of that Sinhala language vocabulary becomes more complex based on the sentiments.

Example:
පංගුව, පඬිගුව [නාම පද.]
අ. කොටස; භාගය.
ආ. අඛණ්ඩ ව නොබෙදීපවතින ඉඩම් කට්ටිය.
ඇ. යම්සේවයක යෙදී සිටින ජනකොට්ඨාසය.
ඈ. වර්ගය; ප්‍රභේදය.

As per the above example, the same word can be used in different contexts to give different meanings. It is necessary to build a centralised knowledge base with all these language vocabularies to preserve the language resources.

With the rise of the research area of Natural Language Processing and Data Science, researchers worked hard to bring their native languages to computer-based platforms to preserve them. The dictionary of a language is the primary key that helps to understand the language for its existing users as well as the new users. Many of the research for Dictionaries are based on European and East Asian languages due to the user community they cater to.

This research project is focused on building an Advanced Interactive Sinhala Dictionary which consists of features that can not be seen in the Sinhala Dictionaries which are already available. This will be extended to a Sinhala, English, Tamil trilingual dictionary as well. This will preserve the Sinhala Language in cloud platforms, and It will provide API which will be very helpful to future research projects as well.

2.Problem Statement

Foreign languages like Chinese, Spanish, and English have many digital dictionary software and online dictionaries with rich features. We researched the publicly available dictionaries of the world's most used languages and their features. [1]

Even Though the other languages used in the world are rich in online resources, the Sinhala language still falls behind compared to them. Sri Lankan researchers are working hard to make the Sinhala language resources publicly available via the internet. Due to some reasons, some of the research approaches which are taken to build a versatile interactive Sinhala Dictionary have come to an end and those products are not working properly at the moment.

To identify the Sinhala language dictionaries which are publicly available as of now, we did another research. We identified the dictionaries and the promising features they provide to the users and documented them.[1]

There are few working Sinhala - English dictionaries available but the only Sinhala to Sinhala online dictionary which gives a Sinhala definition for a Sinhala word we could find was the one from the Sinhala Dictionary Compilation Institute [2]. But it has no features other than finding meaning for a given word and also they only have half of the dictionary available. Compared to widely used English dictionaries such as Cambridge Dictionary [3], it lacks user-friendly features and the Derana dictionary [4] (Sinhala - English) has been implemented to give out example sentences for word search. Although that function is not fully implemented since it’s a beta version and it’s always asking for suggestions for improvement and gives out incorrect statements. The trilingual dictionary from the department of official languages is not available most of the time.

The Sinhala language has a lot of similar-sounding letters which is true for almost all of the children of the Sanskrit language. Some Sinhala words have different spellings with different meanings with the same pronunciation. This is a common complexity for every other language as well. When compared to the English language, the alphabet of the Sinhala language is high in complexity which gives different sounds from the same base sound.

Example: ග ඝ ඟ

Apart from the rich alphabet, it also has a set of symbols called “පිල්ලම්” which makes different forms of the base sound.

Example: In English, different sounds are represented by combinations of different letters.

English: M + U -= Mu

Sinhala: ම් + උ = මු

As per the above example, the sound is formed combining a vowel and a consonant sound but it is written by adding a “පිල්ලම්” to the base consonant. For the same sound form, different “පිල්ලම්” is used with different consonants.

Examples: මු කු

Likewise, several interpretations form a huge set of vocabulary.

All of this makes finding a word in a dictionary quite difficult for Sinhala.

As in any other language, Sinhala words have morphological forms. Available Sinhala dictionaries only contain only one form of a word. Having the ability to find any form of a word is a very useful feature for a digital dictionary.

Adding these features with an online tool can make a very easy-to-use dictionary tool for the Sinhala Language.

Research Objective

Our project is building “Advanced Interactive Sinhala Dictionary” a Sinhala - Sinhala dictionary with advanced interactive features. This will give a comprehensive Sinhala definition for a Sinhala word and the dictionary will be extended to a trilingual dictionary (Sinhala - English - Tamil) as well. Since the lack of digital tools, dictionaries for the Sinhala language with the compatibility of the other two main languages in Sri Lanka, this system is much in need.

As the outcome, this project will deliver a system consisting of a web app, Android mobile app, and web API, which can provide an advanced dictionary that will define a given Sinhala word where a user can also search a word with different spellings which relate to a form of the same verb. It will include many advanced interactive features as mentioned in the methodology. The proposed system will be handed over to the user in the form of a web application and mobile application.

Literature Review

There are few Sinhala to Sinhala dictionaries available in printed format. They are සිංහල ශබ්දකෝෂය By Sinhala Dictionary Compilation Institute, සංක්ෂිප්ත සිංහල ශබ්දකෝෂය By Sinhala Dictionary Compilation Institute, ගුණසේන සිංහල - සිංහල ශබ්දකෝෂය By M.D.Gunasena.

The only online Sinhala to Sinhala dictionary is Sinhala Concise Dictionary available online at the website of Sinhala Dictionary Compilation Institute. According to the website, only a part of the printed dictionary is available online. This dictionary only provides a basic search with word suggestions. [2]

For an online dictionary, it is essential to provide the users with external links for related documents and corpora for further research. According to research by Carolin Muller Spitzer from Institute for German Languages [5], it is indicated that adaptability and multimedia are the least concern among users and the features mentioned prior are the most concern for the users when it comes to an online dictionary.

After researching many online dictionaries we decided that the below features should be embedded as the core features of our dictionary.

Eg: Word pronunciation, Related words with definitions are given with the search results, Synonyms, and antonyms, Example sentences, Word forms, Input methods.

In our proposed system, the primary data source is the digital version (PDF version) of the above Sinhala Concise Dictionary. Thus PDF extraction is a major part of our project. For the trilingual data set, we are sourcing the Trilingual Dictionary with Sinhala Head Words (Sinhala - Tamil - English) from the Department of Official Languages and we are granted permission to extract the data for our use.

Representation of PDF is optimised for printing. PDF format contains text, tables, figures, and images, and their location and layout information. But it does not contain structured data. PDF files created with different tools use different methods to layout content while encoding. So extracting textual data from PDF files is a difficult problem. Paper “Improved Text Extraction from PDF Documents for Large-Scale Natural Language Processing” describes a method for PDF data extraction that is automated and suitable for batch processing. Researchers of this paper use existing tools for extracting text and use their post-processing steps to fix issues in extraction. Then they build an XML file that contains text and structure information. [6]

The internal representation of PDF format is concerned with the view of the document. This fact makes PDF extraction a difficult task. In “Intelligent Text Extraction from PDF Documents”, researchers describe methods for extracting structured information from pdf. They have developed a page segmentation algorithm that segments the content of PDF into words, lines, text blocks, and columns. The segmentation algorithm is made with both bottom-up and top-down processing. Both methods have their advantages and disadvantages.[7]

Spelling errors have two main types, non-word errors, and real-word errors. Non-word errors happen when the given word is not a valid word or a valid form of a word. In “A Data-Driven Approach to Checking and Correcting Spelling Errors in Sinhala” Unicode text input is tokenized and builds a list of unique Sinhala words. Then based on the phonetic similarity of letters, similar-sounding word permutations are generated. Using the n-gram statistics computed from the UCSC Sinhala Corpus, the best suggestion is selected by the Best suggestion selection module. Then post-processing happens and the text is replaced with the best suggestion words.[8]

Morphological analyzers can extract morphological information from a given word. It can also generate other morphological forms for a given word. Research paper “A Morphological Analyzer to Enable English to Sinhala Machine Translation” has built a morphological analyzer from scratch for the Sinhala language. It is can be used to expand the search function of the dictionary.[9]

Autocomplete is a very useful feature when searching a word in the dictionary. For the proposed system, Past search data can be used as the dataset for autocompletion or an API like Google autocomplete API can be used for getting autocomplete suggestions. Using past search data might not be accurate enough because the dataset can be messy with wrong search queries with misspellings. Using an API will give better results. Correcting search-term spelling is a key way in which users choose to make use of the autocomplete feature. Autocomplete improves the quality of initial queries for both known and exploratory tasks. Suggestions provided by the autocomplete validates the user's search idea as well. It should be implemented in a way that the suggestions are not perceived as slowing down the search process and it should be easy to read and quick to be selected. Autocomplete can be optimised for known-word searches and also locating alternative options for researching words.[10]

Specifications

- Unicode Mapping and PDF Extraction Completion
    - A tool to extract data from Sinhala concise dictionary volume 01 and volume 02.
- Backend Implementation with Basic Features (Headword Search) with API
    - A dictionary backend will be implemented with an endpoint to search headwords from the extracted data set.
- Search Word with Morphological Forms
    - Backend will be upgraded with the ability to search a word from its morphological form and based on the morphological roots, suggestions will be given.
- Spell Error Handling and Similar Sounding Word Suggestions
    - Spell correction tool which is developed by the “Sinspell” team will be used for spell error correction and it will be improved to generate similar sounding word suggestions.
- Frontend UI Design
    - Frontend UI Web and Mobile Apps will be developed and integrated with the backend.
- Look up all words in a text
    - Ability to look up words in given text at once will be implemented.
- Trilingual Support
    - Trilingual support will be added by extracting the data from the Trilingual dictionary.
- UX Improvements (Favourite Word List, Suggest entries, etc)
- Testing and Surveying from the public

Methodology

The project is to build an advanced interactive dictionary for the Sinhala language and extend it to a trilingual dictionary that supports Sinhala, Tamil, and English languages and the following procedure will be followed.

1. Extract words and definitions from existing dictionaries which are in pdf formats and store them in proper data structures.
    a. Sinhala Concise Dictionary
    b. Sinhala Tamil English Trilingual Dictionary
2. For extraction, we are implementing tools to extract data from PDF.
3. Then we implemented another tool to convert the non-Unicode font to Unicode by writing a mapping between Unicode and non-Unicode fonts. Create a database with extracted data.
4. We save the extracted dictionary data to a YAML data structure. This will preserve data to use in future research as well. Researchers can parse the data from the YAML structure easily using a YAML parser. This is advantageous to us as well. We can easily adapt to different data structures with the need of the features and data can be easily extracted from the YAML.
5. Build the system to retrieve the searched word definitions.
6. Implement the core features and make a working product.
7. Backend with API endpoints
8. Frontends with basic dictionary features.
9. User Experience improvements will be done based on the survey feedback.
10. The documentation process will be carried out parallel to the development.
11. Publicly release the product with documentation for users.

PDF Extraction

Sinhala Concise Dictionary is a Sinhala monolingual dictionary compiled by Sinhala Dictionary Compilation Institute. This dictionary has two volumes and is available in PDF and printed format. The PDF version of the dictionary is used as the data source of this project. The PDF version of the Trilingual Dictionary by the Department of Official Languages is used to gather Tamil and English meanings of head words.

First step in the project was to extract textual data from the PDF version of the above dictionary. Initially, following tools were used to extract the text:

- PyMuPDF (Python Library)
- PDFMiner (Python Library)
- PDFtk
- Unicode Pleco

Importance of the structure of text inside the pdf was realised.

- Removed pages other than the definition pages. (removed Cover pages, Legends, etc)
- Cropped the PDF files to remove the header and footer.
- Used the positional information of text fragments to detect the starting of a new entry.

We got a code implemented by Gayan Kavirathne for extracting text from the dictionary pdf. It was implemented using Java and ITextPdf Library. It was able to correctly split text into individual definitions in most cases. So we decided to use his code for text extraction. But it had few major issues and they were fixed to complete the text extraction part.

- Missing spaces at the end of some lines.
- Detection of definitions is not working for some pages.
- Can’t handle definitions that continue to the next page.
- Bold letters are repeated multiple times.

Used page margins set in the code instead of cropping PDFs to remove headers and footers.

Text in the PDFs are in legacy fonts. Unicode is preferred for using in the website and mobile application. Above code had unicode conversion which is taken from Unicode Pleco. But it had some issues especially with rare letter combinations.

- Pillam that comes before the letter (eg: ෙ◌, ෙ◌ෙ◌) are not converted due to the way it was integrated with the text extraction code.
- Handling non standard letters. Eg:
    - Use of letter “f” with “ස” to denote “ෆ”.
    - Use of “ඬ” in place of “ඩ”

For converting Tamil non-unicode fonts to unicode, unicode conversion code in Unicode Pleco is used without modification.

Tried Py-Tesseract OCR Library:

- We tried to use an OCR based approach to improve the correctness of PDF extraction. Among a few libraries, the Tesseract OCR library gave us good results but it did not meet the required correctness.
- We tried to improve results by tuning the parameters of the OCR library but it gave a set of errors for one set of parameters and then another set of errors for other parameters.
- We identified that OCR library does not perform well in identifying correct ‘Pillam”
    - Ex: වෙළුම්පට

Data Structure

Previously extracted PDF data was written into a formatted text file in order to store in a data structure.

```
අංගාගී
        අංග+අංගී
                [නා. ප්‍ර.]
                        අංග සහ උපාංග; ශරීරය සහ අත් පා ආදි අවයව.
                [වි.]
                        අංග සහ උපාංග දරන.

අංගාඬිය
                [නා.]
                        කඩ වීදිය; වෙළඳ වීථිය.
```

Figure : formatted concise dictionary text file

Concise Dictionary Data

A python script was written to store this text information in a YAML file. The way indentation was used in the formatted text file made converting the file to YAML easier.YAML was used since it’s legible thus making the manual error correction effective.

```yaml
- word: පදානුගත
  root: null
  comb: පද+අනුගත
  poss:
  - pos: '[වි.]'
    meaning:
    - වචනයෙන් වචනය අනුව යන; වචනාර්ථය මුල් කරගත්.
- word: පදානුපදික
  root: null
  comb: පද+අනුපදික
  poss:
  - pos: '[වි.]'
    meaning:
    - යමකුගේ පා සටහන් අනුව ගමන් කරන.
    - වචනයෙන් වචනය, පදයෙන් පදය අනුගමනය කරන.
```

Figure : YAML file

Trilingual Dictionary Data

Same as the concise dictionary, the trilingual dictionary was also formatted and prepared for storing in YAML data structure.

```
අංගනාව (s.)
                    பெண், மங்கை
                    woman
අංගනාවෝ (pl.)
අංගපූලාව (n.)
                    நல்ல நடத்தை, நன்னடத்தை
                    good behaviour
```

Figure : formatted trilingual dictionary text file

Trilingual dictionary was also stored in YAML format as below.

```
- word: අංකය
  pos: (s.)
  tam: இலக்கம், நாடக அங்கம்
  eng: number, act in a play

- word: අංක ගණිතය
  pos: (n.)
  tam: எண்கணிதம்
  eng: arithmetic

- word: අංක තහඩුව
  pos: (s.)
  tam: இலக்கத் தகடு
  eng: number plate
```

Manual error correction was needed in formatting the trilingual dictionary. The pos tags were identified when there are round brackets, but there were some cases where round brackets are used in the middle of english word definition. Therefore the python script outputted undesirable formattings in some cases.

Part of Speech (POS) tag replacement

Concise dictionary POS tags were written as Sinhala letter abbreviations. Trilingual dictionary POS tags were written as English abbreviations as shown below.

රසවිද්‍යාව (n.)
        இரசவாதம்
        alchemy
රස විඳිනවා (vt.)
        அனுபவி, நய
        enjoy appreciate

Figure : formatted concise dictionary text file

අංගාංගී
    අංග+අංගී
        [නා. පු.]
            අංග සහ උපාංග; ශරීරය සහ අත් පා ආදි අවයව.
        [වි.]
            අංග සහ උපාංග දරන.

Figure : formatted concise dictionary text file

Since the user might have a hard time understanding these POS tags we decided to replace the POS tags with their actual meanings.
The Concise dictionary and Trilingual dictionary are both manually typed dictionaries. In some cases there were different variations of typing for the same POS tag.

In the Concise dictionary "[ක්‍රිවි.]", "[ක්‍රි..වි.]", "[ක්‍රි.වි.]" etc. were used as the POS tags for adverbs “ක්‍රියා විශේෂණ”.

In the Trilingual dictionary “(s..)”,”(s.)”,”(. s.)” dictionary etc. were used as the POS tags for Singular noun “ඒක වචන නාම පදය”.

A python script was written to replace all these differently typed tags that have the same POS tag with their actual definitions. As in "[ක්‍රිවි.]", "[ක්‍රි..වි.]", "[ක්‍රි.වි.]" were replaced by [ක්‍රියා විශේෂණ] and “(s..)”,”(s.)”,”(. s.)” the were replaced by [ඒක වචන නාම පදය]. This helped to improve consistency among the pos tags of the two dictionaries.

Backend Server Implementation

Most of the data processing part is done in the backend leaving only the data viewing part for the frontend. Backend was implemented using Python and Fast API framework. Python was chosen because it has many packages related to natural language processing. FastAPI was chosen as the web framework because it is faster than other python frameworks and it provides automatic API documentation.

Backend provides:

- Definition for the given word.
- Alternative word suggestions using spell correction and morphological analysis.

Morphological Analysis

Using a morphological analysis within the dictionary itself gives us the chance to embed more useful features.

SinMorphy

SinMorphy is a rule-based system with a comprehensive vocabulary of Sinhala words. Therefore, it accurately handles a great majority of contemporary Sinhala text. It can also synthesise lexical forms of words, and tag their parts of speech.The system is based on a finite state transducer, and is written in the Foma and Lexc languages. It handles all types of words including nouns, verbs, compound nouns, adjectives, adverbs and particles. It also includes a guesser to analyze out of vocabulary words. It correctly analyses 94.7% of the most common 10,000 Sinhala words, and has an accuracy of 89.7% on a random test set of 1000 words.[11]

Extracting roots of the dictionary headwords

The headwords were extracted from the dictionary and were fed into the morphological analyser. The output which contained the headwords and their morphological information (root,lexical forms of words) were written into YAML files.

```
- - සහෝදරයා
  - - - සහෝදර+N+M
      - +SG
      - +DF
      - +ACC
    - - සහෝදර+N+M
      - +SG
      - +DF
      - +NOM
- - ලියනවා
  - - - ලිය
      - +V
      - +FIN
      - +NPST
      - +IND
```

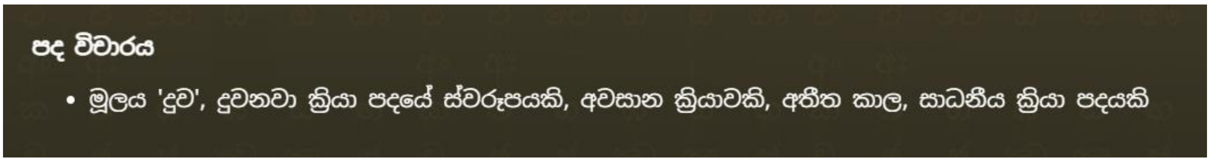
For compound words, the root was considered as the root of the first word.

```
- - බෙදුම් නඩුව
  - - - බෙදුම+
      - N+
      - NUTR
      - +PL
```

```
    - +ACC
  - - බෙදුම+
    - N+
    - NUTR
    - +PL
    - +NOM
```

These lexical tags are shown to user with their meanings such as N = noun (නාම පදයකි), V=verb (ක්‍රියා පදයකි), M= Masculine(පුරුෂ ලිංග) , F= Feminine (ස්ත්‍රී ලිංග) etc.

This morphological analysis is shown under the definition of the word.



පද විචාරය

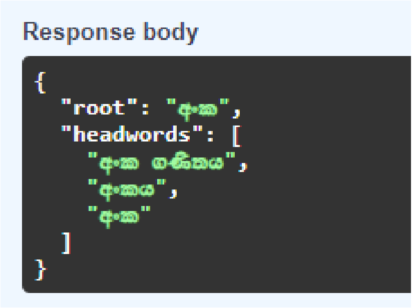
- මූලය 'දුව', දුවනවා ක්‍රියා පදයේ ස්වරූපයකි, අවසාන ක්‍රියාවකි, අතීත කාල, සාධනීය ක්‍රියා පදයකි

Identifying headwords with same roots

Main goal of extracting the morphological information from the headwords is to identify the headwords with the same root. A python script was written to store roots and respective headword lists as key,value pairs in a python dictionary.

```
roots = {}
with open('data/newmorph1.yaml', mode='r',encoding="utf-8") as fin:
    entries = yaml.load(fin)
    for doc in entries:
        headword = doc[0]
        morph = doc[1:]

        for m in morph:
            for i in m:
                root = i[0].split("+")[0]
                if root in roots:
                    roots[root].add(headword)
                else:
                    roots[root]=set([headword])
```



Response body

```
{
  "root": "අංක",
  "headwords": [
    "අංක ගණිතය",
    "අංකය",
    "අංක"
  ]
}
```

Figure : headwords with the same root

This python dictionary is later used to give suggestions for the user, when they search for a word the dictionary will suggest a few words that have the same root.

Embedding SinMorphy in the backend

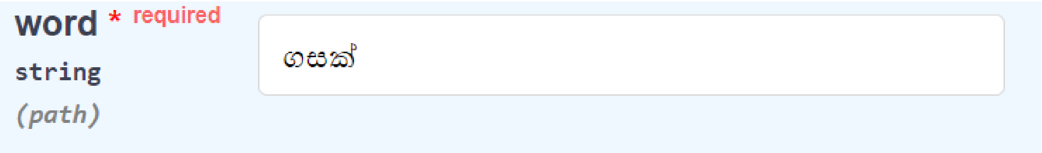
The SinMorphy morphological analyzer is embedded in the backend. When a user searches for a word, the word is passed in to the SinMorphy and the morphological information is extracted from it. Then the root of the search term is used to suggest other words with the same root.

SinMorphy Generator

When the user searches indefinite form of a word like “සබඳතාවක්”, the generator code is able to create the definite form of the word (සබඳතාව), and this form is searched in the dictionary dataset to show the definition. As of now this feature works for neuter nouns and verbs only.

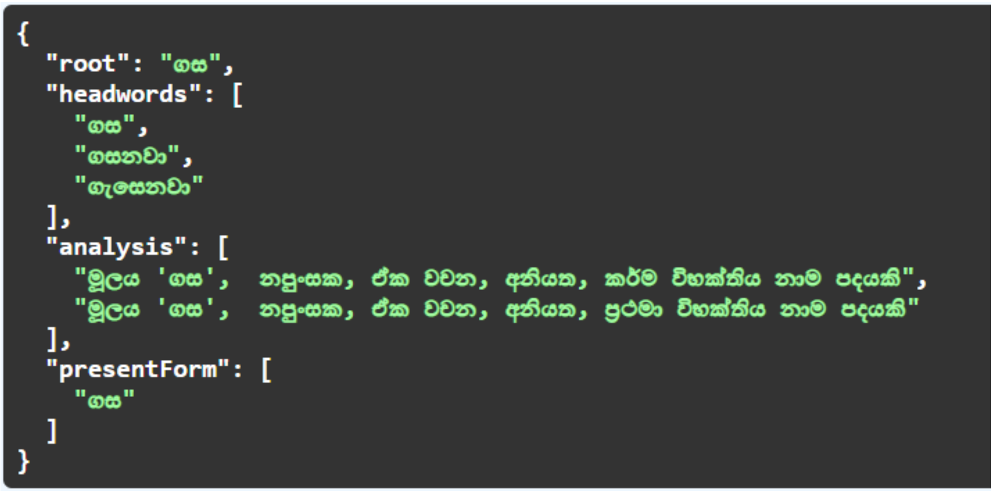
When a user searches a verb from any form except its present form, the SinMorphy generator generates its present form and searches the dictionary for that form.

Eg: if a user searches the word “ගසක්”. Since the dictionary might not have that indefinite form as a headword, it creates the definite form and searches for it.



word * required
string
(path)

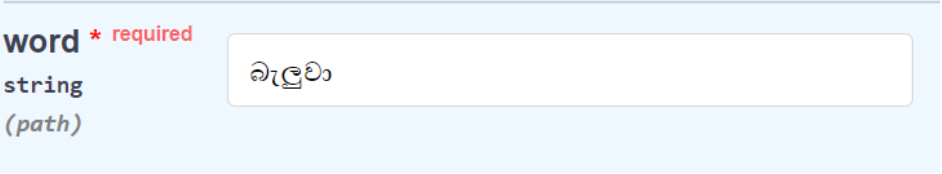
ගසක්



```
{
  "root": "ගස",
  "headwords": [
    "ගස",
    "ගසනවා",
    "ගැසෙනවා"
  ],
  "analysis": [
    "මූලය 'ගස',  නපුංසක, ඒක වචන, අනියත, කර්ම විභක්තිය නාම පදයකි",
    "මූලය 'ගස',  නපුංසක, ඒක වචන, අනියත, ප්‍රථමා විභක්තිය නාම පදයකි"
  ],
  "presentForm": [
    "ගස"
  ]
}
```

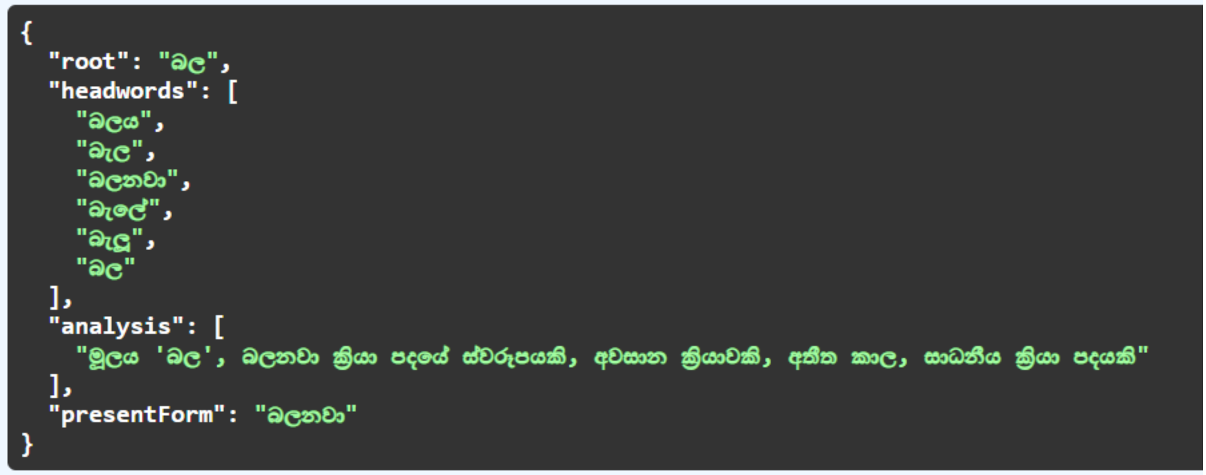
Figure : backend response for search "ගසක්"

Eg: if a user searches the word "බැලුවා". Since the dictionary might not have that past form of the verb as a headword, it creates the present form and searches for it.



word * required
string
(path)

බැලුවා



```
{
  "root": "බල",
  "headwords": [
    "බලය",
    "බැල",
    "බලනවා",
    "බැලේ",
    "බැලූ",
    "බල"
  ],
  "analysis": [
    "මූලය 'බල', බලනවා ක්‍රියා පදයේ ස්වරූපයකි, අවසාන ක්‍රියාවකි, අතීත කාල, සාධනීය ක්‍රියා පදයකි"
  ],
  "presentForm": "බලනවා"
}
```

Figure : backend response for search "බැලුවා"

Spell Correction

Using a spell correction tool developed by “SinSpell” research group, we are integrating the dictionary backend with a python program to correct the spell errors users make by typing the incorrect words.
The Sinspell team has developed this tool using three components.

1. error detection module
2. suggestion generation module
3. auto-correction module

Currently we are integrating our backend with error detection and suggestion generation modules.
Suggestions will be displayed to the users based on the suggestions given by the suggestion generation module.

Error Detection Module
This module is used to identify whether a user input word is correct or not. In the module,a custom created Sinhala dictionary based on Spylls is used. Spylls is a python ported version of Hunspell library.

Suggestion Generator Module
All the possible words for a given incorrect word are generated according to the following
criteria.

- Insert letter - Insert letters and in the alphabet to the wrong word and generate a possible word list.
- Delete letter - Delete the letters in the wrong word and generate a possible word list.
- Transposition letter - Transpose the two letters in the wrong word and generate a possible word list.
- Substitution letter - Replace the letters with other alphabetical letters and
- generate a possible word list.

Then the generated possible word list is looked up through the dictionary headword list and the words in that list are returned as the suggestions for
the given word.

Information Retrieval

Initial plan was to provide a partial word search in the head word list with n-gram indexing. So, users can search by typing any part of a head word. But, later decided to change search into a prefix search because search results look confusing.
Used Whoosh as the search engine for implementing head word search. Whoosh is a python library which provides the ability to build a search engine system without having a dedicated server unlike Elasticsearch.
For the fast search suggestions, we moved this headword list prefix search to the application frontend.

Results for search text: ගම

With Partial Word Search	With Prefix Search
අගමල අගම් ආගම අංගම ගම ගමකය ගමක ගමත් ගමන ගමනාගමනය	ගම ගමක ගමකය ගමකයා ගමකතා, ගමතීනයා ගමතුර ගමත් ගමත්සෙනස්න ගමඵය

Frontend Implementation

It was suggested that it’s effective to use the same framework for the mobile application and web application. Flutter was initially suggested as the framework to build the frontend since it supports Progressive Web Applications and Single Page Applications and brings existing mobile apps to the web.

But after more research we decided to use the Ionic framework since web application support in flutter is new and it is less customizable than Ionic.

Initial Design

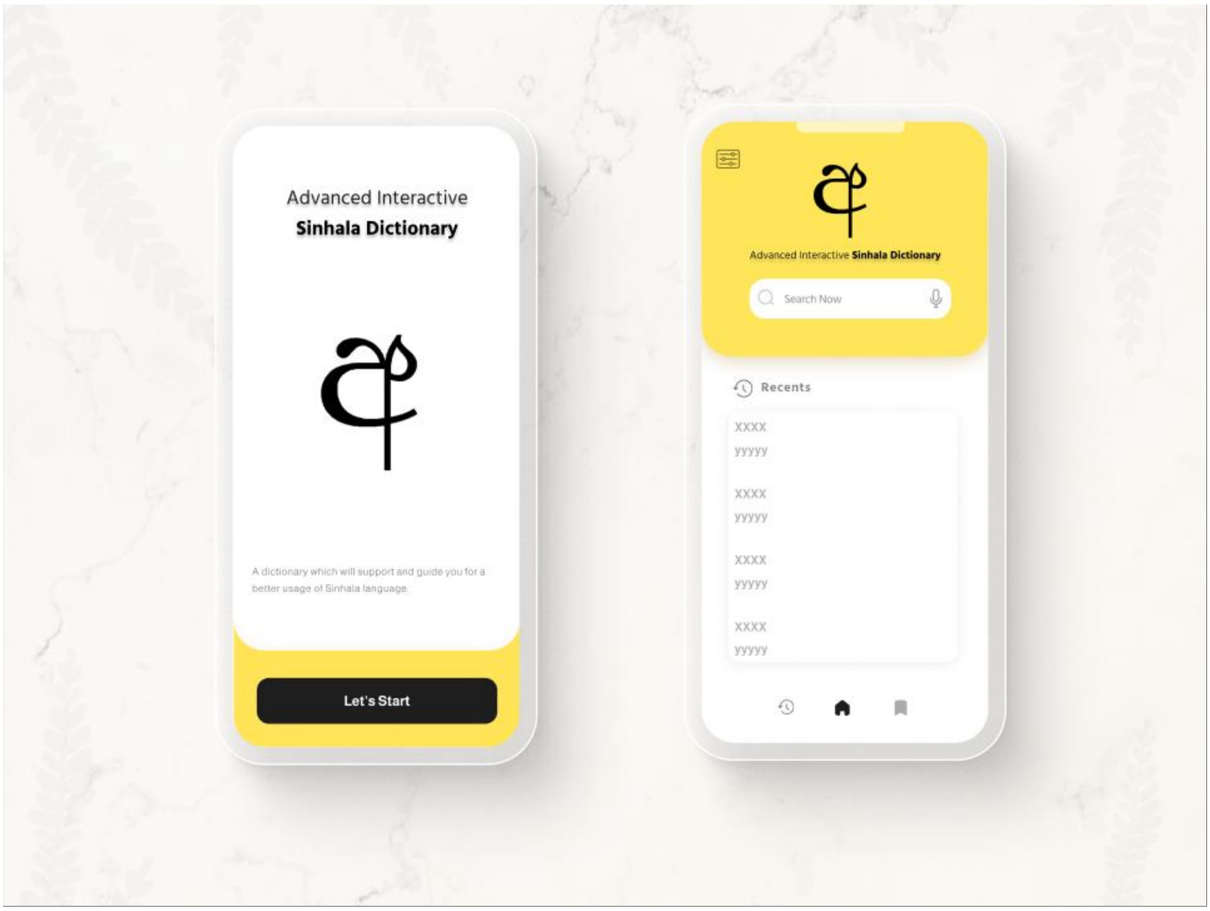

Advanced Interactive
Sinhala Dictionary
ද
A dictionary which will support and guide you for a better usage of Sinhala language
Let's Start
Advanced Interactive Sinhala Dictionary
Search Now
Recents
xxxx
yyyyy
xxxx
yyyyy
xxxx
yyyyy
xxxx
yyyyy

Figure : Mobile Application Frontend

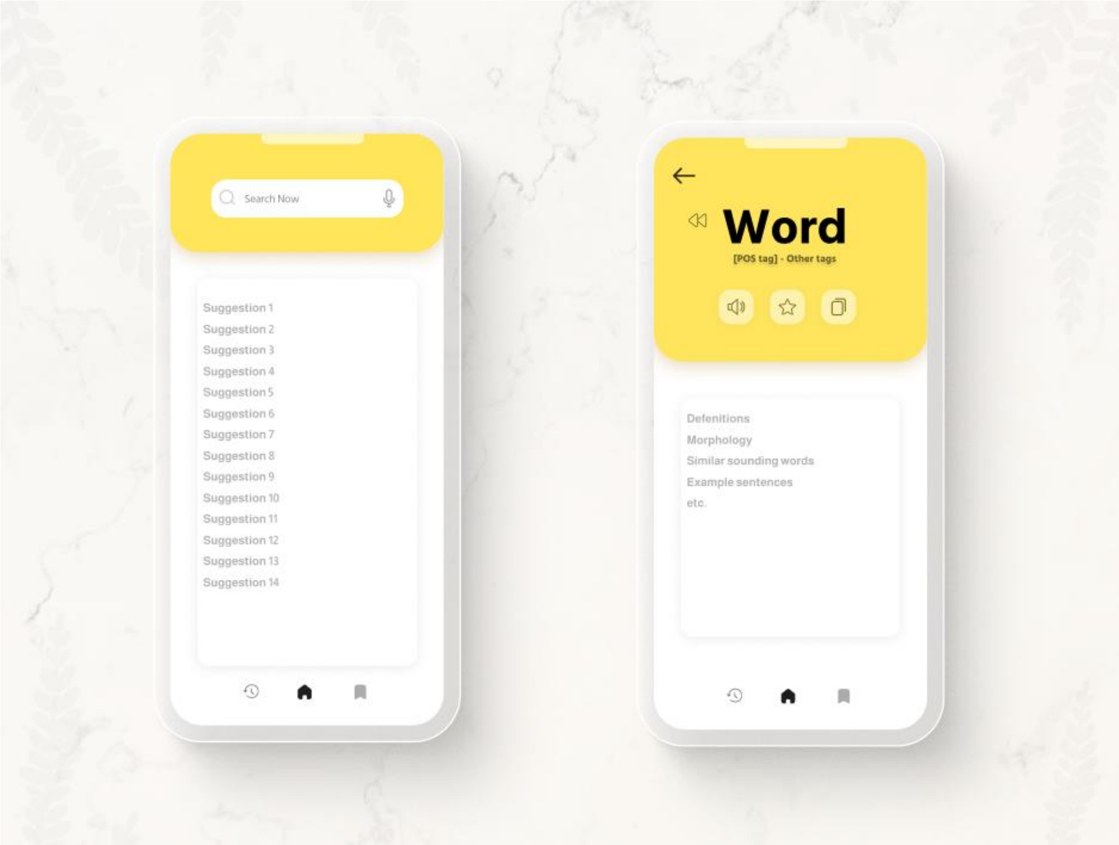


Figure : Mobile Application Frontend

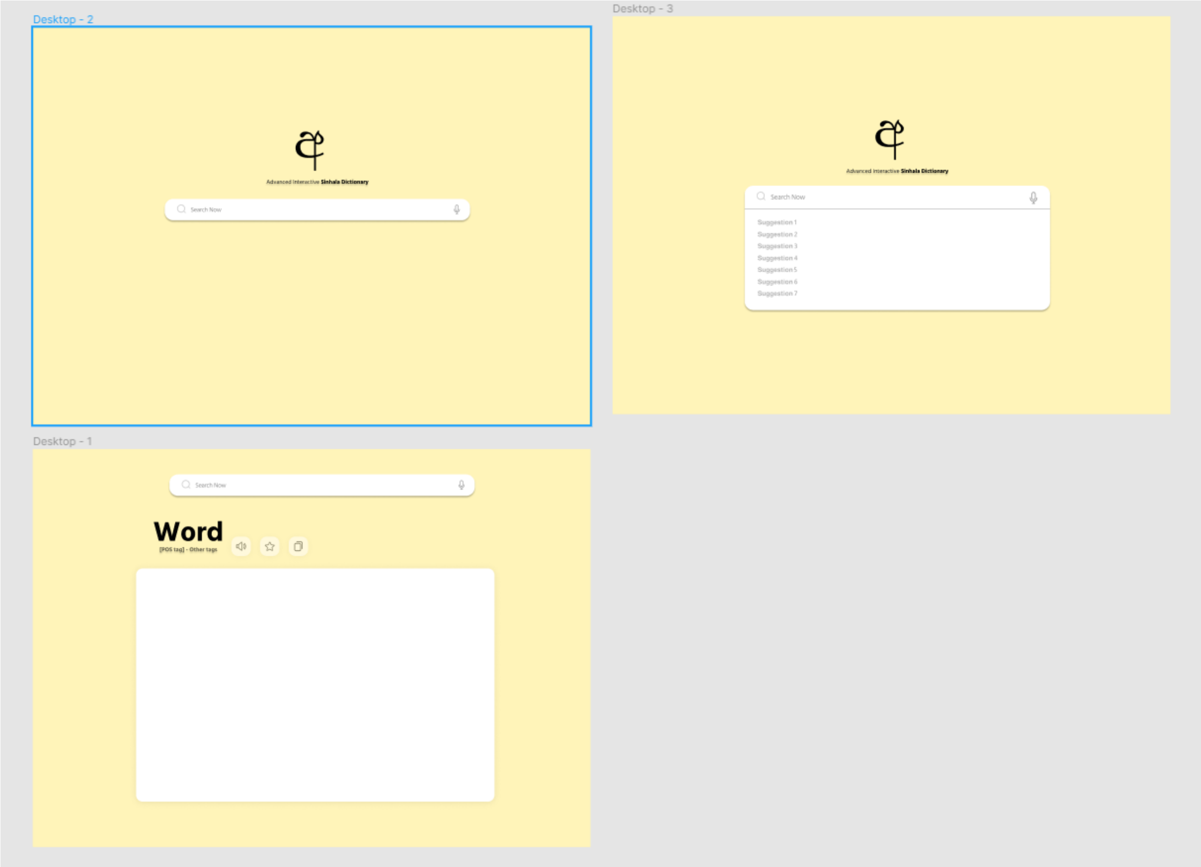
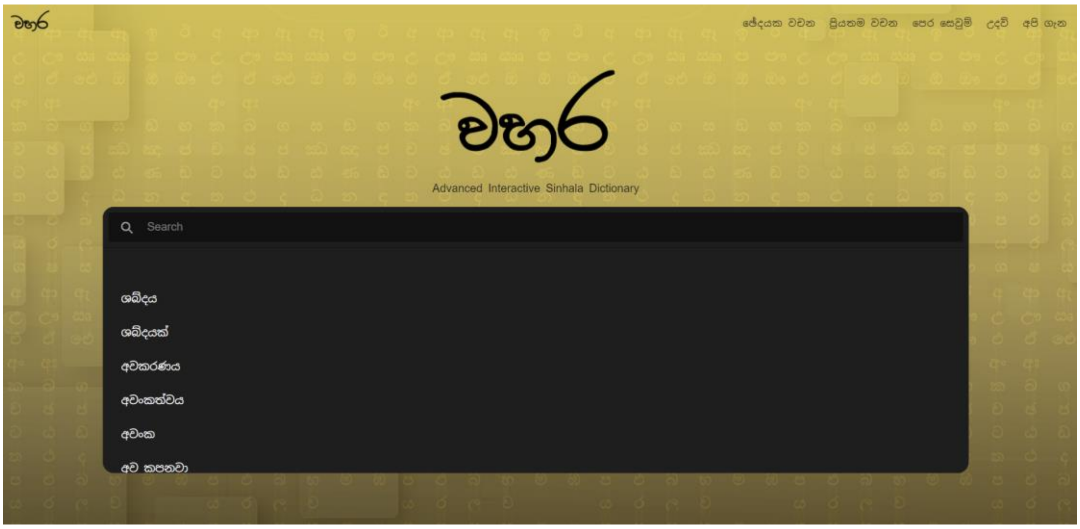


Figure : Web Application Frontend

Implemented Design



ශබ්දය

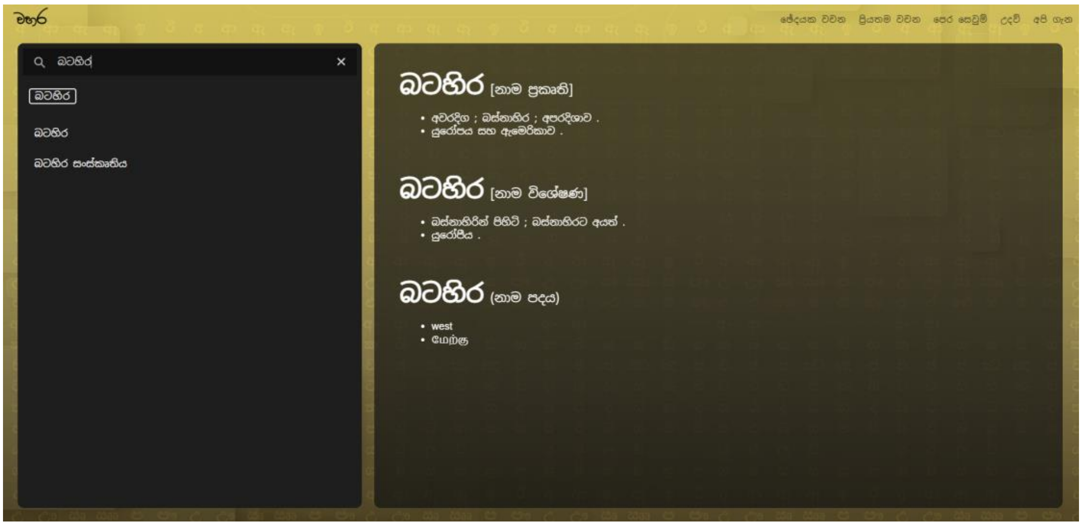
ශබ්දයක්

අවකරණය

අවංකත්වය

අවංක

අව කපනවා



බටහිර

බටහිර

බටහිර සංස්කෘතිය

බටහිර [නාම ප්‍රකෘති]

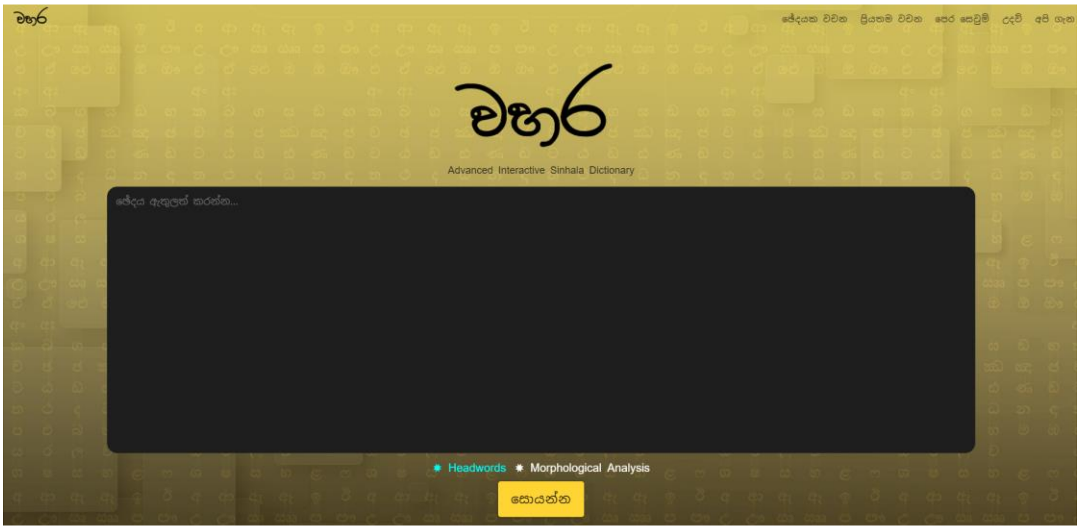
- අවරදිග ; බස්නාහිර ; අපරදිශාව .
- යුරෝපය සහ ඇමෙරිකාව .

බටහිර [නාම විශේෂණ]

- බස්නාහිරින් පිහිටි ; බස්නාහිරට අයත් .
- යුරෝපීය .

බටහිර (නාම පදය)

- west
- மேற்கு


වහර
ඡේදයක වචන
ප්‍රියතම වචන
පෙර සෙවුම්
උදව්
අපි ගැන
වහර
Advanced Interactive Sinhala Dictionary
ඡේදය ඇතුලත් කරන්න...
● Headwords ● Morphological Analysis
සොයන්න

Conclusion

There is a scarcity of online Sinhala language resources available. Therefore our project will help to mend it by providing an advanced Interactive Sinhala dictionary. This will be a product with high availability. So any user from the general public to a Sinhala language researcher can use it. We hope the tools we provide in this project will help people to study and research on Sinhala language more.

REFERENCES

[1]"Features in Dictionaries of Sinhala and Other Languages", *Google Docs*, 2021. [Online]. Available: https://docs.google.com/spreadsheets/d/1iKIEoQj6z3gCWsNPWRyZAyUAyEJBSc6Qcv5KJBjMZUo/edit#gid=0. [Accessed: 12- Aug- 2021].

[2]"Sinhala Dictionary Office", *Dictionary.gov.lk*, 2021. [Online]. Available: http://www.dictionary.gov.lk/index.php?lang=en. [Accessed: 12- Aug- 2021].

[3]"Cambridge Dictionary | English Dictionary, Translations & Thesaurus", *Dictionary.cambridge.org*, 2021. [Online]. Available: https://dictionary.cambridge.org. [Accessed: 12- Aug- 2021].

[4]"Derana Sinhala Dictionary & Glossary [සිංහල]", *Deranadictionary.com*, 2021. [Online]. Available: http://www.deranadictionary.com/. [Accessed: 12- Aug- 2021].

[5] Carolin Müller-Spitzer, Alexander Koplenig, Antje Töpel. "*What Makes a Good Online Dictionary? - Empirical Insights front an Interdisciplinary Research Project*". Institute for German Language (IDS), Mannheim, Nov. 2011.

[6] J. Tiedemann, "Improved Text Extraction from PDF Documents for Large-Scale Natural Language Processing," *Computational Linguistics and Intelligent Text Processing Lecture Notes in Computer Science*, pp. 102–112, 2014.

[7] T. Hassan and R. Baumgartner, "Intelligent Text Extraction from PDF Documents," *International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC06)*.

[8] E. Jayalatharachchi, A. Wasala, and R. Weerasinghe, "Data-driven spell checking: The synergy of two algorithms for spelling error detection and correction," *International Conference on Advances in ICT for Emerging Regions (ICTer2012)*, 2012.

[9] B. Hettige and A. S. Karunananda, "A Morphological Analyzer to Enable English to Sinhala Machine Translation," *2006 International Conference on Information and Automation*, 2006.

[10] D. Ward, J. Hahn, and K. Feist, "Autocomplete as Research Tool: A Study on Providing Search Suggestions," *Information Technology and Libraries*, vol. 31, no. 4, p. 6, 2012.

[11] K. Kumarasinghe, G. Dias and I. Herath, "SinMorphy: A Morphological Analyzer for the Sinhala Language," 2021 Moratuwa Engineering Research Conference (MERCon), 2021, pp. 681-686, doi: 10.1109/MERCon52712.2021.9525636.