ASD.AI: A TOOL TO SCREEN KIDS WITH AUTISM SPECTRUM DISORDER [2021-006]

Herath Mudiyanselage Danushika Nirmani Herath

(IT18081794)

The dissertation was submitted in partial fulfilment of the requirements for the B.Sc. Special Honors degree in Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology
Sri Lanka

October 2021

DECLARATION

I declare that this is my own work, and this proposal does not incorporate without acknowledgement of any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

IT18081794 Herath H.M.D.N.

The above candidate is carrying out research for the undergraduate Dissertation under my supervision.

Signature of the supervisor: Date:

Signature of the co-supervisor: Date:

ABSTRACT

The phrase *"Autism Spectrum Disorder"* refers to a collection of neurodevelopmental disorders marked by difficulties with social interaction, communication, and conduct. It is marked by difficulties in social communication as well as the occurrence of repetitive or strange actions. The prevalence of ASD has risen dramatically in recent years. An autism screening is used to identify the most prevalent early indicators of autism. From the moment they are born, all children begin to develop language. Learning and using language can be more difficult for autistic youngsters than for typically developing children. Autism awareness is increasing in Sri Lanka, although at a gradual pace and only because to the efforts of institutions, initiatives, and devoted individuals.

ASD knowledge and understanding in Sri Lanka is at an all-time low, and related research is insufficient. Nonetheless, according to a 2009 study, ASD affects one out of every 93 children aged 18 to 24 months. As a result, the alleged effort aids in the screening of ASD in Sri Lankan youngsters at a younger age in order to alleviate ASD-related issues. NLP is important in the proposed study since it helps to transform unstructured text data into a meaningful collection of information that can be used to test for ASD.

ACKNOWLEDGEMENT

I would want to offer my gratitude to my research group members in particular for their assistance. In addition, I would like to express my gratitude to our supervisor and co-supervisor for their counsel and assistance.

TABLE OF CONTENTS

DECLARATION i

ABSTRACT ii

ACKNOWLEDGEMENT iii

LIST OF FIGURES v

LIST OF TABLES v

LIST OF ABBREVIATIONS vi

INTRODUCTION 1

- Literature Review 1
- Research Gap 4
- Research Problem 7
- Research Objectives 10

METHODOLOGY 12

- Background 12
- NLP Techniques and Tools 14
- System implementation 18
- System diagram 18
- Individual system diagram 18
- Individual System Diagram 19
- Preprocessing 22
- Tokenization 22
- Stemming & Lemmatization 23
- Intent generation 24
- Vectorization 25
- Commercialization 27

RESULTS & DISCUSSION 28

CONCLUSION 32

REFERENCES 33

Appendices 36

- Plagiarism report screenshot 36

LIST OF FIGURES

		Page
Figure 1	Functional levels of ASD	2
Figure 2	ASD detection framework using structured and unstructured data	5
Figure 3	Frequency distribution for red flag signs	7
Figure 4	Frequency distribution for red flag signs	8
Figure 7	NLP approaches	14
Figure 8	System Overview Diagram	18
Figure 9	NLP architecture	19

LIST OF ABBREVIATIONS

Abbreviation	**Explanation**
APA	American Psychiatric Association
ASD	Autism Spectrum Disorder
ADHD	Attention Deficit Hyperactivity Disorder
OCD	Obsessive Compulsive Disorder
DSM-5	Diagnostic and Statistical Manual of Mental Disorders – 5th edition
NLP	Natural Language Processing
NLTK	Natural Language Toolkit

INTRODUCTION

Literature Review

According to documents of **American Psychiatric Association (APA) Autism Spectrum Disorder (ASD)** falls under a group of neurodevelopmental conditions which are denoted by social communication deficits, stereotyped interests, and repetitive behaviors [1]. On that account, ASD consists with an extensive range of symptoms which affects a kid's ability to communicate verbally as well as non-verbally.

ASD symptoms usually begins in early childhood. However, this may go unnoticed since patients show fewer physical disabilities. Principal ASD symptoms include social communication failures and restricted or repetitive behavior. Furthermore. ASD patients may not able to comprehend language, gestures, facial expressions, tone of voice and any other expression which aids in day to day communication. Similarly, they may find it difficult to express their emotions and/or understand emotions of other people. They also may feel overwhelmed by social interactions. Additionally, repetitive behaviors such as rocking; spinning; flapping one's body or surrounding objects, staring at spinning objects, following the same schedules; meal plans; clothing, showing ritualistic behaviors are some of the symptoms of these patients [2]. Moreover, [3] concludes repetitive behavior severity at the time of diagnosis stipulates a possibility of future anxiety disorders.

Nevertheless, ASD symptoms include some physical medical conditional behaviors not limiting to gastrointestinal issues (constipation, abdominal pain, acid reflux, bowel inflammation), seizure disorder, feeding problems, disrupted sleep, Attention Deficit Hyperactivity Disorder (ADHD), anxiety or depression disorders, Obsessive Compulsive Disorder (OCD), schizophrenia, bipolar disorder [4]. ASD when associated with these symptoms may worsen the situation of a toddler as it may require extensive screening techniques to diagnose ASD.

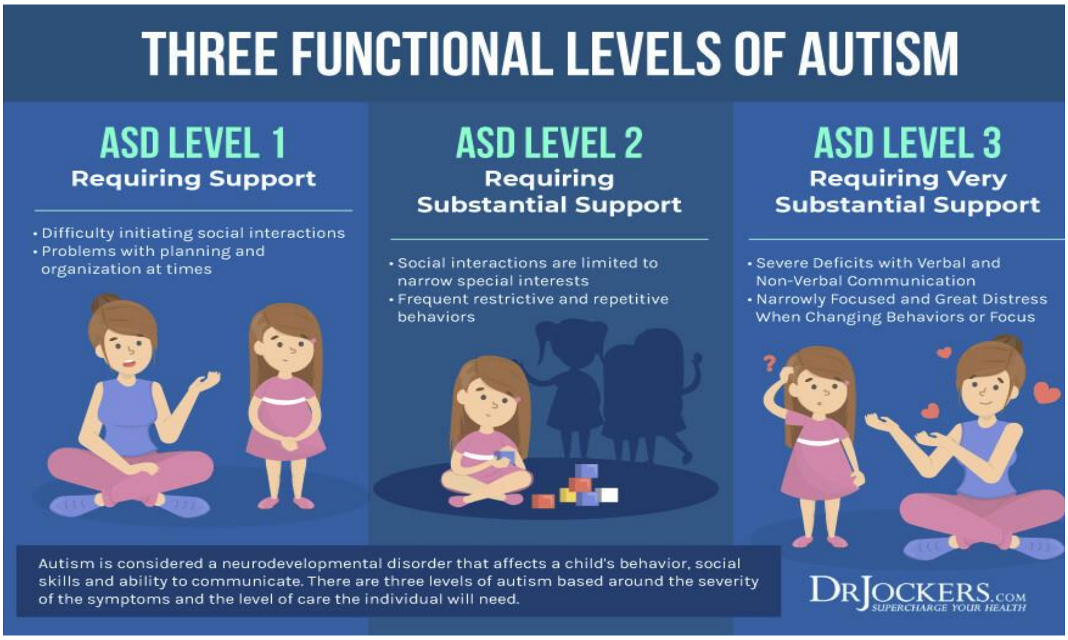

THREE FUNCTIONAL LEVELS OF AUTISM
ASD LEVEL 1
Requiring Support
• Difficulty initiating social interactions
• Problems with planning and organization at times
ASD LEVEL 2
Requiring Substantial Support
• Social interactions are limited to narrow special interests
• Frequent restrictive and repetitive behaviors
ASD LEVEL 3
Requiring Very Substantial Support
• Severe Deficits with Verbal and Non-Verbal Communication
• Narrowly Focused and Great Distress When Changing Behaviors or Focus
Autism is considered a neurodevelopmental disorder that affects a child's behavior, social skills and ability to communicate. There are three levels of autism based around the severity of the symptoms and the level of care the individual will need.
DRJOCKERS.COM
SUPERCHARGE YOUR HEALTH

Figure 1: Functional levels of ASD
Source: Complete Children's Health

As listed in Diagnostic and Statistical Manual of Mental Disorders – 5th edition (DSM-5) by APA, autistic patients fall under three categories based on their strengths and limitations of daily life activities and communicational interactions. These categories are named as functional levels and they indicate how much dependency will the patient demonstrate. Figure 1 shows these different levels and some of their symptoms. Level 1 kids lack the appropriate communication skills, adaptation ability and organizational skills and find it difficult to read social cues of other people. However, level 1 is the mildest among three levels. Level 2 kids manifest narrow interest and tend to engage in repetitive behaviors, requiring substantial support as compared to level 1 kids. Kids in level 3 indicates all behaviors of above categories but in an extreme manner therefore needing higher substantial support [5]. However, autistic patients may not be limited to one of these levels. They may express all the traits [6].

According to the result of a research which analyzed data from 2012 to 2019, the average age of diagnosis of ASD ranges between 43.18 months to 60.48 months. However, the researchers also emphasize that early detection of diagnosis aids massively in early treatment which eventually will lead to the improved quality of the person’s communication skills. Moreover, this research demonstrates that later ASD diagnosis leads to severe impairments, based on three groups of toddlers of ages ranging 12 to 18 months, 19 to 24 months and 25 to 41 months. The research accentuates the fact that ASD screening before the age of 18 months is efficient as this is the age where the symptoms are milder and amenable [7].

Research Gap

Accordingly, screening for ASD at an earlier age is a major necessity in present. This could be a difficult procedure as there are not any medical tests similar to blood tests. Therefore, ASD screening is done via procedures such as developmental monitoring, screening and evaluation [8].

Developmental monitoring consists of observing kids' growth over time and comparing the standard milestones with these growth behaviors. This can be done using a checklist of milestones by parents or by a thorough examination by a professional physician. Modified Checklist for Autism in Toddlers (M-CHAT) is a vastly used questionnaire for toddlers aged 16 to 30 months. However, [8] shows that the efficiency is at a lower level. But it could be modified to increase the accuracy. These monitoring techniques aid in determining the risk of ASD possibility.

Developmental screening includes more formal evaluation techniques especially by professional physicians. After primarily identification of the risk of ASD, these may be useful to confirm the condition. The procedure has to be done through clinical visits for a longer period. Social Responsiveness Scale (SRS) is a questionnaire which supports in finding an accurate history of the symptoms. Furthermore, for behavioral health assessments Diagnostic Interview for Social and Communication Disorders (DISCO) and Child Behavior Checklist are used. Autism Diagnostic Inventory-Revised (ADI-R) is another possible parent interviewing technique. Structured observation uses validated evaluation tools such as Childhood Autism Rating Scale - Second Edition (CARS-2) and Autism Diagnostic Observation Schedule - Second Edition (ADOS-2) which provide more accurate screening opportunity.

Developmental evaluation involves in reasonable measurement of cognitive ability, language skills and other behavioral conditions. Intelligent testing can either be primarily used for diagnosis or can be used to establish predeterminations. Formal evaluation of conversational language skills efficiently supports in diagnosis of ASD. Vineland Adaptive Behavior Scales and the Adaptive Behavior Assessment System are two common ways of measuring adaptivity behaviors for ASD diagnosis. Motor and sensory assessments including vision, hearing, and processing can be done for the diagnosis according to DSM-5 criteria.

In addition to aforesaid paper-based methods, there are several Artificial Intelligence (AI) based methods are presented by researches. A Machine Learning (ML) based framework with an 83.4% accuracy of test results [9] has been implemented. This model, however, has implemented with using text data from primarily diagnosed potential ASD children. An image of the proposed framework is shown in figure 2. Another [10] proposal shows an Artificial Neural Network (ANN) model with an accuracy of 100% with test data. ANN approach has been also taken into consideration by [11] with a higher accuracy as 95.79%. A Deep Learning (DL) model with using brain images as a dataset has been implemented with an accuracy of 70%. Another research [12] proposes a similar approach with using a Magnetic Resonance Images (MRI) scan dataset. Moreover, classification techniques have been used such as Random Forest (RF) and Support Vector Machine (SVM) with 91% and 70% test result accuracies respectively. Researchers also have suggested AI based platforms to evaluate motor movement [13] [14] in order to diagnose ASD with higher accuracies such as 82.5% [15].

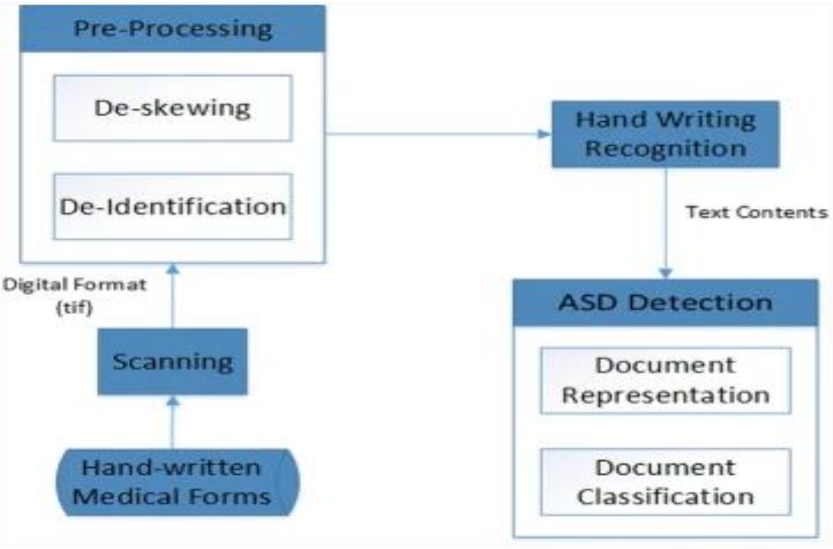


Figure 2: ASD detection framework using structured and unstructured data
Source: [9]

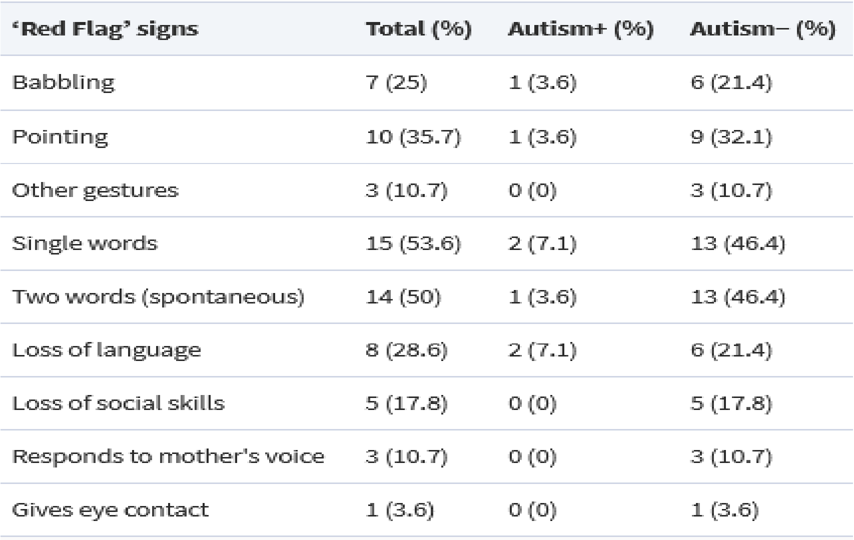
Natural Language Processing (NLP) is the approach where computers can understand and process human languages which has become another interesting research area for ASD diagnosis. NLP techniques usually aids in implementation of general purpose Chatbots. These chatbots have specifically enhanced for ASD diagnosis [16] with higher accuracy scores as 88% for age group of 1 to 3 years, 87.6% for age group of 4 to 6 years and 87.53% for age group of children above or equal to 7 years. Another application has been developed based on tracking eye movements of the patients and converting these data into text strings [17]so that data processing could be done via NLP techniques. Another NLP framework has taken into consideration the behavior and interaction of autistic patients with their siblings [18].

Consequently, these research findings suggest that AI approach is an efficient technique for ASD diagnosis as compared to traditional paper-based approaches. AI application aid in competent tool for screening of ASD.

Research Problem

As per the latest ASD prevalence findings, one in 54 children aged 8 years shows ASD traits in the US [19]. That is 1.85 as a percentage. However, in Sri Lanka research are scarce regarding ASD prevalence. Nevertheless, a 2009 research [20] which carried out ASD screening in Sri Lankan children aged 18 to 24 months shows ASD prevalence is a value of one in 93 children. This as a percentage is 1.07%.

The research has been carried out through selecting a sample of 374 children from a sub urban area of a population around 56,000. Methodology had included three levels. Level I has carried out via a 'Red Flag' criteria according to American Academy of Neurology and Child Neurology Society's parameters of ASD diagnosis. Accordingly, mothers of these sample children were contacted to explore if the children presents any of the above said red flags. Frequency diagram for failed red flag scenarios among 28 kids with and without ASD is shown in figure 3. Those who showed one or more rad flags were taken for level II screening. This has been done via a translated M-CHAT questionnaire to Sinhala language. The questionnaire had been filled again by those mothers. Finally, level III screening has been done via clinical assessment by a psychiatrist by a home visit and at clinic diagnosis. Final diagnosis has been then confirmed by a hospital based clinical evaluation.

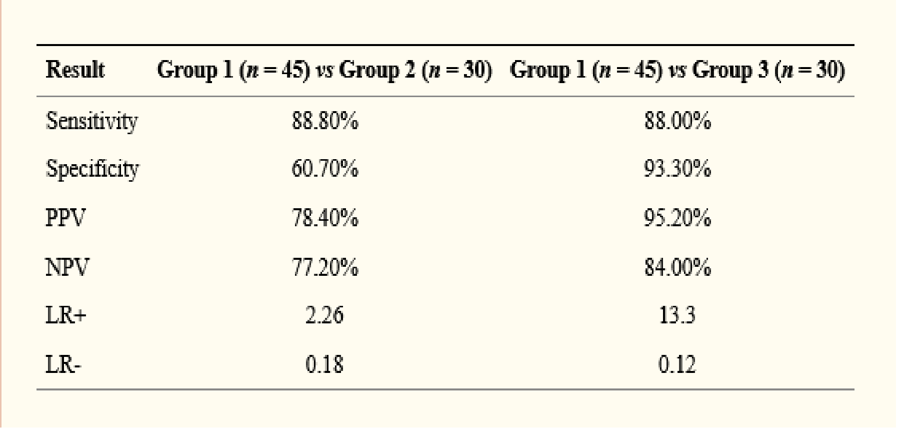


'Red Flag' signs	Total (%)	Autism+ (%)	Autism− (%)
Babbling	7 (25)	1 (3.6)	6 (21.4)
Pointing	10 (35.7)	1 (3.6)	9 (32.1)
Other gestures	3 (10.7)	0 (0)	3 (10.7)
Single words	15 (53.6)	2 (7.1)	13 (46.4)
Two words (spontaneous)	14 (50)	1 (3.6)	13 (46.4)
Loss of language	8 (28.6)	2 (7.1)	6 (21.4)
Loss of social skills	5 (17.8)	0 (0)	5 (17.8)
Responds to mother's voice	3 (10.7)	0 (0)	3 (10.7)
Gives eye contact	1 (3.6)	0 (0)	1 (3.6)

Figure 3: Frequency distribution for red flag signs
Source: [20]

However, the researchers further discusses that the sensitivity and the validity of the M-CHAT tool is as low as 25% and this could have been affected due to cultural and social preferences. Therefore, the additional evaluations of red flags criteria and clinical assessment have done for the increased efficiency. Furthermore, the research emphasizes the fact that ASD prevalence is alarmingly high and the need for culturally sensitive specific screening tools for ASD diagnosis in Sri Lanka.

Another interesting 2017 research [21] presents an extended version of [20] regarding this matter by bringing forward a culturally adapted ASD screening tool via a pictorial approach. A group of 105 kids were grouped into 3 groups and screened via a culturally adapted screening checklist. The checklist has been translated to Sinhala and Tamil languages based on DSM- 5 and M-CHAT tools but not directly converted. Moreover, the customized too l has been evaluated by language experts in order to check the appropriateness and accuracy. Later the entries in the checklist have been matched with an illustration to emphasize the meaning of the entry and the pictures used are form local kids. The completed instrument is named as Pictorial Autism Assessment Schedule (PAAS). Figure 4 shows the performance data of the PAAS. Accordingly, the PAAS has demonstrated a higher accuracy level of 88.8%.



Result	Group 1 (*n* = 45) *vs* Group 2 (*n* = 30)	Group 1 (*n* = 45) *vs* Group 3 (*n* = 30)
Sensitivity	88.80%	88.00%
Specificity	60.70%	93.30%
PPV	78.40%	95.20%
NPV	77.20%	84.00%
LR+	2.26	13.3
LR-	0.18	0.12

Figure 4: Frequency distribution for red flag signs
Source: [21]

Another latest research [22] presents a software package for ASD screening. This aids in differentiating neurotypical behaviors form ASD behaviors. The application is based on maturity level, intelligent level and eye contact behaviors of children. Thus, the package provides an eye movement tracking tool which records eye movements of participants and analyze the data via Symmetric Mass Center Algorithm and results will be shown graphically. Intelligence level evaluation is done via an interactive color and number-based set of activities where the gathered response data has been used to analyze the ASD condition. Moreover, the tool comprises a maturity level analyzer which provides the opportunity for the participants draw a said shape and the drawings have been analyzed by Standard Deviation of all acquired data.

On the other hand, [23] accentuates the fact of absence of standard ASD screening tools in Sri Lanka. In order to carry out a globally standardized Autism Diagnostic Observation Schedule (ADOS-2) screening physicians have to be certified and they ought to have standardized materials. It is a difficulty in Sri Lanka as both these requirements are very costly.

After carefully considering all the above research findings, it is obvious that there is not enough screening support tools in Sri Lanka which are easy to operate and there is a limitation in gathering required prevalence data due to difficulties is screening capacity. Hence, there is a need for a screening tool with a higher efficiency and ease of use. Moreover, the screening tool should be customized to Sinhala language support. Additionally, use of an AI based approach is suggested over a traditional paper-based screening technique.

Research Objectives

The suggested solution's main goal is to develop a machine learning-based automated autism screening tool that will reduce or eliminate the need for inefficient and error-prone human involvement in the field. The ability of the proposed system to support English and Sinhala. The system's effective and reliable operation will have a direct impact on service quality.

Furthermore, the suggested platform's intelligent agents can work 24 hours a day, 365 days a year to achieve their set goals. Humans are emotional beings, and their current mental state can have a direct impact on the level of service they deliver. The intelligent agent, on the other hand, has a zero likelihood of this happening.

Furthermore, the suggested platform would deploy intelligent agents based on the current load of requests to be handled, and the agents will be able to conduct many conversations at the same time, as opposed to their human counterparts.

In addition, the system is simple to configure to fit a variety of purposes. When compared to present systems, intelligent agents will have a low to unknown maintenance cost once implemented. Because of the modular framework on which it was created, the system will be straightforward to adapt to multiple languages.

Furthermore, intelligent agents will be reactive and efficient once deployed using the suggested platform. Users of the system can gain productivity, time, and scalability thanks to the intelligent agent's capacity to communicate with several users at once.

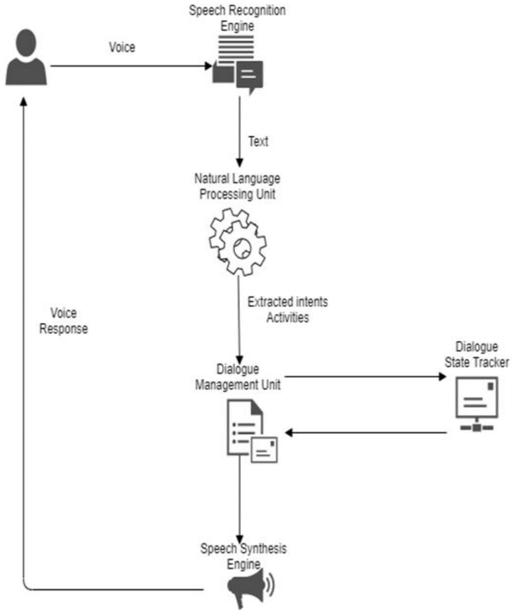
Furthermore, the solution should develop a customized NLU tool with Sinhala language support as a component of a Machine Learning based automated autism screening tool that supports both English and Sinhala languages in order to reduce or eliminate error-prone, inefficient human intervention, for efficient and robust performance, to increase availability, for simultaneous user access, which is cost effective, and to improve overall quality and productivity of the service.

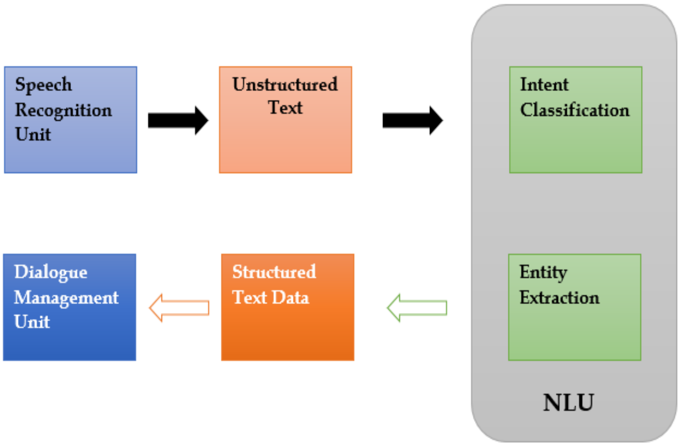
METHODOLOGY

Background

As per the objectives ASD.AI is the solution we propose as a complete ASD screening tool. The system includes four basic components. They are the components of *speech recognition*, *Natural Language Understanding (NLU)*, *dialogue management and speech synthesis.*

However, in this instrument implementation of NLP component is further discussed.





NLP is the process of interaction of computers and human languages which includes inputting natural human language data, processing and outputting structured data. NLP has a history running to 1950s however, with the evolution of ML algorithms NLP has improved vastly in late 80s to present. At present NLP has more evolved with the use of DL and neural networks. NLP has been used in variety of fields such as medical, business and finance, entertainment, security, education etc.

NLP deals with comprehending unstructured data in text format which is known as information extraction. Then the data can be used for sentiment analysis which is an application of NLP. Sentiment analysis is the process of making decisions whether positive or negative aspects based on text review data. Language conversion tools is another real-life application of NLP technique.

In present time, NLP is being used efficiently in applications such as spam detection, Part-of-Speech PoS tagging, and Named Entity Recognition (NER). Moreover, NLP has shown higher accuracies in applications of information extraction, sentiment analysis, Word Sense Disambiguation (WSD), parsing and machine translation. Paraphrasing, summarization, answering questions, dialog management, however, are few fields which needs more enhancements. However, NLP faces constraints when the input text has ambiguities, nonstandard usage of languages, idioms, neologisms, punctuation, etc. especially in Asian languages.

NLP Techniques and Tools

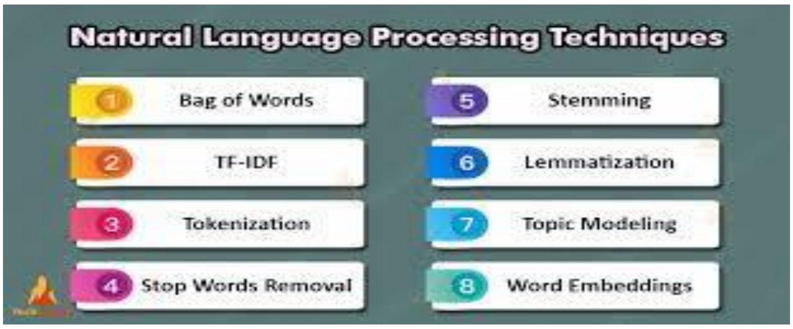


Figure 7: NLP approaches

A bag of words (BoW) is a text modeling technique used in word processing. From a technical standpoint, it is a method of extracting attributes from textual data. This method is a straightforward and adaptable method for extracting characteristics from documents. A bag of words is a textual representation that describes how words appear in a document. We only keep track of the word count and ignore grammatical details and word order. Because all information about the order or structure of words in the document is removed, this is referred to as a "bag" of words. The model only considers whether known words appear in the document, not whether the document is true. To easily implement the BoW model in Python, the Sk-learn library's CountVectorizer() function can be used.

The scoring method described above counts each word and arranges the words in a vector based on the count of that particular word. If a word appears multiple times in a document, as well as many other documents in this dataset, it is possible that the word is just a regular word, not because it is relevant or meaningful. One method is to match the frequency of words with their frequency in all documents, so that many common words like "the" that are also common in all documents are penalized. This method is known as the TF-IDF (Inverse Document Frequency Frequency Notation).

The abbreviation TF-IDF is meant to reflect the term's importance in this document. By multiplying two different metrics, the TF-IDF of a word in a document is calculated. The frequency with which a word appears in a document (TF). The raw number of occurrences of a word in a document is the simplest way to calculate this frequency. There are additional methods for adjusting the frequency.

Tokenization is a common task in natural language processing (NLP). This is a key step in both traditional NLP techniques like the Count Vectorizer and advanced deep learning architectures like Transformers. Tokenization is the process of dividing a text into smaller units known as tokens. Tokens may be words, symbols, or subwords in this context. As a result, tokenization can be classified as one of three types: tokenization of words, symbols, and subwords. If a space is used as a separator, it is the most common way to create tokens. Because each token is a word, it exemplifies Word tokenization. Symbols and subwords can also be tokens.

The tokenizer divides unstructured data and natural language text into information pieces that can be treated independently. Token instances in a document can be used directly to represent that document as a vector. Sentence tokenization refers to the process of breaking down text into sentences. This directly converts the unstructured string into a machine-learning-friendly digital data structure. They can also be used by the computer to initiate useful actions and reactions. Alternatively, they can be used as functions in a machine learning pipeline to drive more complex decisions or behaviors.

Stopwords are the most frequently occurring words in all-natural languages. These stop words may not add much value to the meaning of the document when analyzing text data and building NLP models. Prepositions are the most commonly used textual elements in English. For example, in the sentence "This is a new book," the words "is" and "a" add no meaning to the statement. Words like "this," "new," and "book," on the other hand, are keywords that tell us what the script is looking at.

Text normalization is the process of converting a word into a single canonical form. This is accomplished through two processes: stemming and lemmatization. Stemming is the task of shortening a word to its root, which is used for suffixes, prefixes, and term roots. More results are obtained by recognizing, searching for, and extracting more word forms. Once the wordform is recognized, you can return previously lost search results. Constraints are an essential part of information search and retrieval because they provide additional information. Stemming is a subset of linguistic research in AI morphology and information retrieval. Because additional aspects of the topic word will need to be learned, sentiment and artificial intelligence evaluation draw vital information from vast sources such as big data or the Internet.

When a new word is discovered, it can offer up new study possibilities. Using the basic morphological form of the word: lemma, you can often get the best results. The inference is done by a human or by an algorithm that can be employed by an AI system to find the stem. Mood employs a variety of techniques to reduce a word to its stem, regardless of declension. Creating an inference algorithm is not difficult. Some simple algorithms just remove prefixes and suffixes that are identified. These basic algorithms, however, are frequently incorrect. Such algorithms may also have difficulty with phrases like saw and vision, which have inflectional forms that may not properly express the stem.

Over stemming is the act of removing far more from a word than is necessary, resulting in two or more words being reduced to the same root word or poor root when they should not have been. At least two. Words that are radical. Some root algorithms can track two words all the way back to the root universe, implying that they both mean the same thing, which is obviously not the case. Furthermore, under stemming is the inverse case. When two or more words are inadequately stemmed, they may be erroneously reduced to more than one root word, when they must be reduced to the same root. Both must be reduced to the same root, although optimizing such models might introduce bias.

The project is developed with the Natural Language Toolkit (NLTK) which is a Python package for natural language processing that is open source. It includes a suite of text editing libraries for classification, tokenization, rooting, and markup, as well as user-friendly interfaces for over 50 fraternities and lexical resources like WordNet. It's easy to tokenize sentences and words in a text with the NLTK tokenization module. Many fantastic ways for conducting various phases of data preprocessing are available in the NLTK toolkit. For radicalization and lemmatization, there are methods like "PorterStemmer" and "WordNetLemmatizer," respectively.

A supervised learning model is provided by NLTK. In addition, several additional languages make it easy to customize. It also protects datasets from prying eyes. It also allows you to use pre-trained models to fit certain datasets. In addition, many intents can be handled in a single message. It also has out-of-the-box model testing features, allowing it to improve accuracy over time.

System implementation

System diagram

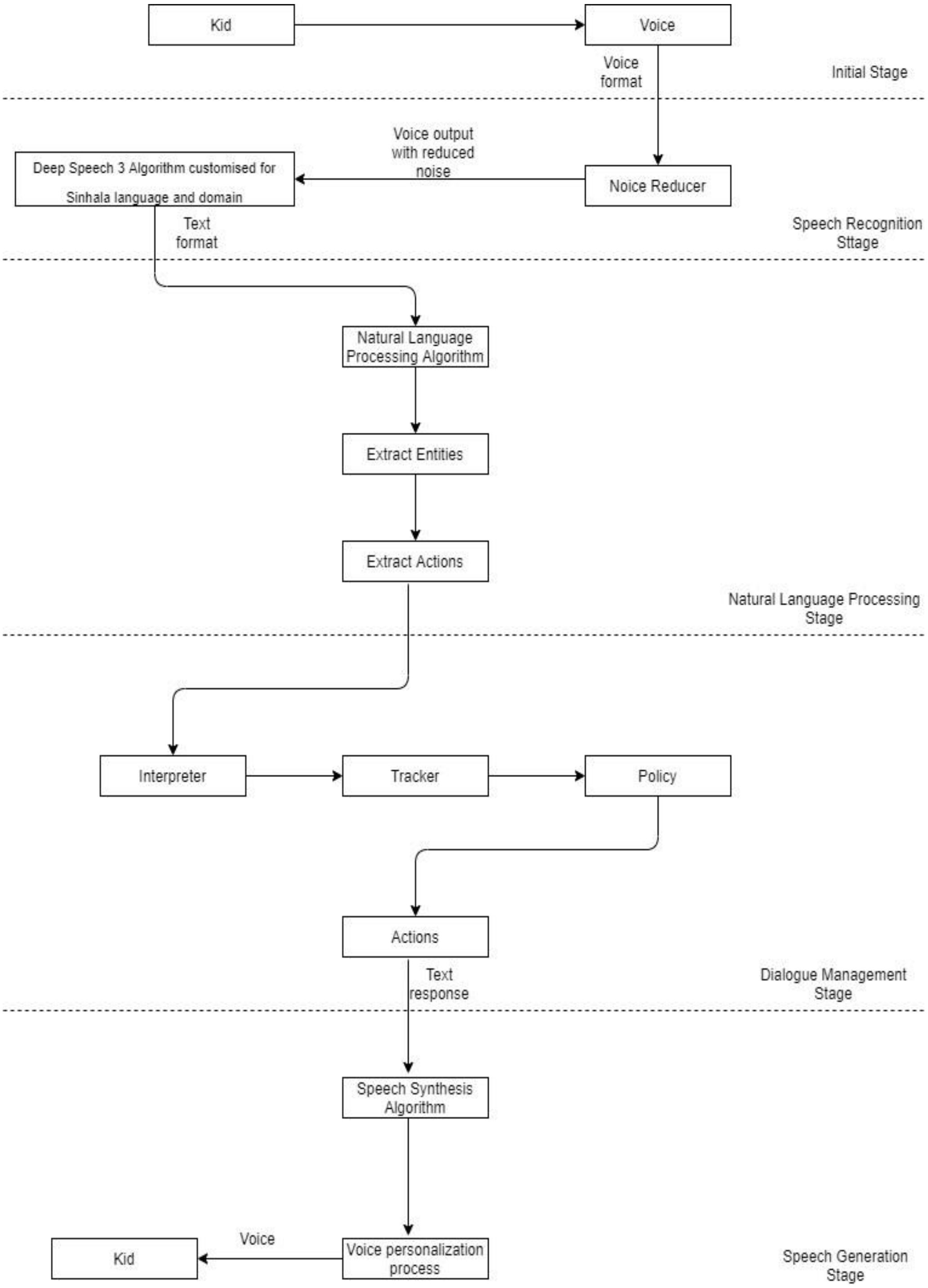

Kid
Voice
Voice format
Initial Stage
Voice output with reduced noise
Deep Speech 3 Algorithm customised for Sinhala language and domain
Noice Reducer
Text format
Speech Recognition Sttage
Natural Language Processing Algorithm
Extract Entities
Extract Actions
Natural Language Processing Stage
Interpreter
Tracker
Policy
Actions
Text response
Dialogue Management Stage
Speech Synthesis Algorithm
Voice
Kid
Voice personalization process
Speech Generation Stage

Figure 8: System Overview Diagram

Individual System Diagram

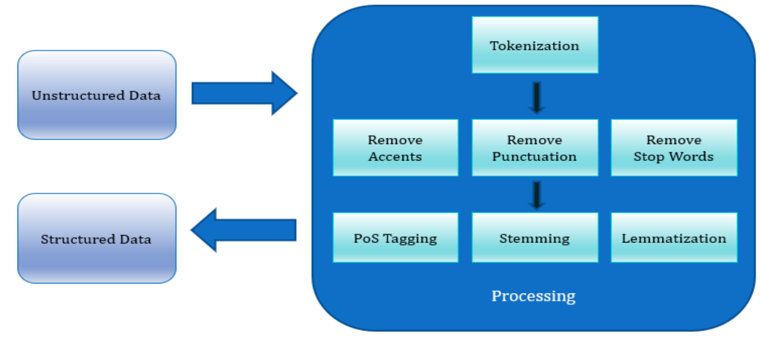


Figure 9: NLP architecture

Collecting enough text was a critical component of the project, and it proved difficult for a variety of reasons. To begin with, the amount of Sinhala textual content available on the internet is limited, and the content is inconsistent because different websites use different text encodings and fonts. This problem was solved by gathering articles from newspaper website archives and encoding them with the LTRL Unicode character encoding tool. The second issue was that many NLP tools only support ASCII encoding, whereas Sinhala text is encoded in Unicode. This was overcome by preprocessing the text to correspond to each of the algorithms. The tests section shows the preprocessing steps for each test. For the sake of simplicity, most non-Sinhala characters were removed during preprocessing.

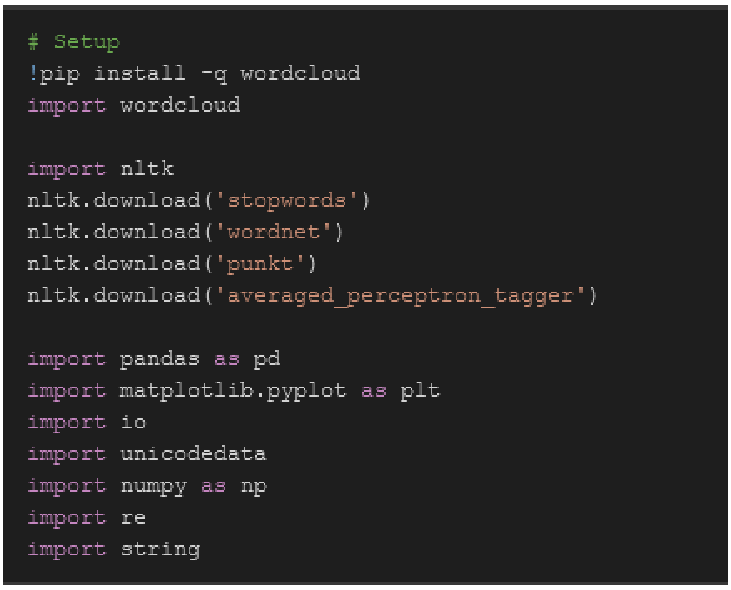
There are various types of preprocessing that can be used to improve the way we model with words. The first of these is "lemmatization." A word's "lemma" is its basic form. "is," for example, is a lemma for the word "is." As a result, when lemmatize the word "is," it becomes "is." Ignored words are frequently deleted as well. Stop words are words that appear frequently in the language but provide little information. In English, stop words contain linguistic data that contains a lot of noise mixed in with informational content. The key words in the preceding sentence are tea, which is both healthy and soothing. By removing stop words, the predictive model is able to focus on relevant words.

Matching tokens or phrases in chunks of text or entire documents is another common NLP task. Regular expressions can be used to perform pattern matching, but the NLTK matching capabilities are generally easier to use. The matcher is built with the model's vocabulary. Lowercase sentences are matched by the definition of attr = 'LOWER.' This guarantees a case-insensitive match. Stemming works in the same way, combining multiple forms of the same word into a single basic form. However, stemming and skip word removal can reduce the success of the models. As a result, consider this pre-processing to be part of the hyperparameter optimization process. Then a list of terms in the text is made. As document objects, the sentence finder requires models. The simplest way to accomplish this is to use the NLP model to comprehend a list.

Hot coding is the most basic and widely used representation. Each document is represented as a vector of term frequency for each term in the vocabulary. The dictionary is made up of all corpus files. Count the number of times the term appears in each document and assign that score to the appropriate vector element. The first sentence contains several words repeated twice, and because this is the first position in this dictionary, we put the number 2 in the first element of the vector. The vector proposals look like this. This is referred to as a set of words. Documents with similar terms will have similar vectors, as   can see. Because dictionaries typically contain tens of thousands of terms, these vectors can be quite large.

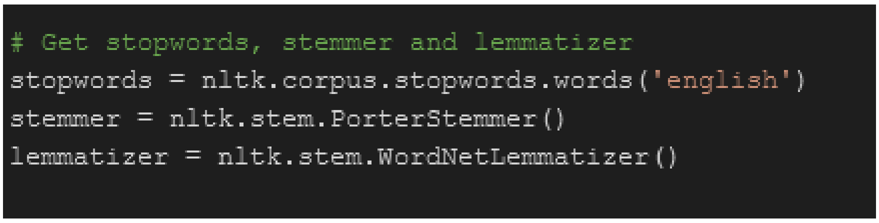
Word embeddings numerically represent each word, resulting in a vector that corresponds to how the word is used or what it means. The context in which the words occur is taken into account when studying vector coding. There are numerous methods for combining all of the word vectors into a single document vector that can be used to train the model. Simply averaging the vectors for each word in the document is a simple and surprisingly effective approach. Then you can model with these document vectors. Words found in similar contexts have vectors that are similar. Furthermore, the relationship between words can be investigated using mathematical operations. Subtraction of vectors produces another vector.

Preprocessing

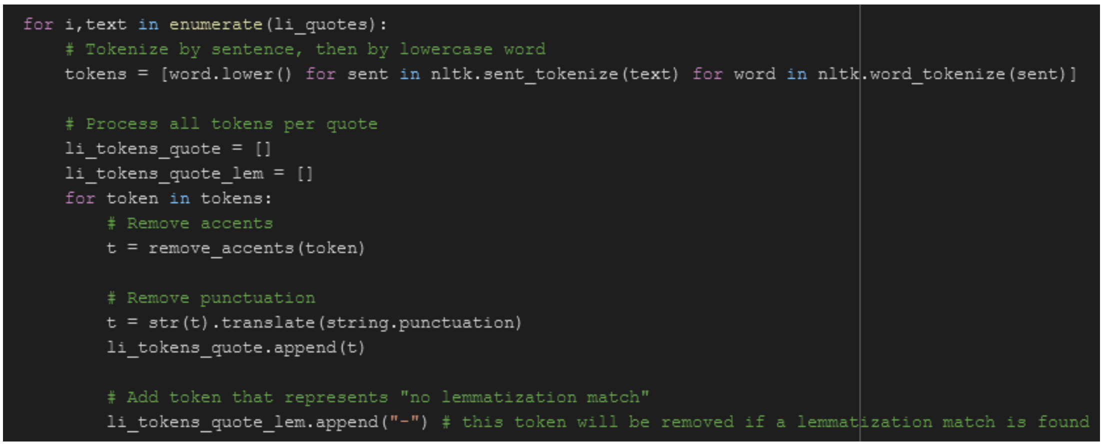


```
# Setup
!pip install -q wordcloud
import wordcloud

import nltk
nltk.download('stopwords')
nltk.download('wordnet')
nltk.download('punkt')
nltk.download('averaged_perceptron_tagger')

import pandas as pd
import matplotlib.pyplot as plt
import io
import unicodedata
import numpy as np
import re
import string
```

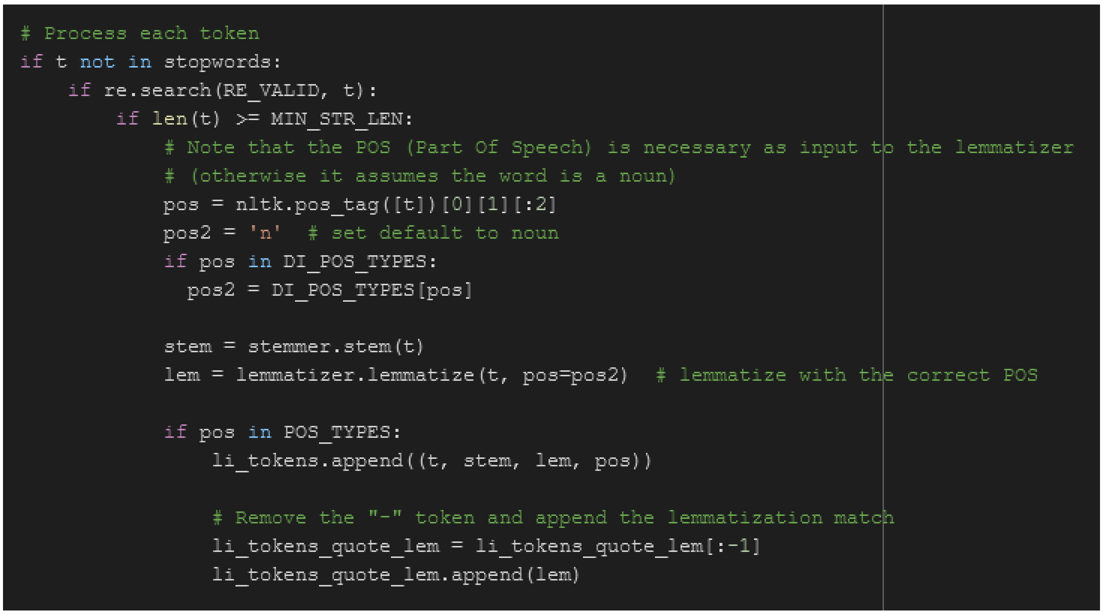
Tokenization



```
# Get stopwords, stemmer and lemmatizer
stopwords = nltk.corpus.stopwords.words('english')
stemmer = nltk.stem.PorterStemmer()
lemmatizer = nltk.stem.WordNetLemmatizer()
```

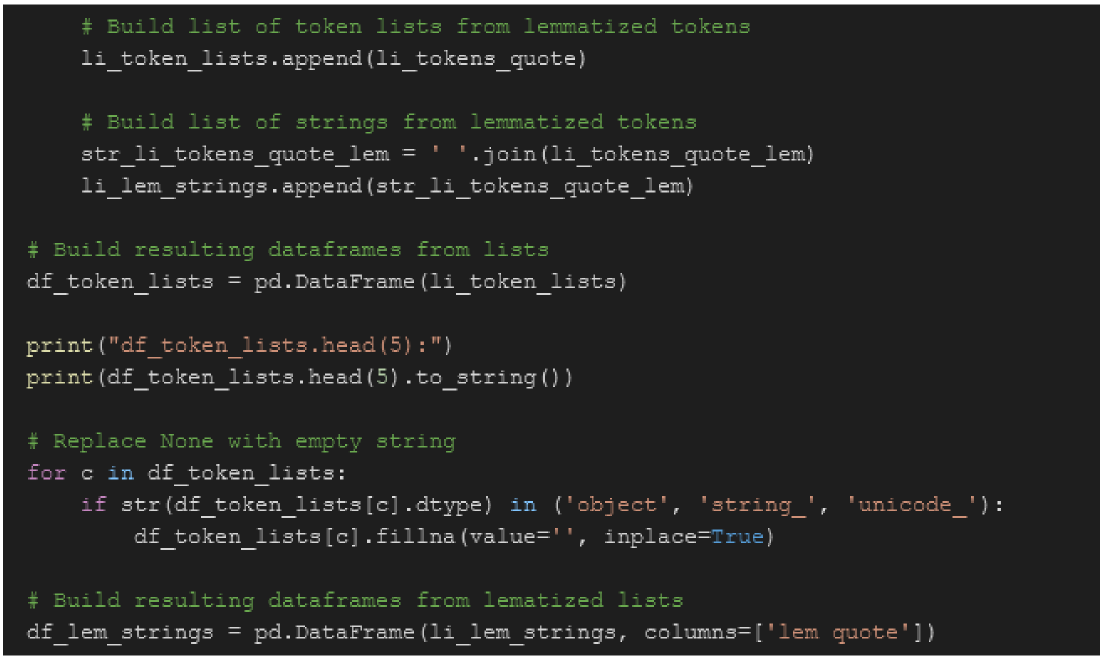


```
for i,text in enumerate(li_quotes):
    # Tokenize by sentence, then by lowercase word
    tokens = [word.lower() for sent in nltk.sent_tokenize(text) for word in nltk.word_tokenize(sent)]

    # Process all tokens per quote
    li_tokens_quote = []
    li_tokens_quote_lem = []
    for token in tokens:
        # Remove accents
        t = remove_accents(token)

        # Remove punctuation
        t = str(t).translate(string.punctuation)
        li_tokens_quote.append(t)

        # Add token that represents "no lemmatization match"
        li_tokens_quote_lem.append("-") # this token will be removed if a lemmatization match is found
```



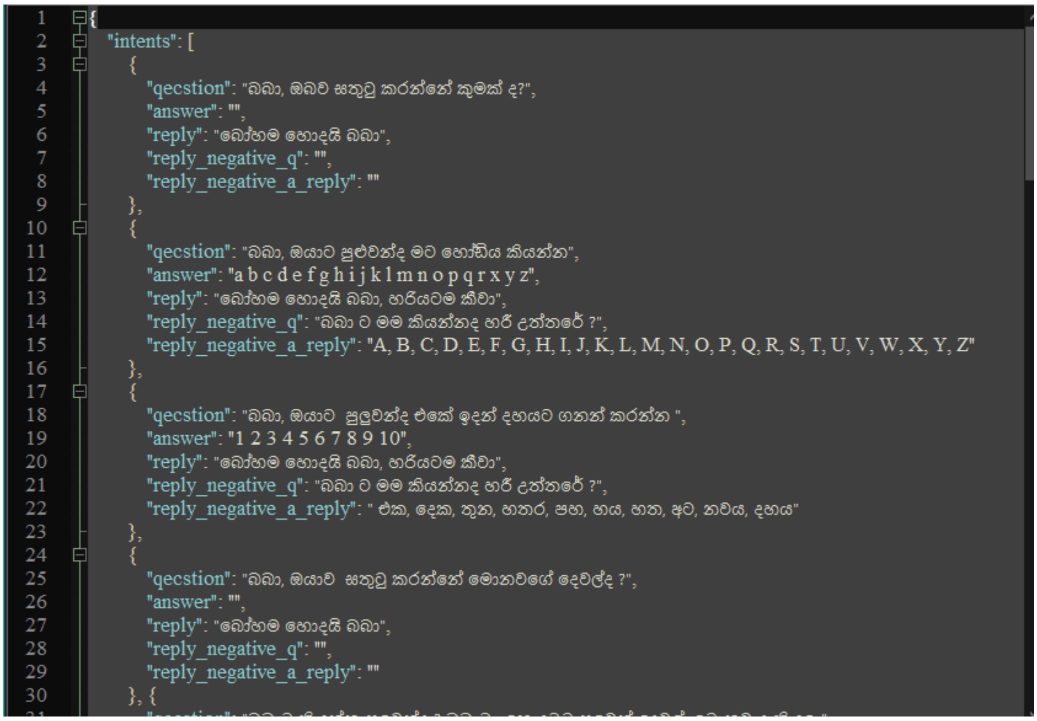
```
# Process each token
if t not in stopwords:
    if re.search(RE_VALID, t):
        if len(t) >= MIN_STR_LEN:
            # Note that the POS (Part Of Speech) is necessary as input to the lemmatizer
            # (otherwise it assumes the word is a noun)
            pos = nltk.pos_tag([t])[0][1][:2]
            pos2 = 'n'  # set default to noun
            if pos in DI_POS_TYPES:
              pos2 = DI_POS_TYPES[pos]

            stem = stemmer.stem(t)
            lem = lemmatizer.lemmatize(t, pos=pos2)  # lemmatize with the correct POS

            if pos in POS_TYPES:
                li_tokens.append((t, stem, lem, pos))

                # Remove the "-" token and append the lemmatization match
                li_tokens_quote_lem = li_tokens_quote_lem[:-1]
                li_tokens_quote_lem.append(lem)
```

Stemming & Lemmatization

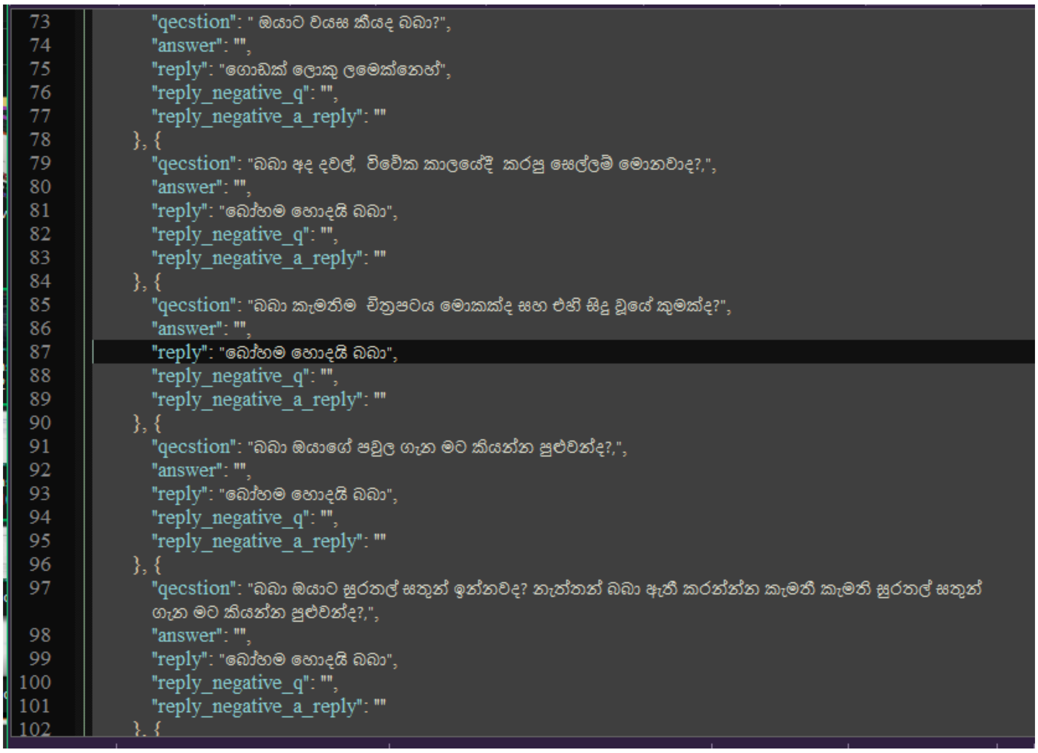


```
    # Build list of token lists from lemmatized tokens
    li_token_lists.append(li_tokens_quote)

    # Build list of strings from lemmatized tokens
    str_li_tokens_quote_lem = ' '.join(li_tokens_quote_lem)
    li_lem_strings.append(str_li_tokens_quote_lem)

# Build resulting dataframes from lists
df_token_lists = pd.DataFrame(li_token_lists)

print("df_token_lists.head(5):")
print(df_token_lists.head(5).to_string())

# Replace None with empty string
for c in df_token_lists:
    if str(df_token_lists[c].dtype) in ('object', 'string_', 'unicode_'):
        df_token_lists[c].fillna(value='', inplace=True)

# Build resulting dataframes from lematized lists
df_lem_strings = pd.DataFrame(li_lem_strings, columns=['lem quote'])
```

Intent generation

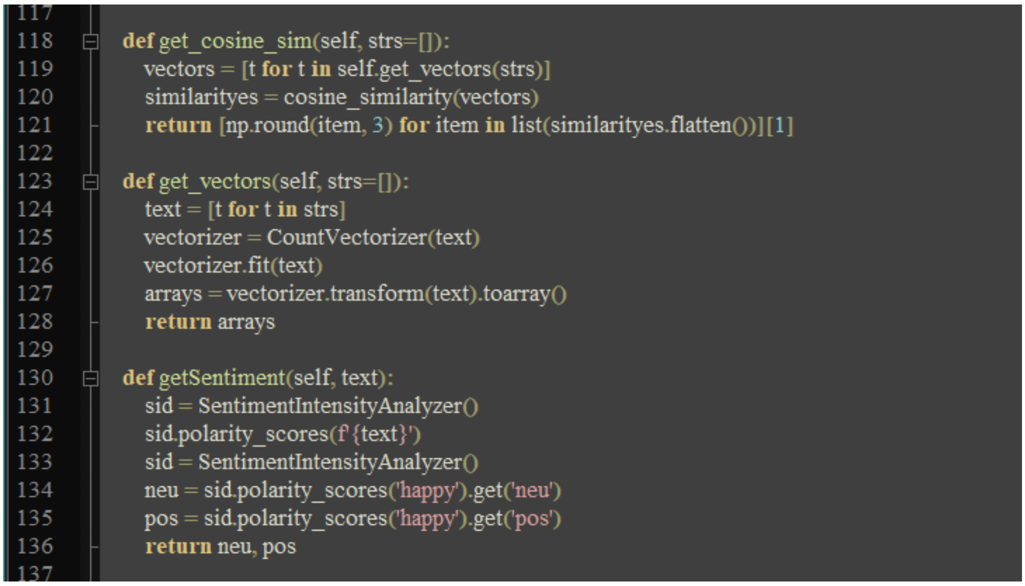


```
{
  "intents": [
    {
      "qecstion": "බබා, ඔබව සතුටු කරන්නේ කුමක් ද?",
      "answer": "",
      "reply": "බෝහම හොදයි බබා",
      "reply_negative_q": "",
      "reply_negative_a_reply": ""
    },
    {
      "qecstion": "බබා, ඔයාට පුළුවන්ද මට හෝඩිය කියන්න",
      "answer": "a b c d e f g h i j k l m n o p q r x y z",
      "reply": "බෝහම හොදයි බබා, හරියටම කීවා",
      "reply_negative_q": "බබා ට මම කියන්නද හරි උත්තරේ ?",
      "reply_negative_a_reply": "A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z"
    },
    {
      "qecstion": "බබා, ඔයාට  පුලුවන්ද එකේ ඉදන් දහයට ගනන් කරන්න ",
      "answer": "1 2 3 4 5 6 7 8 9 10",
      "reply": "බෝහම හොදයි බබා, හරියටම කීවා",
      "reply_negative_q": "බබා ට මම කියන්නද හරි උත්තරේ ?",
      "reply_negative_a_reply": " එක, දෙක, තුන, හතර, පහ, හය, හත, අට, නවය, දහය"
    },
    {
      "qecstion": "බබා, ඔයාව  සතුටු කරන්නේ මොනවගේ දෙවල්ද ?",
      "answer": "",
      "reply": "බෝහම හොදයි බබා",
      "reply_negative_q": "",
      "reply_negative_a_reply": ""
    }, {
```

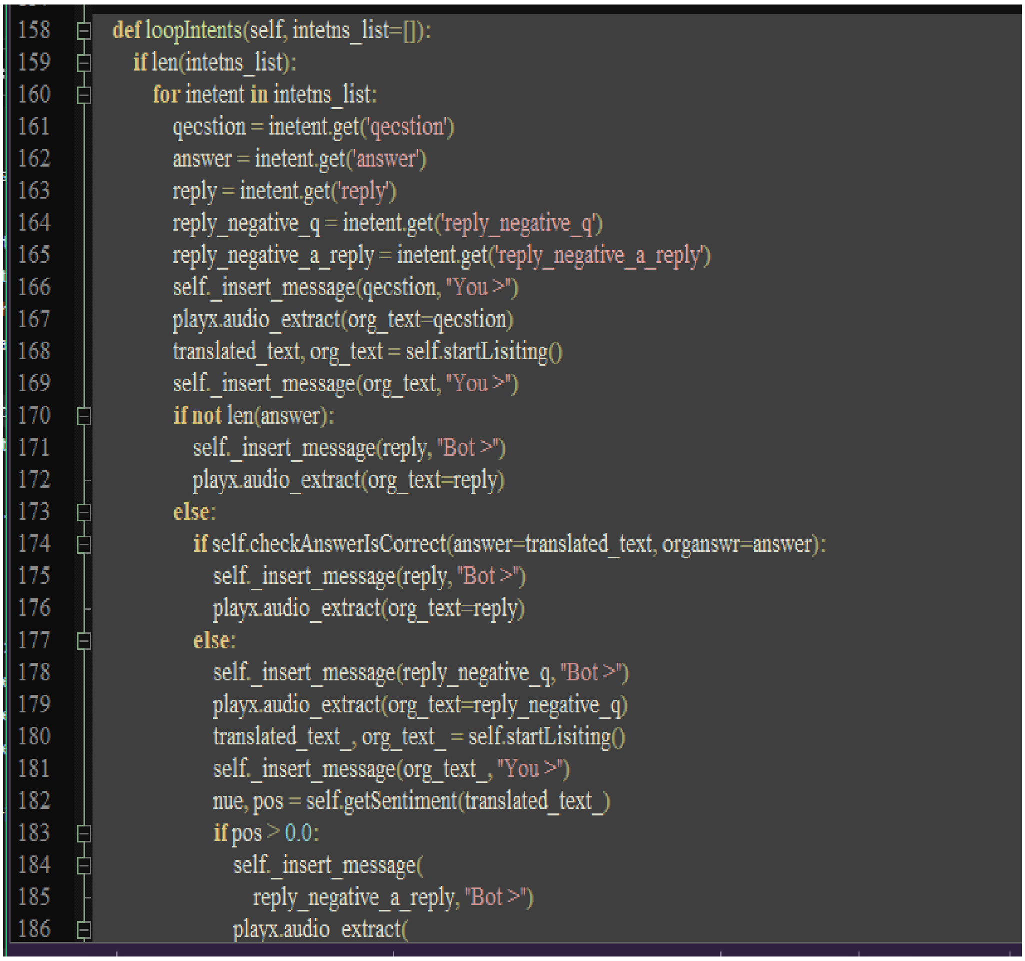


```
        "qecstion": " ඔයාට වයස කීයද බබා?",
        "answer": "",
        "reply": "ගොඩක් ලොකු ලමෙක්නෙහ්",
        "reply_negative_q": "",
        "reply_negative_a_reply": ""
    }, {
        "qecstion": "බබා අද දවල්,  විවේක කාලයේදී  කරපු සෙල්ලම් මොනවාද?,",
        "answer": "",
        "reply": "බෝ්හම හොදයි බබා",
        "reply_negative_q": "",
        "reply_negative_a_reply": ""
    }, {
        "qecstion": "බබා කැමතිම  චිත්‍රපටය මොකක්ද සහ එහි සිදු වූයේ කුමක්ද?",
        "answer": "",
        "reply": "බෝ්හම හොදයි බබා",
        "reply_negative_q": "",
        "reply_negative_a_reply": ""
    }, {
        "qecstion": "බබා ඔයාගේ පවුල ගැන මට කියන්න පුළුවන්ද?,",
        "answer": "",
        "reply": "බෝ්හම හොදයි බබා",
        "reply_negative_q": "",
        "reply_negative_a_reply": ""
    }, {
        "qecstion": "බබා ඔයාට සුරතල් සතුන් ඉන්නවද? නැත්තන් බබා ඇතී කරන්න්න කැමතී කැමති සුරතල් සතුන්
        ගැන මට කියන්න පුළුවන්ද?,",
        "answer": "",
        "reply": "බෝ්හම හොදයි බබා",
        "reply_negative_q": "",
        "reply_negative_a_reply": ""
    }, {
```

Vectorization



```
    def get_cosine_sim(self, strs=[]):
        vectors = [t for t in self.get_vectors(strs)]
        similarityes = cosine_similarity(vectors)
        return [np.round(item, 3) for item in list(similarityes.flatten())][1]

    def get_vectors(self, strs=[]):
        text = [t for t in strs]
        vectorizer = CountVectorizer(text)
        vectorizer.fit(text)
        arrays = vectorizer.transform(text).toarray()
        return arrays

    def getSentiment(self, text):
        sid = SentimentIntensityAnalyzer()
        sid.polarity_scores(f'{text}')
        sid = SentimentIntensityAnalyzer()
        neu = sid.polarity_scores('happy').get('neu')
        pos = sid.polarity_scores('happy').get('pos')
        return neu, pos
```



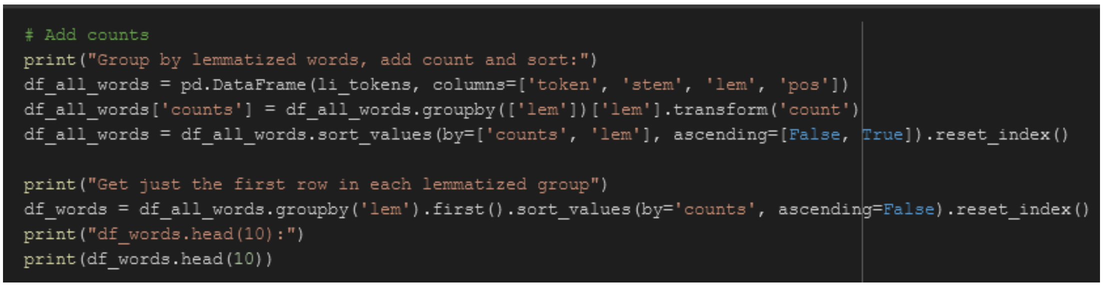
```
def loopIntents(self, intetns_list=[]):
    if len(intetns_list):
        for inetent in intetns_list:
            qecstion = inetent.get('qecstion')
            answer = inetent.get('answer')
            reply = inetent.get('reply')
            reply_negative_q = inetent.get('reply_negative_q')
            reply_negative_a_reply = inetent.get('reply_negative_a_reply')
            self._insert_message(qecstion, "You >")
            playx.audio_extract(org_text=qecstion)
            translated_text, org_text = self.startLisiting()
            self._insert_message(org_text, "You >")
            if not len(answer):
                self._insert_message(reply, "Bot >")
                playx.audio_extract(org_text=reply)
            else:
                if self.checkAnswerIsCorrect(answer=translated_text, organswr=answer):
                    self._insert_message(reply, "Bot >")
                    playx.audio_extract(org_text=reply)
                else:
                    self._insert_message(reply_negative_q, "Bot >")
                    playx.audio_extract(org_text=reply_negative_q)
                    translated_text_, org_text_ = self.startLisiting()
                    self._insert_message(org_text_, "You >")
                    nue, pos = self.getSentiment(translated_text_)
                    if pos > 0.0:
                        self._insert_message(
                            reply_negative_a_reply, "Bot >")
                        playx.audio_extract(
```

Commercialization

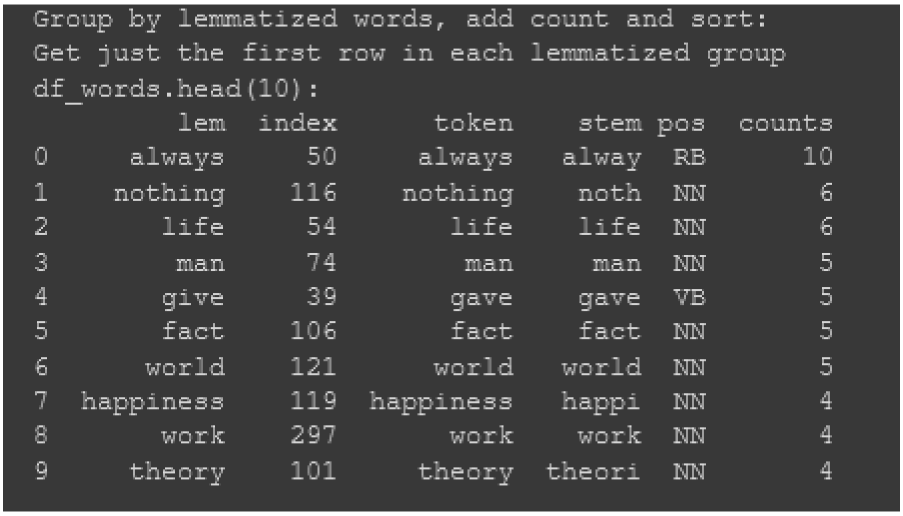
The system can be commercialized due to the innovative nature of the system and ease of customizability. Moreover, it is easier to customized with the exiting technologies. Also, the system is user friendly and easy to maintain and learn.

-

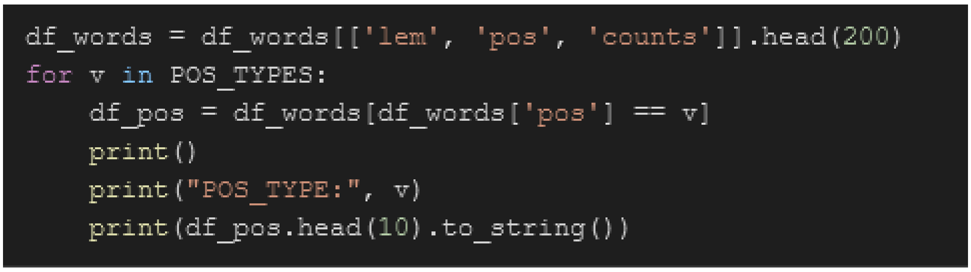
RESULTS & DISCUSSION



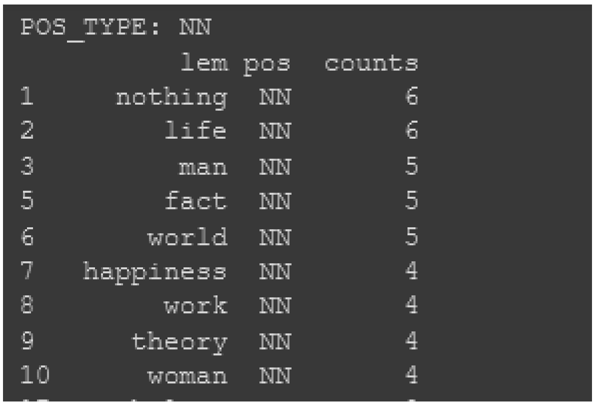
```
# Add counts
print("Group by lemmatized words, add count and sort:")
df_all_words = pd.DataFrame(li_tokens, columns=['token', 'stem', 'lem', 'pos'])
df_all_words['counts'] = df_all_words.groupby(['lem'])['lem'].transform('count')
df_all_words = df_all_words.sort_values(by=['counts', 'lem'], ascending=[False, True]).reset_index()

print("Get just the first row in each lemmatized group")
df_words = df_all_words.groupby('lem').first().sort_values(by='counts', ascending=False).reset_index()
print("df_words.head(10):")
print(df_words.head(10))
```



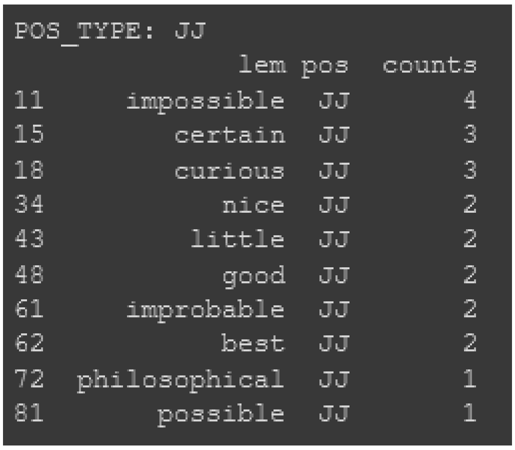
```
Group by lemmatized words, add count and sort:
Get just the first row in each lemmatized group
df_words.head(10):
         lem  index      token    stem pos  counts
0     always     50     always  alway  RB      10
1    nothing    116    nothing   noth  NN       6
2       life     54       life   life  NN       6
3        man     74        man    man  NN       5
4       give     39       gave   gave  VB       5
5       fact    106       fact   fact  NN       5
6      world    121      world  world  NN       5
7  happiness    119  happiness  happi  NN       4
8       work    297       work   work  NN       4
9     theory    101     theory theori  NN       4
```



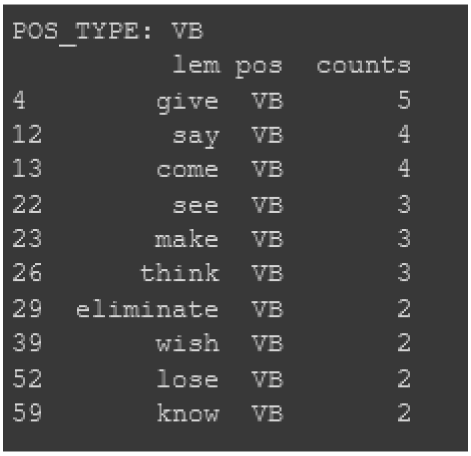
```
df_words = df_words[['lem', 'pos', 'counts']].head(200)
for v in POS_TYPES:
    df_pos = df_words[df_words['pos'] == v]
    print()
    print("POS_TYPE:", v)
    print(df_pos.head(10).to_string())
```



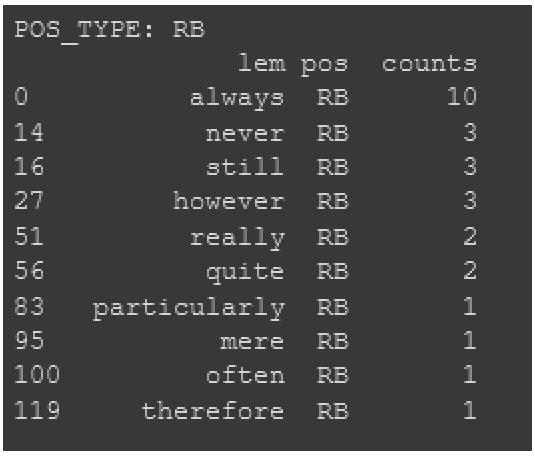
```
POS_TYPE: NN
          lem pos  counts
1     nothing  NN       6
2        life  NN       6
3         man  NN       5
5        fact  NN       5
6       world  NN       5
7   happiness  NN       4
8        work  NN       4
9      theory  NN       4
10      woman  NN       4
```



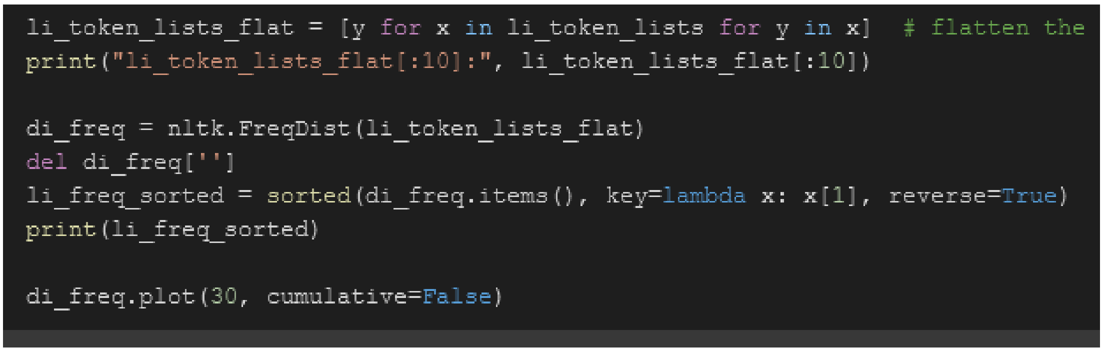
```
POS_TYPE: JJ
              lem pos  counts
11     impossible  JJ       4
15        certain  JJ       3
18        curious  JJ       3
34           nice  JJ       2
43         little  JJ       2
48           good  JJ       2
61     improbable  JJ       2
62           best  JJ       2
72  philosophical  JJ       1
81       possible  JJ       1
```



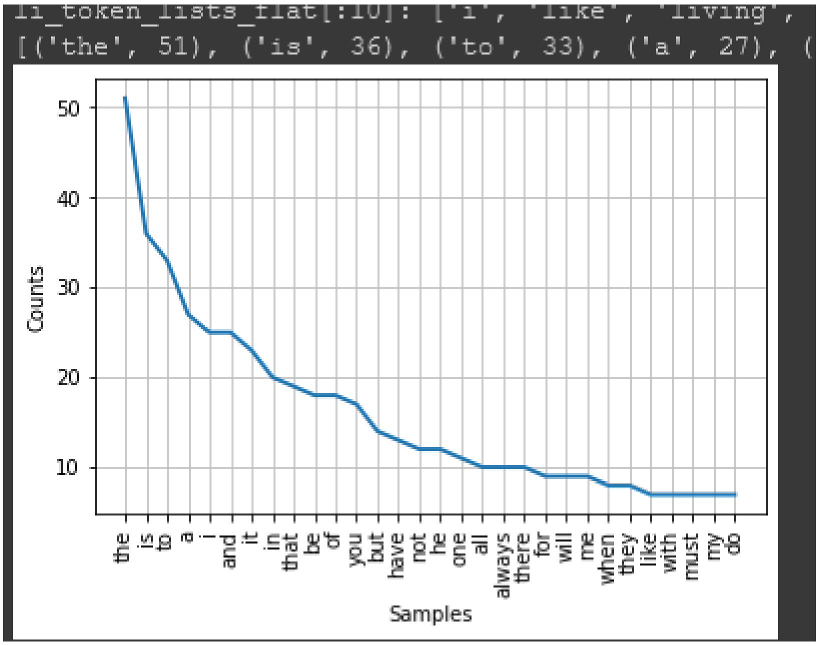
```
POS_TYPE: VB
          lem pos  counts
4        give  VB       5
12        say  VB       4
13       come  VB       4
22        see  VB       3
23       make  VB       3
26      think  VB       3
29  eliminate  VB       2
39       wish  VB       2
52       lose  VB       2
59       know  VB       2
```



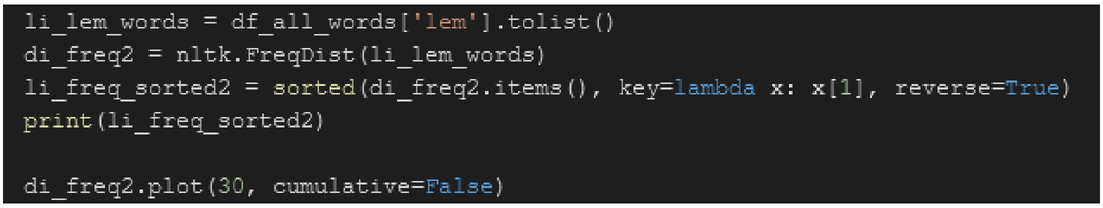
```
POS_TYPE: RB
               lem pos  counts
0           always  RB      10
14           never  RB       3
16           still  RB       3
27         however  RB       3
51          really  RB       2
56           quite  RB       2
83    particularly  RB       1
95            mere  RB       1
100          often  RB       1
119      therefore  RB       1
```



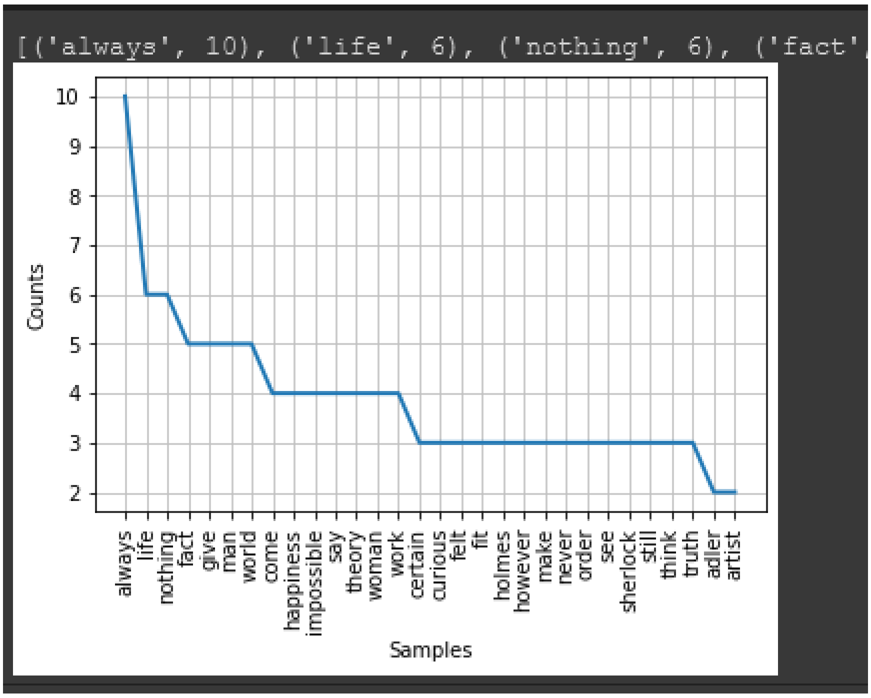
```
li_token_lists_flat = [y for x in li_token_lists for y in x]  # flatten the
print("li_token_lists_flat[:10]:", li_token_lists_flat[:10])

di_freq = nltk.FreqDist(li_token_lists_flat)
del di_freq['']
li_freq_sorted = sorted(di_freq.items(), key=lambda x: x[1], reverse=True)
print(li_freq_sorted)

di_freq.plot(30, cumulative=False)
```



```
li_token_lists_flat[:10]: ['i', 'like', 'living',
[('the', 51), ('is', 36), ('to', 33), ('a', 27), (
```



```
li_lem_words = df_all_words['lem'].tolist()
di_freq2 = nltk.FreqDist(li_lem_words)
li_freq_sorted2 = sorted(di_freq2.items(), key=lambda x: x[1], reverse=True)
print(li_freq_sorted2)

di_freq2.plot(30, cumulative=False)
```



```
[('always', 10), ('life', 6), ('nothing', 6), ('fact',
```

CONCLUSION

This work describes an automated technique for identifying children with autism who, with the help of Sinhala language support, can surpass human agents in terms of efficiency, effectiveness, and profitability. Our method combines a data-driven conversational system with deep learning models that learn from massive datasets using several GPUs. We expect that as more processing power and Sinhalese datasets become available, this approach will continue to improve, revolutionizing the autism screening procedure.

The suggested approach is an automated autism diagnostic tool based on machine learning to decrease or remove ineffective and error-prone human intervention in the field. The suggested system's capacity to support both English and Sinhalese enables eficient and dependable operation, which has a direct impact on service quality.

REFERENCES

[1] M. D. Hossain, H. U. Ahmed, M. J. Uddin, W. A. Chowdhury, M. S. Iqbal, R. I. Kabir, I. A. Chowdhury, A. Aftab, P. G. Datta, G. Rabbani and S. W. Hossain, "Autism Spectrum disorders (ASD) in South Asia: a systematic review," *BMC psychiatry,* pp. 1-7, 2017.

[2] A. T. Geiger, "Autism Speaks," Autism Speaks, [Online]. Available: https://www.autismspeaks.org/what-are-symptoms-autism. [Accessed 10 10 2021].

[3] D. A. Baribeau, S. Vigod, E. Pullenayegum, C. M. Kerns, P. Mirenda, I. M. Smith, T. Vaillancourt, J. Volden, C. Waddell, L. Zwaigenbaum and T. Bennett, "Repetitive behavior severity as an early indicator of risk for elevated anxiety symptoms in autism spectrum disorder," *Journal of the American Academy of Child & Adolescent Psychiatry,* pp. 890-899, 2020.

[4] A. T. Geiger, "Autism Speaks," [Online]. Available: https://www.autismspeaks.org/medical-conditions-associated-autism. [Accessed 10 10 2021].

[5] "verywell health," Dotdash, Inc., [Online]. Available: https://www.verywellhealth.com/what-are-the-three-levels-of-autism-260233. [Accessed 10 10 2021].

[6] "Complete Children's Health," Complete Children's Health, 2021. [Online]. Available: https://www.completechildrenshealth.com.au/what-are-the-three-levels-of-asd/. [Accessed 10 10 2021].

[7] M. van't Hof, C. Tisseur, I. van Berckelear-Onnes, A. van Nieuwenhuyzen, A. M. Daniels, M. Deen, H. W. Hoek and W. A. Ester, "Age at autism spectrum disorder diagnosis: A systematic review and meta-analysis from 2012 to 2019," pp. 862-873, 2021.

[8] S. L. Hyman, S. E. Levy and S. M. Myers, "Identification, evaluation, and management of children with autism spectrum disorder," *Pediatrics,* p. 145, 2020.

[9] J. Yuan, C. Holtz, T. Smith and J. Luo, "Autism spectrum disorder detection from semi-structured and unstructured medical data," *EURASIP Journal on Bioinformatics and Systems Biology,* pp. 1-9, 2017.

[10] I. M. Nasser, M. Al-Shawwa and S. S. Abu-Naser, "Artificial Neural Network for Diagnose Autism Spectrum Disorder," *International Journal of Academic Information Systems Research (IJAISR),* 2019.

[11] N. Van Hieu and N. L. Hien, "Artificial neural network and fuzzy logic approach to diagnose autism spectrum disorder," *International Research Journal of Engineering and Technology,* pp. 1-7, 2018.

[12] M. Thomas and A. Chandran, "Artificial Neural Network for Diagnosing Autism Spectrum Disorder," *International Conference on Trends in Electronics and Informatics (ICOEI),* pp. 930-933, 2018.

[13] N. M. Rad and C. Furlanello, "Applying deep learning to stereotypical motor movement detection in autism spectrum disorders," *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW),* pp. 1235-1242, 2016.

[14] A. Crippa, C. Salvatore, P. Perego, S. Forti, M. Nobile, M. Molteni and C. I., "Use of machine learning to identify children with autism and their motor abnormalities," *Journal of autism and developmental disorders,* pp. 2146-2156, 2015.

[15] C. Z. Hasan, R. Jailani, N. M. Tahir, I. M. Yassin and Z. I. Rizman, "Automated classification of autism spectrum disorders gait patterns using discriminant analysis based on kinematic and kinetic gait features," *Journal of Applied Environmental and Biological Sciences ,* vol. 7, pp. 150-156, 2017.

[16] S. Mujeeb, M. Hafeez and T. Arshad, "Aquabot: a diagnostic chatbot for achluophobia and autism," *Int J Adv Comput Sci Appl,* pp. 39-46, 2017.

[17] M. Elbattah, J. L. Guérin, R. Carette, F. Cilia and G. Dequen, "NLP-Based Approach to Detect Autism Spectrum Disorder in Saccadic Eye Movement," *IEEE Symposium Series on Computational Intelligence (SSCI),* pp. 1581-1587, 2020.

[18] V. Spector and M. H. Charlop, "Spector, V. and Charlop, M.H., 2018. A sibling-mediated intervention for children with autism spectrum disorder: Using the natural language paradigm (NLP)," *Journal of autism and developmental disorders,,* pp. 1508-1522, 2018.

[19] M. J. Maenner, K. A. Shaw and J. Baio, "Prevalence of autism spectrum disorder among children aged 8 years—autism and developmental disabilities monitoring network, 11 sites, United States, 2016," *MMWR Surveillance Summaries,* pp. 1-12, 2020.

[20] H. Perera, K. Wijewardena and R. Aluthwelage, "Screening of 18–24-month-old children for autism in a semi-urban community in Sri Lanka," *Journal of tropical pediatrics,* pp. 402-405, 2009.

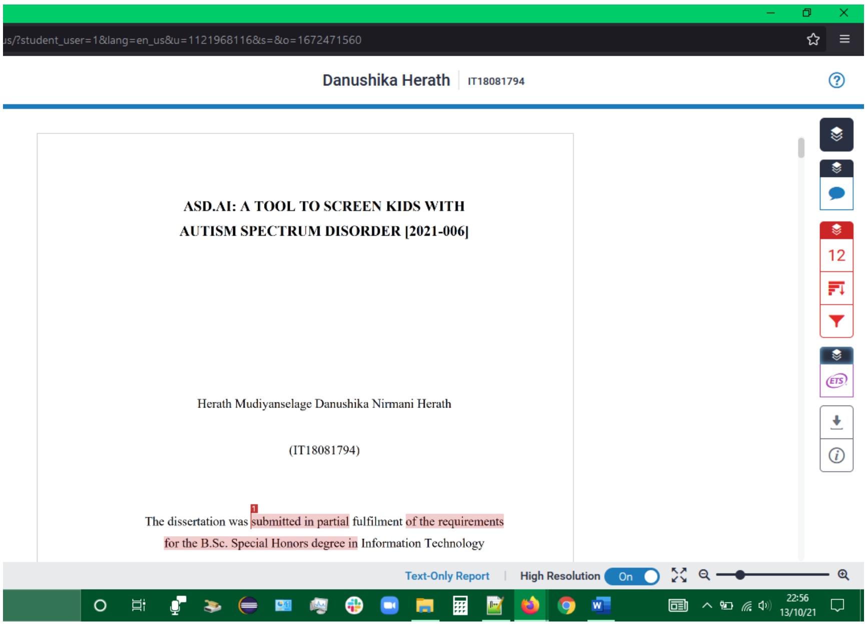
[21] H. Perera, K. C. Jeewandara, S. Seneviratne and C. Guruge, "Culturally adapted pictorial screening tool for autism spectrum disorder: A new approach," *World journal of clinical pediatrics,* 2017.

[22] N. Nazardeen, L. Yapa, M. Imthath, S. Pathirana and P. A. Jayathunga, "Software Package for Analyzing and Evaluating Children with Autism Spectrum Disorder (ASD) in Sri Lanka," *2020 International Conference on Decision Aid Sciences and Application (DASA),* pp. 985-990, 2020.

[23] N. Muttiah, "Ground realities of autism spectrum disorders in Sri Lanka," *Disability, CBR & Inclusive Development,* 2021.

Appendices

Plagiarism report screenshot



us/?student_user=1&lang=en_us&u=1121968116&s=&o=1672471560

Danushika Herath | IT18081794

ASD.AI: A TOOL TO SCREEN KIDS WITH AUTISM SPECTRUM DISORDER [2021-006]

Herath Mudiyanselage Danushika Nirmani Herath

(IT18081794)

The dissertation was [1] submitted in partial fulfilment of the requirements for the B.Sc. Special Honors degree in Information Technology