TAMIL CORPUS CREATION, ANALYSIS AND SENTIMENT ANALYSIS

A PROJECT REPORT

Submitted by

GUNA S
(Reg. No: 19S011)

of

5 Year Integrated M.Sc., (Data Science)

in

DEPARTMENT OF APPLIED MATHEMATICS AND COMPUTATIONAL SCIENCE

THIAGARAJAR COLLEGE OF ENGINEERING

(A Govt. Aided Autonomous Institution affiliated to Anna University)

MADURAI – 625 015

December 2022

THIAGARAJAR COLLEGE OF ENGINEERING, MADURAI

DEPARTMENT OF APPLIED MATHEMATICS AND COMPUTATIONAL SCIENCE

BONAFIDE CERTIFICATE

Certified that this project report "TAMIL CORPUS CREATION, ANALYSIS AND SENTIMENT ANALYSIS" is the bonafide work of S.GUNA (19S011), seventh Semester student of 5 Year Integrated MSc (Data Science) Degree Programme, who carried out the project under my supervision from July to December during the academic year 2022-2023.

Head of the Department
Dr. S. Parthasarathy
Professor & Head
Department of Applied Mathematics and Computational Science

Dr. T. Chandrakumar
Project Guide (Internal)
Assistant Professor in Data Science
Department of Applied Mathematics and Computational Science

Submitted for the viva – voce Examination held at Thiagarajar college of Engineering, Madurai on _____.

_____ _____

Internal Examiner **External Examiner**

ACKNOWLEDGEMENT

My endeavor stands incomplete without dedicating my gratitude to a few people who have contributed a lot towards the successful completion of my project work."

I sincerely thank Almighty, for making my life to be more interesting, challenging and happier during project tenure.

I am pleased to convey my gratitude to **Dr.M.Palaninatharaja**, Principal, Thiagarajar College of Engineering, Madurai, for providing me this opportunity to do my project at Indraprastha Institute Of Information and Technology, Delhi.

I wish to express my sincere thanks to **Dr.S.Parthasarathy,** Head of the Department of Applied Mathematics and Computational Science, Thiagarajar College of Engineering, Madurai for his support and ardent guidance.

I am extremely thankful at the most to my internal guide **Dr.T.Chandrakumar**, Associate Professor, Dept. of Applied Mathematics and Computational Science, Thiagarajar College of Engineering, for his continual support and enduring guidance throughout my project tenure. My earnest thanks to all the staff members, Department of Applied Mathematics and Computational Science, for their constant care and support.

I express my faithful thanks to External Guide **Dr.Rajiv Ratn Shah,** IIIT Delhi, for his guidance and expert advice rendered by him in modest attempt at preparing this project.

I wish to express my deep-felt gratitude to my beloved parents for their constant support and contribution to this project work. Also, I would like to thank all my teachers, friends and well-wishers who have helped me for doing this project and throughout the 5 Year Integrated M.Sc (Data Science) course.

Guna S

INDRAPRASTHA INSTITUTE OF INFORMATION TECHNOLOGY DELHI
(A state university established by Govt. of NCT of Delhi)
Okhla, Phase III (Near Govind Puri Metro Station), New Delhi-110020, INDIA
Telephone: +91 11 26907561 Website: http://www.iiitd.ac.in

Rajiv Ratn Shah
Director, Multimodal Digital Media Analysis (MIDAS) Lab
Head, Department of Human-Centered Design
Faculty, Department of Computer Science and Engineering
Indraprastha Institute of Information Technology (IIIT)
Delhi, India 110020
Tel.: (+91) (11) 26907495; Email: rajivratn@iiitd.ac.in
https://www.iiitd.ac.in/rajivratn
November 17, 2022

To Whom It May Concerns

Internship Experience for S. GUNA

This is to certify that S. GUNA, a student of Thiagarajar College of Engineering, pursuing MSc Data Science has worked under my supervision from July 2022 and will continue working until December 2022.

In the development of Tamil Natural Language Project (NLP), the student had been exposed to different processes and tasks. The student was found diligent, hardworking, and inquisitive. I wish all success in the candidate's life and career.

Please do not hesitate to contact me if I can be of any further assistance

Sincerely,

Dr. Rajiv Ratn Shah
Head of The Department, Human-Centered Design (HCD)
Indraprastha Institute of Information Technology
(A State University Established by Govt. of Delhi)
Okhla Phase-III, New Delhi-110020

ABSTRACT

Natural language processing is a subfield of linguistics, computer science, and artificial intelligence concerned with the interactions between computers and human language, in particular how to program computers to process and analyse large amounts of natural language data. In order to machine understand the human language NLP combines computational linguistics—rule-based modelling of human language with statistical, machine learning, and deep learning models.

Sentiment analysis is the systematic identification, extraction, quantification, and study of affective states and subjective data. It uses natural language processing, text analysis, computational linguistics, and biometrics. Sentiment analysis, often known as opinion mining, is a technique used in natural language processing (NLP) to determine the emotional undertone of a document.

This project's goal is to create a python library for Tamil language in order to carry out various Natural Language Processing tasks which includes tokenization, Lemmatization, Part Of Speech tagging, Morphological tagging, Dependency Relation Tagging and Named Entity Recognition tagging. In order to understand necessity of our library, I built a model to perform sentiment analysis on Tamil text.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVATIONS

S. No	Abbreviation / Acronym	Description
1	IIITD	Indraprastha Institute of Information and Technology, Delhi
2	GUI	Graphical User Interface
3	POS	Part Of Speech
4	UPOS	Universal Part Of Speech
5	XPOS	Language-specific Part Of Speech
6	IAA	Inter Annotator Agreement
7	NLP	Natural Language Processing

1. INTRODUCTION

1.1 ORGANIZATION PROFILE

Indraprastha Institute of Information Technology Delhi (IIIT-Delhi) was created by an act of Delhi legislature empowering it to carry out R&D, conduct educational programs, and grant degrees. The General Council is the apex body of the Institute, chaired by Hon'ble Lt. Governor of Delhi and the Board of Governors is the policy and decision-making body of the Institute. The Senate is empowered to take all academic decisions.

IIIT-Delhi is accelerating on the path of becoming one of the leading comprehensive research-led teaching institutes in India and has proven to be consistently responsive towards the evolving needs of society. The faculty members at IIIT-Delhi are among the finest in the country and are internationally recognized. Carrying out cutting-edge research is in the institutional DNA of IIIT-Delhi.

MIDAS is a group of researchers at IIIT-Delhi who study, analyze, and build different multimedia systems for society leveraging multimodal information. MIDAS stands for Multimodal Digital Media Analysis Lab and it is founded by Dr. Rajiv Ratn Shah. Dr. Shah is an assistant professor in the Department of Computer Science and Engineering at IIIT-Delhi.

MIDAS works in the fields of Machine Learning, Multimedia Content Processing, Natural Language Processing, Image Processing, Multimodal Computing, Data Science, and Social Media Computing towards AI for Social Good.

1.2 EXISTING SYSTEM

INLTK library is one of the libraries which supports Tamil language. But it lacks the capability to split the token in accurate manner. Since the tokenization forms the basis of NLP pre-processing it is not a reliable library. TRANKIT is a light-weight transformer-based toolkit which can be used for Tamil text pre-processing. However, it inaccurate with splitting the multi-word tokens. In Tamil language multi-word token splitting is crucial to

find the underlying semantic meaning.

Other major issue of Trankit is it don't perform lemmatization based on Part of Speech tag, while standard libraries like NLTK in English use this method to produce exact lemmatization of words. Trankit output format is also hard to interpret.

1.3 PROPOSED SYSTEM

This project's goal is to create a python library for Tamil language in order to carry out various Natural Language Processing tasks which includes tokenization, Lemmatization, Part Of Speech tagging, Morphological tagging, Dependency Relation Tagging and Named Entity Recognition tagging and to create a sentiment analysis model for Tamil in order to know the challenges while processing low resource language.

1.3.1 Advantages of Proposed System

- By Creating a python library which support Natural Language Processing for Tamil language, we can do various NLP tasks with ease.
- This Library will help in the preprocessing of Tamil language, which opens up the number of NLP Applications including sentiment analysis, text generation that can be done in Tamil language which deals a very huge impact in business point-of-view.
- Basically, preprocessing in NLP is a huge time-consuming process by using this library it can be done in a fast manner without compensating in quality of data.

2. SYSTEM ANALYSIS

2.1 FEASIBILITY STUDY

A feasibility study is merely an evaluation of how realistic a project plan or procedure is. By examining technical, economic, legal, operational, and time feasibility factors, this is accomplished. A proposed plan or project's viability is assessed in a feasibility study. A project or business venture is assessed for its viability as part of a feasibility study to ascertain whether it will be successful. The Primary feasibility for this project as follows:

- Does it accurate like Manual Preprocessing?
- Does the automation faster than preprocessing it manually?

2.1.1 Operational Feasibility

Operational feasibility is the measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development.

The primary considerations involved in the Operational Feasibility analysis are as follows:

- Is the quality of the corpus measured with IAA Score >80?
- Time taken for performing task is considerably low than other libraries out there?

2.1.2 technical feasibility

Technical feasibility studies determine how well it can support proposed additions. Technical feasibility analysis assists in determining whether technical resources meet capacity and in evaluating hardware, software, and other technical requirements. The most difficult aspect to ensure in the early stages is technical feasibility. Because the objective function and performance cannot be fully predicted, everything appears to be to be possible if proper assumptions are made. It is pivotal that the analysis and definition processes run concurrently with a technical feasibility assessment.

The resource availability at the organization where the project is to be developed and implemented is typically associated with technical feasibility. This includes financial considerations to allow for technological advancement. If it is not satisfied, the project is deemed unfeasible. The proposed system was developed using streamlit and python. The technical benefits of streamlit are as follows:

- Works with TensorFlow, Keras, Pandas, NumPy, Matplotlib, Gspread and more.
- Free and open source.
- Build a Graphical User Interface (GUI) in a dozen lines of Python.

2.1.3 Economic Feasibility

To ascertain whether there is an economic basis for the investment decision, economic analysis is performed. This analysis covers more ground than what is generally covered in a financial analysis. The economic interest from the project.

The economic costs of the project.

- The balance of these expressed in present value terms.

Economic costs and benefits are not always the same as financial cost and benefits. Economic analysis includes project impacts that do not have a market price and positive and negative impacts that are experienced by people who are not the direct users of the services. The packages and software used in this project are google spreadsheet, gspread, trankit, tensorflow and keras these all library and software all are of open source. The proposed solution will incur no software costs.

2.2 USE CASE DIAGRAM

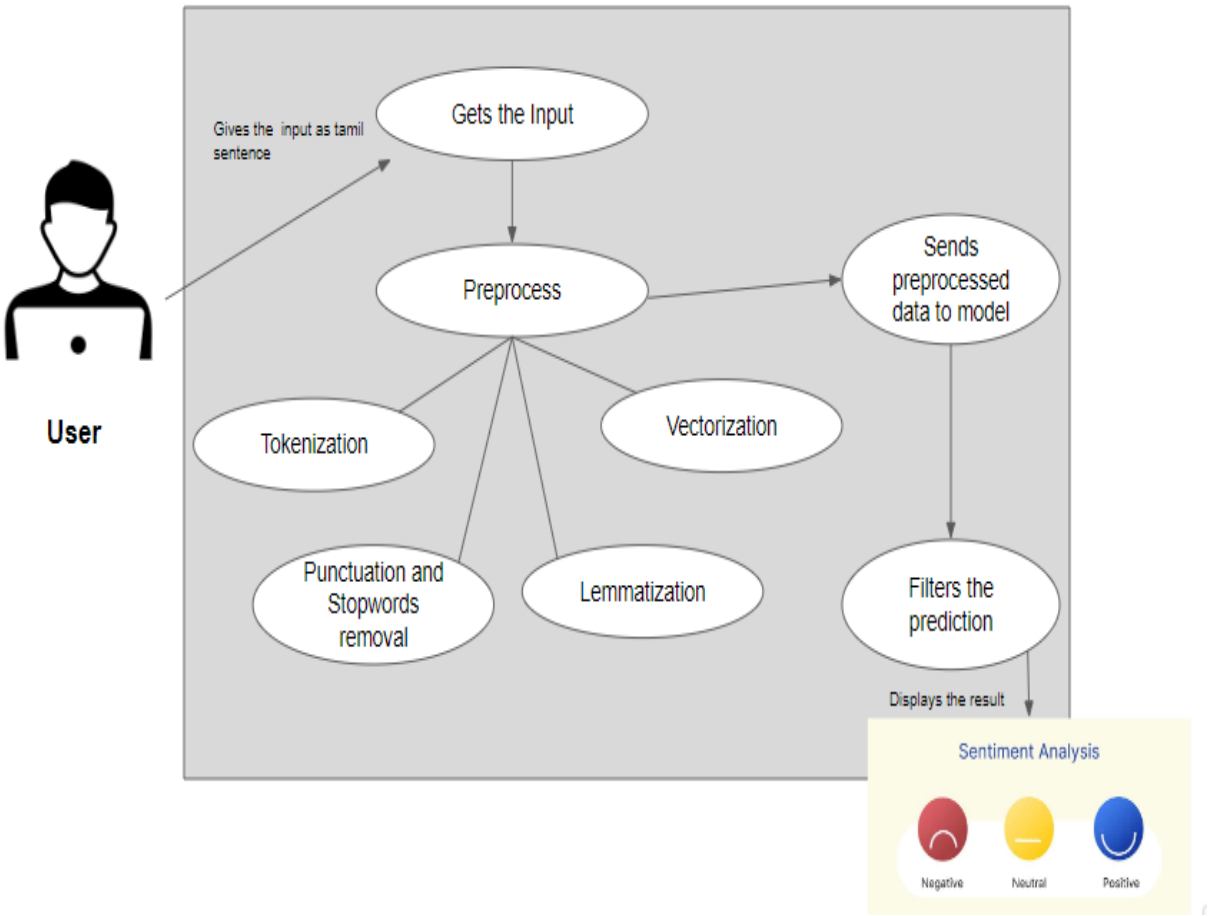


Fig 2.2.1 Use case diagram for sentiment analysis model

3. SYSTEM REQUIREMENT SPECIFICATION

3.1 SCOPE

The Scope of this project is to create a python library to carry out the various Natural Language Processing (NLP) tasks in Tamil language. As a part of a library creation, a corpus of 10000 Tamil sentence is annotated in CONLL-U Format. I created a model to perform sentiment analysis for Tamil language in order to understand the importance of Natural Language Processing Library for Tamil Language.

3.2 FUNCTIONAL REQUIREMENTS

The functional Requirements for Library Creation

Data Collection

- For Data Collection, initially 10000 Tamil sentence is collected.
- Data is collected in topics like News, Movies, Wikipedia, eBooks and Grammar from various open-source resources.

Data Upload

- In order to upload the data into a corpus, a python script is written.
- The python script will extract data from collected data and upload it into a corpus.

Annotation

- Data annotation follows CO – NLLU format and it is done manually.

CO -NLLU Format consist of Various tasks like.

- Tokenization where every word in a sentence is split into individual token.
- Tokenized word is then lemmatized to its root form.
- Part Of Speech Tagging is done for each words.

- Morphological Feature forms the basis of formation of words. Many tags are used for a word to find out the underlying meaning.
- Dependency parsing tagging is done for each word in order to find surface-level semantic relationship between each word in a sentence.
- Named Entity Recognition tags is used to classify the pre-defined entity to a certain class.
- Translation and transliteration of a sentence is given as a meta data.

Evaluation

- Inter-Annotator Agreement (IAA), a measure of how well multiple annotators can make the same annotation decision for a certain category.
- IAA Score will be calculated using a tool developed in Flask.

The functional Requirements for model

Preprocessing

- Tamil Sentence is initially given as input.
- The preprocessing of data like tokenization, lemmatization will be done using Trankit library.

Sentiment Analysis

- The GUI pass the processed input into saved LSTM model and process it.
- The GUI get the result from the model and display the result.

3.3 PERFORMANCE REQUIREMENT

System design and development require Performance Requirements (PR). There are three types of performance requirements: reaction time, throughput, and concurrency. Response time measures how quickly the system can respond to individual requests, as experienced by a real user (how many users or threads work simultaneously). The performance requirements are given below

3.3.1 Reusability

In this System, to reduce implementation time, reusable modules and classes are used, that is, a segment of source code that can be used again to add new functionalities with slight or no modification. The proposed system is developed in a way that it can be reused at any point of time without spending too much time on it.

3.3.2 Flexibility

Proposed system is developed using python and Streamlit, so the system or module can be easily modified for operational change. Both the sources used is open-source and one of the easiest tools to work with. This makes the module highly flexible for any operational changes in future.

3.3.3 Reliability

The module created here will be a failure free operation. The system shall be available all time. The module is created in a way that it can handle any issues. The module is built in a way that it can handle any exception at any point of time. The Module is highly reliable in terms of quality.

3.3.4 Performance

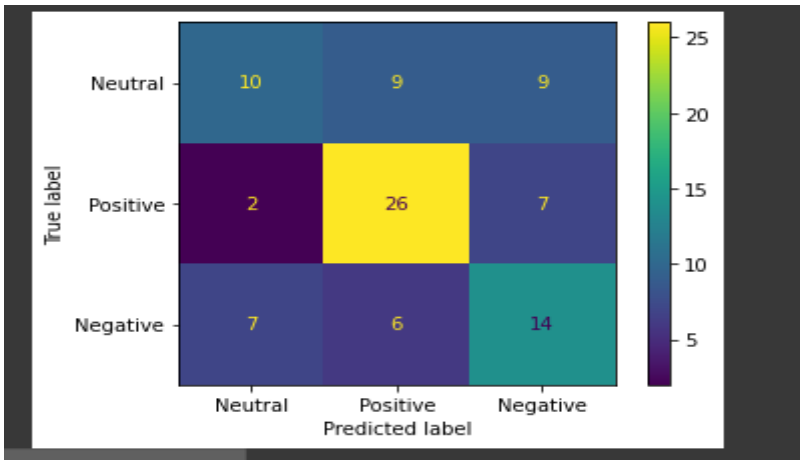


Fig 3.3.4.1 Performance Evaluation of model

3.3.5 Network Coverage

For optimal performance, the suggested system requires network connectivity. Wi-Fi with a high bandwidth is the preferable option.

3.4 SOFTWARE SPECIFICATION

Table 3.4.1 Software Specification Table

Package	**Specification**
Pip	22.0.3
Gspread	3.6.0
Trankit	1.1.1
Keras	2.7.0
Numpy	1.21.6
Transformers	4.24.0
Pandas	1.3.5
Sklearn	1.0.2
XlsXwriter	3.0.3
Streamlit	1.14.0

3.5 HARDWARE SPECIFICATION (USED)

Table 3.5.1 Hardware Specification Table

Unit	**Configuration (recommended)**
Installed RAM	4.00 GB
Processor	AMD PRO A4-4350B R4, 5 COMPUTE CORES 2C+3G 2.50 GHz

3.6 TECHNOLOGIES AND LIBRARIES USED

Technology: Python

Libraries: Gspread, Trankit, Keras, Streamlit, XlsxWriter

3.6.1 Python

Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured, object-oriented and functional programming.

Some of the important features of python are

- Python strives for a simpler, less-cluttered syntax and grammar while giving developers a choice in their coding methodology.
- Python code looks like simple English words. There is no use of semicolons or brackets, and the indentations define the code block. You can tell what the code is supposed to do simply by looking at it.
- Python has an extensive standard library available for anyone to use. This means that programmers don't have to write their code for every single thing unlike other programming languages.

3.6.2 Gspread

GSpread is built on Google Sheets API some authentication needs to be done for the app via a google account to allow access to sheets via script. It enables you to easily pull data from Google spreadsheets into DataFrames as well as push data into spreadsheets from DataFrames. It will access the google drive with the help of key file generated from google service account.

3.6.3 Trankit

Trankit is a light-weight Transformer-based Python Toolkit for multilingual Natural Language Processing (NLP). It provides a trainable pipeline for fundamental NLP tasks over 100 languages, and 90 downloadable pretrained pipelines for 56 languages. Trankit perform NLP tasks like tokenization, lemmatization, POS tagging etc., It is used in the model for preprocessing.

NLP's Transformer is a new architecture that aims to solve tasks sequence-to-sequence while easily handling long-distance dependencies. Computing the input and output representations without using sequence-aligned RNNs or convolutions and it relies entirely on self-attention.

3.6.4 Keras

Keras is an open-source software library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library. Keras is a deep learning API written in Python, running on top of the machine learning platform TensorFlow. It was developed with a focus on enabling fast experimentation. Being able to go from idea to result as fast as possible is key to doing good research.

Some of the key features of Keras which makes it a very popular library for neural networks

- Keras reduces developer *cognitive load* to free you to focus on the parts of the problem that really matter.
- Keras adopts the principle of *progressive disclosure of complexity*: simple workflows should be quick and easy, while arbitrarily advanced workflows should be *possible* via a clear path that builds upon what you've already learned.
- Keras provides industry-strength performance and scalability: it is used by organizations and companies including NASA, YouTube, or Waymo.

3.6.5 Streamlit

Streamlit is an open-source Python library that makes it easy to create and share beautiful, custom web apps for machine learning and data science. Streamlit is one of the libraries we can easily design and launch a web application. Streamlit allows you to write code in the same way as in python. Streamlit is a very light library it can be even used in low specification system. Streamlit can provide seamless integration with libraries like keras, matplotlib, transformers, scikit-learn and many other libraries which supports machine learning and deep learning.

Streamlit has a distinct data flow; whenever something changes in the code or something needs to be updated on the screen, Streamlit completely restarts the Python script from top to bottom. This occurs when a user interacts with a widget, such as a choose box or drop-down box, or when the source code is altered.

Streamlit can be used to create a application with catchy and more effective Graphical User Interface. Many heavy libraries lack the ability to develop an application in a quick manner without going through its complicated process. On the other hand, Streamlit is light, quick and extremely easy to develop. Streamlit is a potential open-source Python toolkit that allows developers to quickly create appealing user interfaces.
Required packages and applications:

1. Python — Install version 3.7.9 along with the creation of virtual environment.
2. pip — Install pip with the help of the terminal or using the code editor.
3. Streamlit — Install the Streamlit library before launching any Streamlit

application. Run the following command in the terminal to install Streamlit.
Command to run streamlit app: “Streamlit run app.py.

3.6.6 XlsxWriter

XlsxWriter is a Python module that can be used to write text, numbers, formulas and hyperlinks to multiple worksheets in an Excel 2007+ XLSX file. It supports features such as formatting and many more, including Integration with Pandas, 100% compatible Excel XLSX files, Integration with Pandas, memory optimization mode for writing large files.

4. System Design

Systems design is the process of defining a system's components, including modules, architecture, components, their interfaces, and data, depending on the requirements that have been given. The main procedure that forms the basis for the development of every software product is system design. Every project has a design phase that creates a prototype of the product that is almost identical to the product being created while taking into account the findings of the analysis phase.

4.1 SYSTEM ARCHITECTURE DIAGRAM

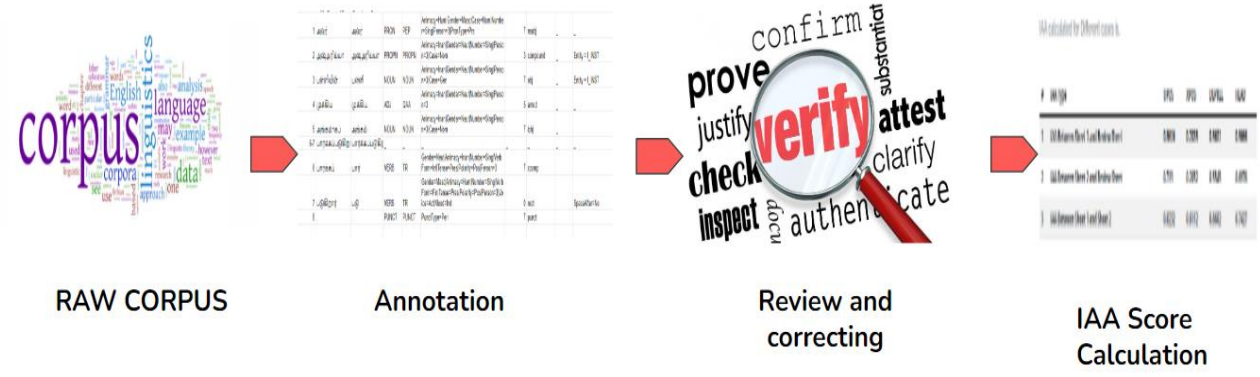


Fig 4.1.1 System Architecture Design

4.2 DATA FLOW DIAGRAM

The movement of data within a system from a point of origin to a predetermined destination that is denoted by a line or arrow is known as dataflow. The graphical representation of the data movements, procedures, and files (data storage) used to support information systems is called a dataflow diagram.

The data flow diagram (DFD) is one of the most important tools used by system analysts. Data flow diagrams are made up of a number symbols, which represent system components. Most data flow modeling methods use four kinds of symbols. These symbols are used to represent four kinds of system components. Processes, data stores, data flows and external entities.

DFD is the graphic representation of data movement process, and files used in support of an information system.

There are several rules of thumb used in drawing DFDs.

- Process should be named and numbered for easy references.
- The direction of flow is from top to bottom and from left to right.

4.2.1 Level 0 DFD

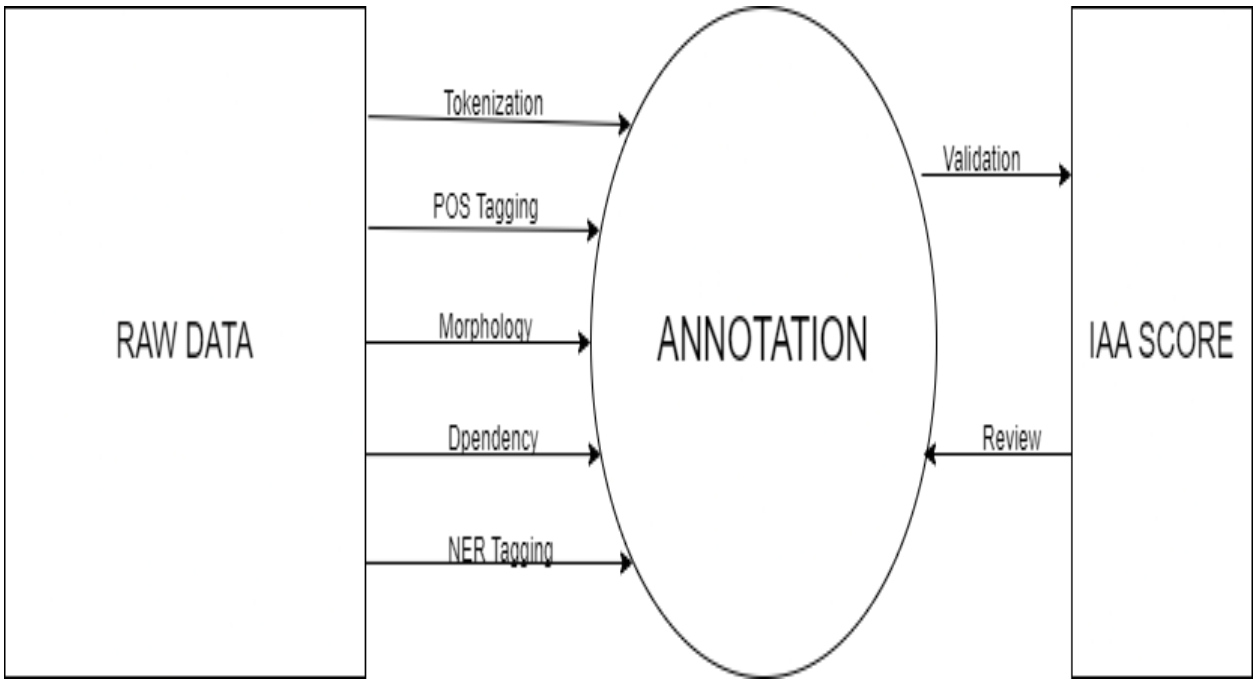


Fig 4.2.1.1 DFD Level 0 for Corpus

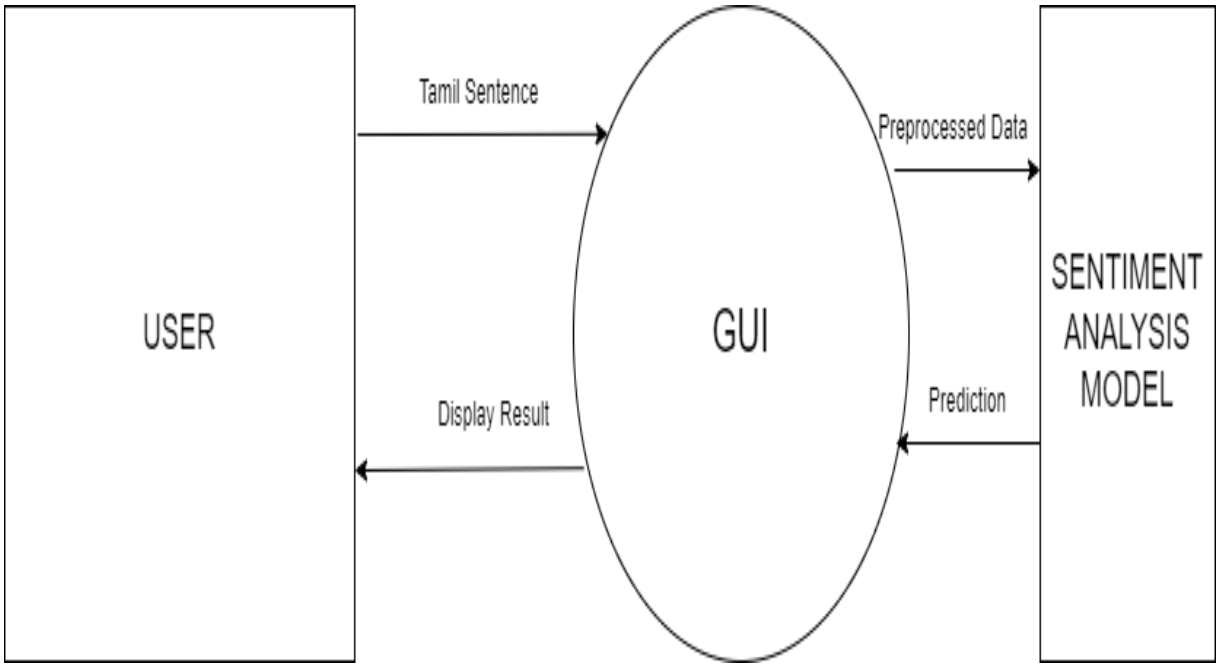


Fig 4.2.1.2 DFD Level 0 for Model

4.3 PROCESS FLOW DIAGRAM

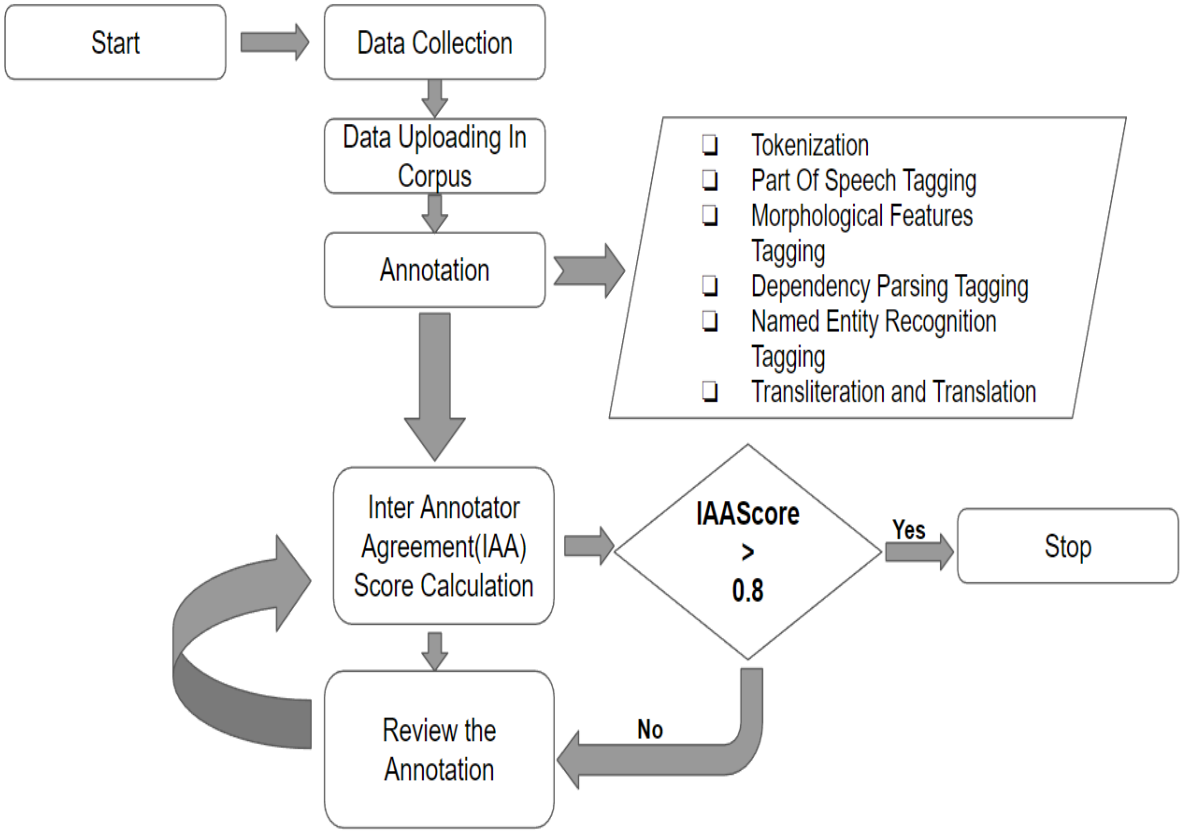


Fig 4.3.1 Process Flow Diagram for Model

4.4 LOGICAL DESIGN USING UML

4.4.1 Sequence Diagram

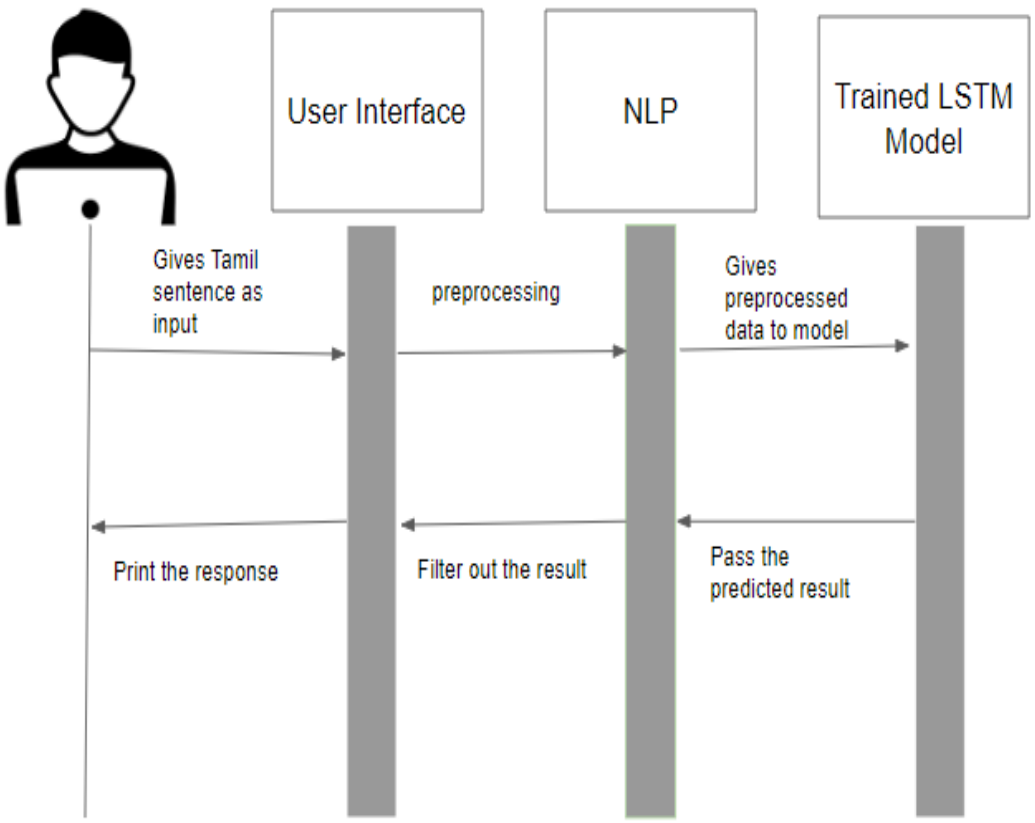


Fig 4.4.1 Sequence Diagram for model

4.5 MODULE DESCRIPTION

Module Description for Corpus

Data Upload

- Tamil Data is Collected in topics of News, Movies, Wiki, eBooks and grammar from various open- source resources of internet and it is stored.
- Then the collected data is uploaded into google spreadsheet using a python script.
- The python script uses nltk library for initial tokenization of words.
- On using nltk library, multi word tokens in Tamil language could not be split, So further annotation is needed.
- Each topic of collected data has its own google spreadsheet.
- Data is uploaded in one reference sheet and two different team sheet.

Annotation

- Data Annotation follows CoNLL-U format. Annotation is done manually.
- CoNLL-U. Annotations are encoded in plain text files with three types of lines
    - Word lines containing the annotation of a word/token in 10 fields separated by single tab characters.
    - Blank lines marking sentence boundaries.
    - Comment lines starting with hash
- Tokenization is the process of splitting text into tokens. The tokens can be words, numbers or punctuations, sub words etc... Sentence tokenization is performed on the input text to split the input into individual sentences. In Tamil Language, words are combined and formed into a compound word. These compound words have to be decomposed into separate words during the process of tokenization.
- Lemmatization aims to remove inflectional endings only and to return the base or dictionary form of a word.
- Universal Part Of Speech (UPOS) tags mark the core part of speech Categories. The part of speech indicates how the word functions in meaning as well as

grammatically within the sentence. It consists of various tags to indicate the words functionalities grammatically in a Sentence.

- Language-specific part-of-speech (XPOS) contains a set of tags for some Universal Part Of Speech (UPOS) to indicate the functionalities of the word in a sentence more explicit manner.
- Morphology is used study for word formation how words are built from small pieces. There are two types of Morphological Features. Lexical and Inflectional Features. Features are additional pieces of information about the word, its part of speech and morphosyntactic properties. Every feature has the form Name = Value and every word can have any number of features, separated by the vertical bar.
- Dependency parsing is the process of identifying the semantic relations between the words in the sentence. In Dependency Parsing, various tags represent the various relationship between two words in a sentence.
- A named entity is a real-world object, such as a person, location, organization, product, etc., that can be denoted with a proper name. It can be abstract or have a physical existence. MISC field is for storing any additional information that does not fit into any of the other fields.
- Translation and Transliteration of a sentence is also given as a meta data.

Review

- In order to maintain quality annotation, various review process will be done.
- A python script is used to check whether used tags are all legal tags and credible tags.
- Python Script is used to validate the correctness of tag used.
- If any changes are made in guidelines, another python script is used to automate the updating.
- A file will display the words with its POS tags, by using this method quality annotation is ensured.

Validation

- Inter-Annotator Agreement (IAA), a measure of how well multiple annotators can make the same annotation decision for a certain category. The measures taking expected chance agreement into account:
  - Cohen's κ: two annotators annotating each instance with a category
  - Fleiss' κ: each instance was annotated n times with a category
  - Cohen kappa is calculated between a pair of annotators and Fleiss' kappa over a group of multiple annotators.
- In this module, Fleiss' Kappa is used for validation between two team sheets and a validation sheet.
- IAA Score is calculated using a tool developed in Flask.
- IAA score ranges between 0 to 1, the IAA score is directly proportional to quality of annotation.

Module description for Sentiment Analysis Model

Sentiment Analysis

- Sentiment analysis is the use of natural language processing, text analysis, computational linguistics, and biometrics to systematically identify, extract, quantify, and study affective states and subjective information.
- Data is collected from the Tamil corpus and annotated with labels like positive, neutral and negative with respective to the semantic meaning of the sentence.

Data Preprocessing

- In Natural Language processing, preprocessing of data involves various steps.
- For Tamil language, data preprocessing is done using Trankit library.
- Initially, a sentence is tokenized, where each token represents each word in a sentence.
- Then punctuations in the tokens is removed.

- Then stop words is removed from the tokens. Stop words are the words in a stop list which are filtered out before or after processing of natural language data because they are insignificant.
- Then tokens are lemmatized into its root word form.
- Then, Keras Tokenizer is used to vectorize a text corpus, which turns each text into either a sequence of integers where each integer being the index of a token in a dictionary.

LSTM Model

- Keras Embedding Layer is used for data embedding, which turns positive integer indices into dense vector of fixed size. It is capable of capturing context of a word in a document, semantic and syntactic similarity, relation with other words, etc.
- Data is split into train and test sets.
- Then the embedded data is passed into LSTM Neural Network architecture. LSTM model get trained using the train data and evaluate the performance of the model using test dataset.
- Long Short-Term Memory (LSTM) model is a kind of Recurrent Neural Network (RNN) that is capable of learning long term dependencies in data.
- I used LSTM model, since it performs well in small dataset rather than BERT and other Machine Learning algorithms.

Graphical User Interface

- Graphical User Interface (GUI) for this model is created using streamlit.
- Streamlit is one of the libraries we can easily design and launch a web application.
- The web application has two different features one liner and whole review.
- In one liner, input data will be a single sentence and it will get result based on that single sentence.
- Whole Review, input data will be a collection of sentences about a same topic based on that it will evaluate and give the result as a whole.

5. TESTING

5.1 TESTING OBJECTIVE

Software testing is a crucial phase in the creation of the system. In general, system testing involves examining how well each system part is integrated. System testing's mission is to find inconsistencies between the system and the system's original objective. An information system's quality is determined by its creation, design, and execution. Testing is the step of development that is most crucial. Testing is the process of identifying flaws or bugs in a system.

Testing ensures that the user's requirements are met. In other words, it is a procedure for detecting flaws in a system.

- Unit testing
- Integration Testing
- User Interface Testing
- Functional Testing
- System Testing

5.2 UNIT TESTING

The smallest chunk of code that can be logically separated in a system is called a unit, and a unit test is a method of testing a unit. That is a method, a function, a subroutine, or a property in the majority of programming languages. The definition's isolated portion is crucial. Instead of focusing on the user experience, unit testing analyses the

Table 5.2.1 Sample Test Cases for Corpus

Test Condition Description	**Test Case No.**	**Test case/ scenario/ data description**	**Expected Result**	**Actual Result**	**Test Result**
Check whether all tags used in corpus comes under guidelines	UT01	Get all Unique tags In UPOS Column	Checks whether all tags used in in UPOS column comes under guidelines	Checks whether all tags used in in UPOS column comes under guidelines	PASS
	UT02	Get all Unique tags In XPOS Column	Check whether all tags used in in XPOS column comes under guidelines	Checks whether all tags used in in XPOS column comes under guidelines	PASS
	UT03	Get all Unique tags In FEATS Column	Checks whether all tags used in in FEATS column comes under guidelines	Check whether all tags used in in FEATS column comes under guidelines	PASS
	UT04	Get all Unique tags In DEPREL Column	Checks whether all tags used in in DEPREL column comes	Checks whether all tags used in in DEPREL column	PASS

			under guidelines	comes under guidelines	
	UT05	Get all Unique tags In MISC Column	Checks whether all tags used in in MISC column comes under guidelines	Checks whether all tags used in in MISC column comes under guidelines	PASS
Check whether corpus has any missing value	UT06	Get the count of all columns.	Checks whether all the tokens are properly annotated without any missing value	Checks whether all the tokens are properly annotated without any missing value	PASS
Check the correctness of POS annotation	UT07	Extract the word with its respective UPOS and XPOS tags.	Checks whether POS tagging for every token is done correctly	Checks whether POS tagging for every token is done correctly	PASS

Table 5.2.2 Sample Test Cases for Model

Test Condition Description	**Test Case No.**	**Test case/ scenario/ data description**	**Expected Result**	**Actual Result**	**Test Result**
Check whether the input is accepted	UT11	Give Tamil sentence as input	Checks whether the given single Tamil sentence is accepted.	Checks whether the given single Tamil sentence is accepted.	PASS
	UT12	Give a .txt format file which contains n number of Tamil sentence as input	Checks whether the given file of Tamil sentence is accepted.	Checks whether the given file of Tamil sentence is accepted.	PASS
Check whether the model predicts the answer correctly	UT13	Give a positive sentence	Checks whether the given single Tamil sentence is positive.	Checks whether the given single Tamil sentence is positive.	PASS
	UT14	Give a Negative sentence	Checks whether the given	Checks whether the given single Tamil sentence is	PASS

			single Tamil sentence is Negative.	Negative.	
	UT15	Give a Neutral sentence	Checks whether the given single Tamil sentence is Neutral.	Checks whether the given single Tamil sentence is Neutral.	PASS

5.3 INTEGRATION TESTING

Integration testing is the phase in software testing in which individual software modules are combined and tested as a group. Integration testing is conducted to evaluate the compliance of a system or component with specified functional requirements

The integration testing is used in this way in the corpus

Table 5.3.1 Sample Integration Test for Corpus

Test Condition Description	**Test Case No.**	**Test case/ scenario/ data description**	**Expected Result**	**Actual Result**	**Test Result**
To Check the Inter Annotator Agreement	IT01	Give the two data sheet and Reference sheet as input	The calculated IAA Score Value for every filed is greater than 0.8	The calculated IAA Score Value for every filed is greater than 0.8	PASS

5.4 USER INTERFACE TESTING

The interface of a component is frequently used to build tests. Undoubtedly, the interface itself is a necessity for the component, which is why this type of testing is frequently referred to as black box testing. However, the emphasis on interfaces compels us to think about interface testing separately. Tests are produced from an interface specification for a component using methods like pair wise testing and interface mutation.

5.5 PERFORMANCE TESTING

Performance testing is intended to evaluate how well the software performs when used in the context of an integrated system. The Program's various features are tested. It requires assessing every navigation and each field value on the front-end pages under both favourable and unfavourable circumstances.

5.6 SYSTEM TESTING

The system test is run following the integration of each module. System testing examines each module's integration into the system rather than the software itself. Additionally, it checks for inconsistencies between the system's current specs, system documentation, and its original aim. The compatibility of specific modules is the main issue. The end users must check the project to see if it meets their needs after all the revisions are finished.

6. IMPLEMENTATION

Implementation forms an important phase in the development life cycle. This phase of software development is concerned with translating design specification into working model. Implementation is the final phase in achieving a successful system and in giving the users confidence that the system will work efficiently. The implementation of sentiment analysis model undergoes these process

6.1 NATURE OF DATA

- Data is collected from the corpus, which is extracted from the Tamil corpus used for annotation.
- Data consists of 450 Tamil sentences.
- Each sentence is labeled with Positive or Negative or Neutral Based on the context of the Sentence.

6.2 PREPROCESSING

- Initially, data is checked for whether the data consists of any missing value.
- Then, each sentence is tokenized into tokens.
- Stop words and Punctuations are removed from tokens, since they don’t give any contextual meaning.
- Then each token is lemmatized into its root form.
- Using Keras Tokenizer, tokens are vectorized.
- The Dependent value is converted into vectors using one hot encoding.

6.3 EMBEDDING

- An embedding is a relatively low-dimensional space into which you can translate high-dimensional vectors.
- Vectorized independent data is embedded in order to find out the semantic relationships.

6.4 LSTM MODEL

- Then Embedded data is passed into a LSTM Neural Network Model.
- The model will give the prediction of the sentence for each dependent label.
- Then, by using argmax () function the highest possibility is extracted.
- And on basis of that result is displayed.

7. CONCLUSION AND FUTUTRE ENHANCEMENT

7.1 CONCLUSION

The corpus for NLP library for Tamil language which performs various Natural Language Processing tasks was designed and implemented in accordance with the guidelines, and the corpus was confirmed to be bug and error free after tested with available test cases. The various units were thoroughly tested for all possible bugs using all applicable parameters in the context. After unit testing, integration was completed successfully, and the entire system was tested again via integration before implementation.

A Neural Network model is built for sentiment analysis in Tamil sentence in order to find out the challenges faced by other libraries which also support Tamil Language Preprocessing. The model undergoes various unit testing were thoroughly tested for all possible bugs using all applicable parameters in the context.

7.2 FEATURES

- A concise, user-friendly interface.
- Fast retrieval of information.
- Limited internet usage.
- Accurate and automatic detection.

7.3 FUTURE ENHANCEMENT

Currently, Annotations were done in manual only, future enhancement will be automating the process of annotation using a well-trained Sequence to Sequence Neural Network Model.

7.4 LIMITATIONS

System output are completely dependent on the data that is captured, hence when a new defect is encountered the training of model should be performed again. Since the dataset is so large the library will be little heavy to download and perform act

8. BIBLIOGRAPHY

8.1 BOOK REFERENCE

1. “A Grammar of Modern Tamil” by Thomas Lehman

8.2 WEB REFERENCE

1. https://universaldependencies.org/format.html

2. https://towardsdatascience.com/inter-annotator-agreement-2f46c6d37bf3

3. https://www.analyticsvidhya.com/blog/2021/06/natural-language-processing-sentiment-analysis-using-lstm/

4. https://www.nltk.org/

5. https://github.com/narVidhai/tamil-nlp-catalog

6. https://docs.streamlit.io/

9. APPENDICES

9.1 APPENDIX –A: PROJECT SCREEN SHOTS

Data Upload

****UPLOADING THE DATA TO THE SHEETS**

In [16]:
```
# SET THESE HYPER PARAMETERS

data_row_d = {'first':13755,'second':14309,'review':2995}
json_file_path = 'tamil-nlp-4f5946578e44.json'
```

In [17]:
```
sheet_editor = SheetEditor(json_file_path)
for stype in sheets_type:
    sheet = sheet_editor.client.open(gsheets_names[stype])
    sheet_instance = sheet.worksheet("sentences_to_be_annotated")
    sheet_editor.writeSheetFromIndex(sheet_instance=sheet_instance,data_df=data_df_d[stype], header_row=1,data_row=data_row_d[sty
    print(gsheets_names[stype],' uploaded with data..')
```

```
Team 1 - Real Dataset Annotation  uploaded with data..
Team 2 - Real Dataset Annotation  uploaded with data..
Gold Dataset Annotation - Next 5000 sents  uploaded with data..
```

Fig 9.1 Data Uploading in Corpus

Annotation

# sent_no = 154									
# sent_id = movie_2549_A72E58E6_3									
# text = இதனால் அந்த மந்திரியை ஏதாவது செய்ய வேண்டும் என்று இருவரும் ப்ளான் செய்கிறார்கள்.									
# text_en = Both plan to make the minister do something.									
# translit = itaṉāl anta mantiriyai ētāvatu ceyya vēṇṭum eṉṟu iruvarum plāṉ ceykiṟārkaḷ.									
# source = kaggle_sudalairajkumar_tamil-nlp_movie_reviews									
1	இதனால்	இதனால்	ADV	REA	_	5	advmod	_	_
2	அந்த	அந்த	PRON	DP	Animacy=Inan|Gender=Neut|Number=Sing|Person=3|Case=Nom|PronType=Dem	3	det	_	_
3	மந்திரியை	மந்திரி	NOUN	NOUN	Animacy=Hum|Gender=Masc|Number=Sing|Person=3|Case=Acc	5	nsubj	_	Entity=I_TITLE
4	ஏதாவது	ஏதாவது	ADJ	QAA	_	5	amod	_	_
5	செய்ய	செய்	VERB	TR	Gender=Neut|Animacy=Inan|Number=Sing|VerbForm=Inf|Tense=Fut|Polarity=Pos|Person=3	10	advcl	_	_
6	வேண்டும்	வேண்டு	AUX	AUX	Gender=Neut|Animacy=Inan|Number=Sing|VerbForm=Fin|Tense=Fut|Polarity=Pos|Person=3	5	aux	_	_
7	என்று	என்று	ADP	ADP	AdpType=Post	6	case	_	_
8	இருவரும்	இருவர்	PRON	PEP	Animacy=Hum|Gender=Com|Number=Dual|Person=3|Case=Nom|PronType=Prs	10	obj	_	Entity=I_CARDINAL
9	ப்ளான்	ப்ளான்	X	FW	Foreign=Yes	10	iobj	_	
10	செய்கிறார்கள்	செய்	VERB	TR	Gender=Com|Animacy=Hum|Number=Plur|VerbForm=Fin|Tense=Pres|Polarity=Pos|Person=3|Voice=Act|Mood=Ind	0	root	_	SpaceAfter=No
11	.	.	PUNCT	PUNCT	PunctType=Peri	10	punct	_	_

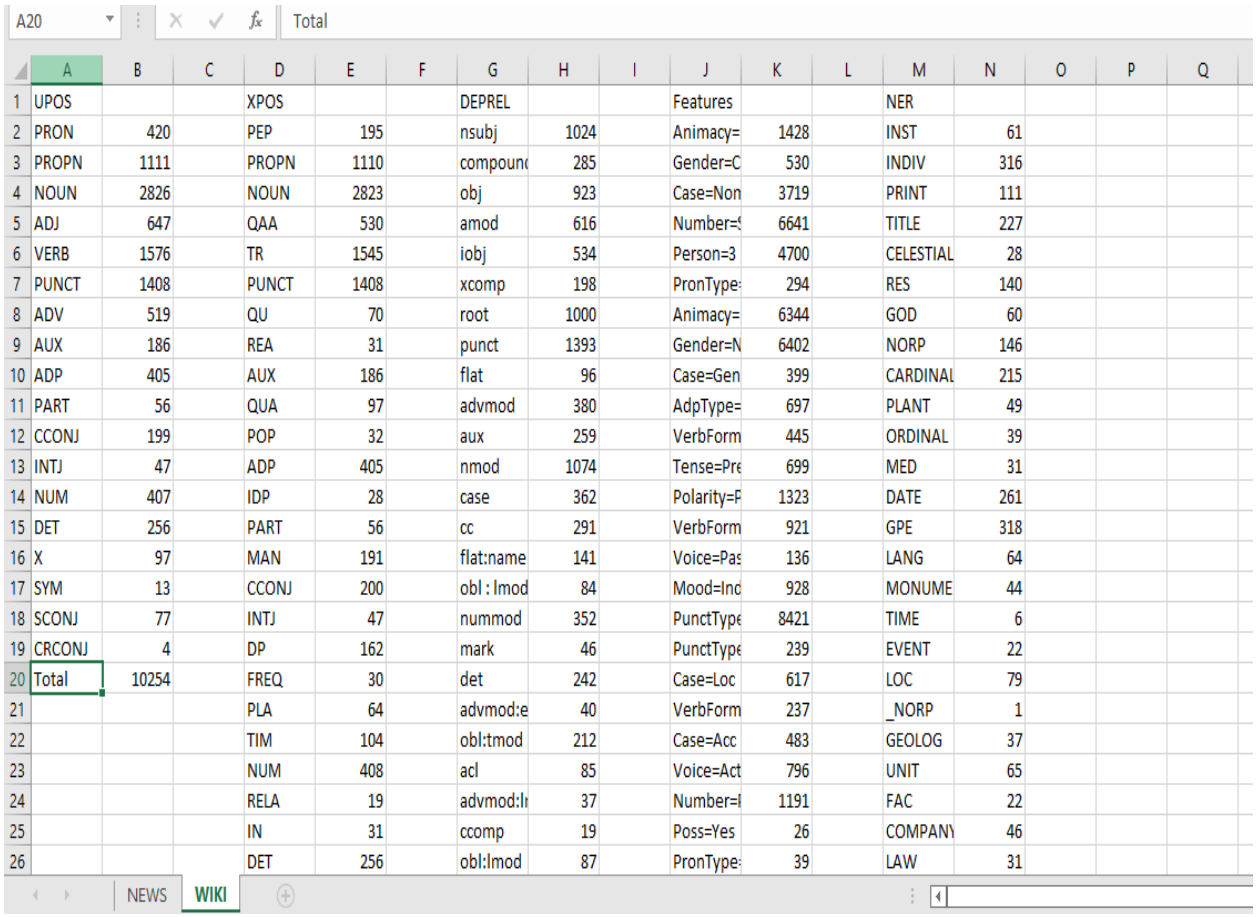
Fig 9.2 Sample Annotation in Corpus

GUI For Model

Fig 9.3 GUI Interface

9.2 APPENDIX –B: UNIT TESTING SCREEN SHOTS

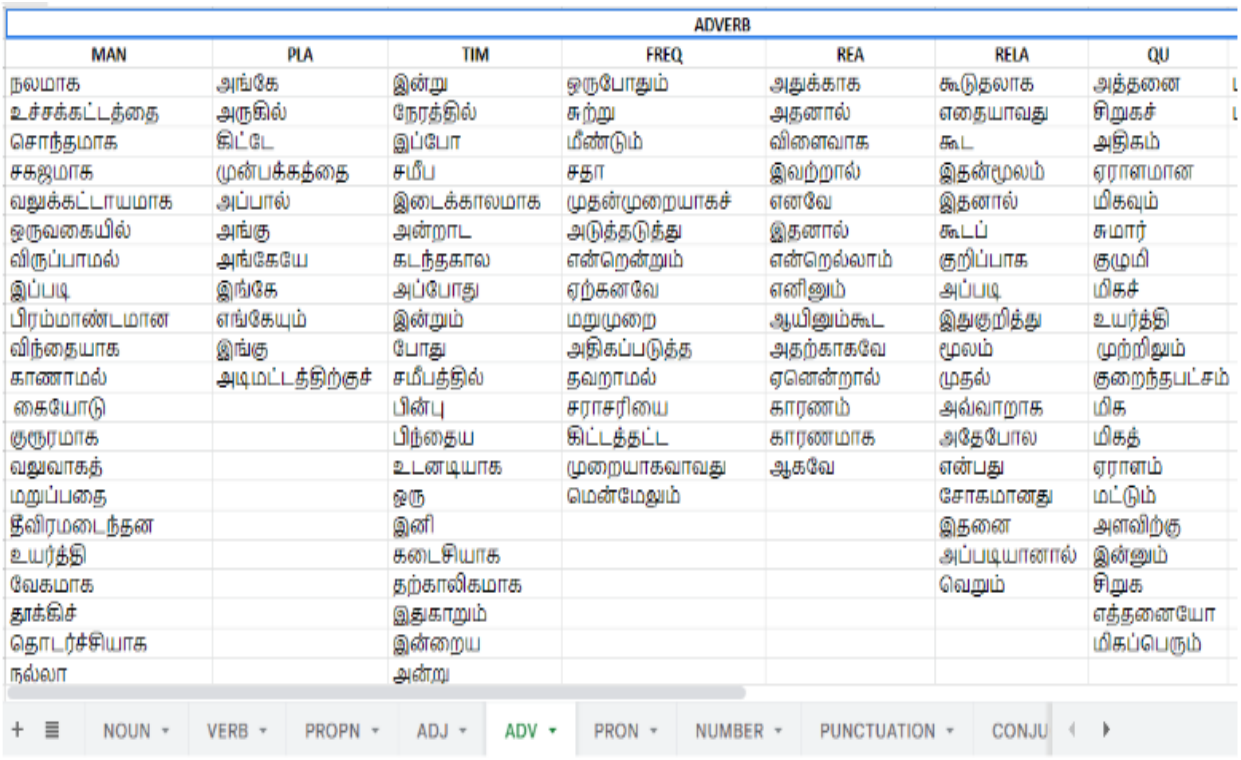
Statistical Analysis



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	UPOS			XPOS			DEPREL			Features			NER				
2	PRON	420		PEP	195		nsubj	1024		Animacy=	1428		INST	61			
3	PROPN	1111		PROPN	1110		compoun	285		Gender=C	530		INDIV	316			
4	NOUN	2826		NOUN	2823		obj	923		Case=Non	3719		PRINT	111			
5	ADJ	647		QAA	530		amod	616		Number=S	6641		TITLE	227			
6	VERB	1576		TR	1545		iobj	534		Person=3	4700		CELESTIAL	28			
7	PUNCT	1408		PUNCT	1408		xcomp	198		PronType=	294		RES	140			
8	ADV	519		QU	70		root	1000		Animacy=	6344		GOD	60			
9	AUX	186		REA	31		punct	1393		Gender=N	6402		NORP	146			
10	ADP	405		AUX	186		flat	96		Case=Gen	399		CARDINAL	215			
11	PART	56		QUA	97		advmod	380		AdpType=	697		PLANT	49			
12	CCONJ	199		POP	32		aux	259		VerbForm	445		ORDINAL	39			
13	INTJ	47		ADP	405		nmod	1074		Tense=Pre	699		MED	31			
14	NUM	407		IDP	28		case	362		Polarity=P	1323		DATE	261			
15	DET	256		PART	56		cc	291		VerbForm	921		GPE	318			
16	X	97		MAN	191		flat:name	141		Voice=Pas	136		LANG	64			
17	SYM	13		CCONJ	200		obl : lmod	84		Mood=Ind	928		MONUME	44			
18	SCONJ	77		INTJ	47		nummod	352		PunctType	8421		TIME	6			
19	CRCONJ	4		DP	162		mark	46		PunctType	239		EVENT	22			
20	Total	10254		FREQ	30		det	242		Case=Loc	617		LOC	79			
21				PLA	64		advmod:e	40		VerbForm	237		_NORP	1			
22				TIM	104		obl:tmod	212		Case=Acc	483		GEOLOG	37			
23				NUM	408		acl	85		Voice=Act	796		UNIT	65			
24				RELA	19		advmod:l	37		Number=I	1191		FAC	22			
25				IN	31		ccomp	19		Poss=Yes	26		COMPANY	46			
26				DET	256		obl:lmod	87		PronType=	39		LAW	31			

Fig 9.4 Corpus Statistical Analysis

Checking Corpus Correctness



ADVERB						
MAN	PLA	TIM	FREQ	REA	RELA	QU
நலமாக	அங்கே	இன்று	ஒருபோதும்	அதுக்காக	கூடுதலாக	அத்தனை
உச்சக்கட்டத்தை	அருகில்	நேரத்தில்	சுற்று	அதனால்	எதையாவது	சிறுகச்
சொந்தமாக	கிட்டே	இப்போ	மீண்டும்	விளைவாக	கூட	அதிகம்
சகஜமாக	முன்பக்கத்தை	சமீப	சதா	இவற்றால்	இதன்மூலம்	ஏராளமான
வலுக்கட்டாயமாக	அப்பால்	இடைக்காலமாக	முதன்முறையாகச்	எனவே	இதனால்	மிகவும்
ஒருவகையில்	அங்கு	அன்றாட	அடுத்தடுத்து	இதனால்	கூடப்	சுமார்
விருப்பாமல்	அங்கேயே	கடந்தகால	என்றென்றும்	என்றெல்லாம்	குறிப்பாக	குமுமி
இப்படி	இங்கே	அப்போது	ஏற்கனவே	எனினும்	அப்படி	மிகச்
பிரம்மாண்டமான	எங்கேயும்	இன்றும்	மறுமுறை	ஆயினும்கூட	இதுகுறித்து	உயர்த்தி
விந்தையாக	இங்கு	போது	அதிகப்படுத்த	அதற்காகவே	மூலம்	முற்றிலும்
காணாமல்	அடிமட்டத்திற்குச்	சமீபத்தில்	தவறாமல்	ஏனென்றால்	முதல்	குறைந்தபட்சம்
கையோடு		பின்பு	சராசரியை	காரணம்	அவ்வாறாக	மிக
குரூரமாக		பிந்தைய	கிட்டத்தட்ட	காரணமாக	அதேபோல	மிகத்
வலுவாகத்		உடனடியாக	முறையாகவாவது	ஆகவே	என்பது	ஏராளம்
மறுப்பதை		ஒரு	மென்மேலும்		சோகமானது	மட்டும்
தீவிரமடைந்தன		இனி			இதனை	அளவிற்கு
உயர்த்தி		கடைசியாக			அப்படியானால்	இன்னும்
வேகமாக		தற்காலிகமாக			வெறும்	சிறுக
தூக்கிச்		இதுகாறும்				எத்தனையோ
தொடர்ச்சியாக		இன்றைய				மிகப்பெரும்
நல்லா		அன்று				

NOUN | VERB | PROPN | ADJ | ADV | PRON | NUMBER | PUNCTUATION | CONJU

Fig 9.5 List of Word with its POS tags

Model Performance

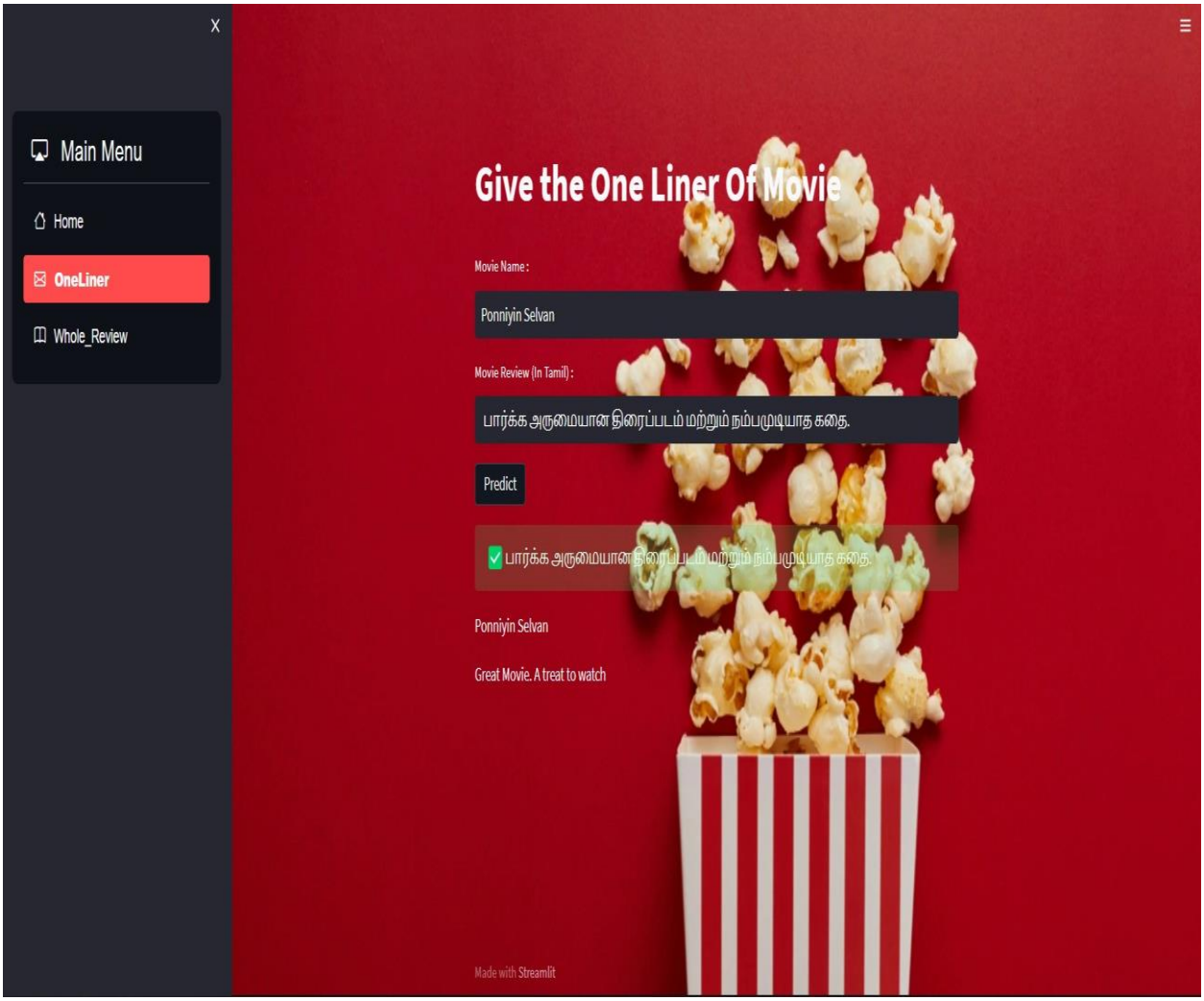


Fig 9.6 checking model for positive sentence

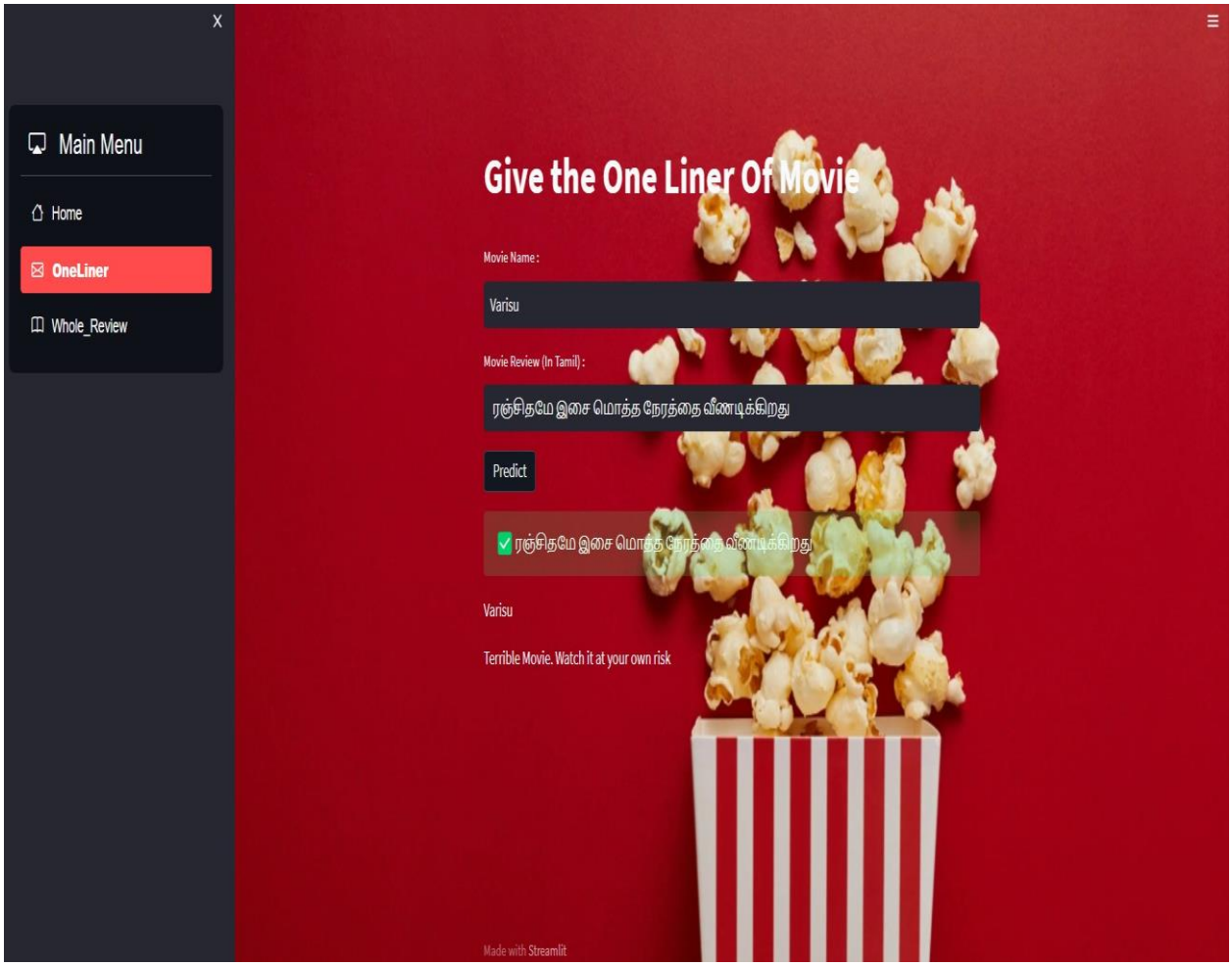


Fig 9.7 checking model for Negative sentence

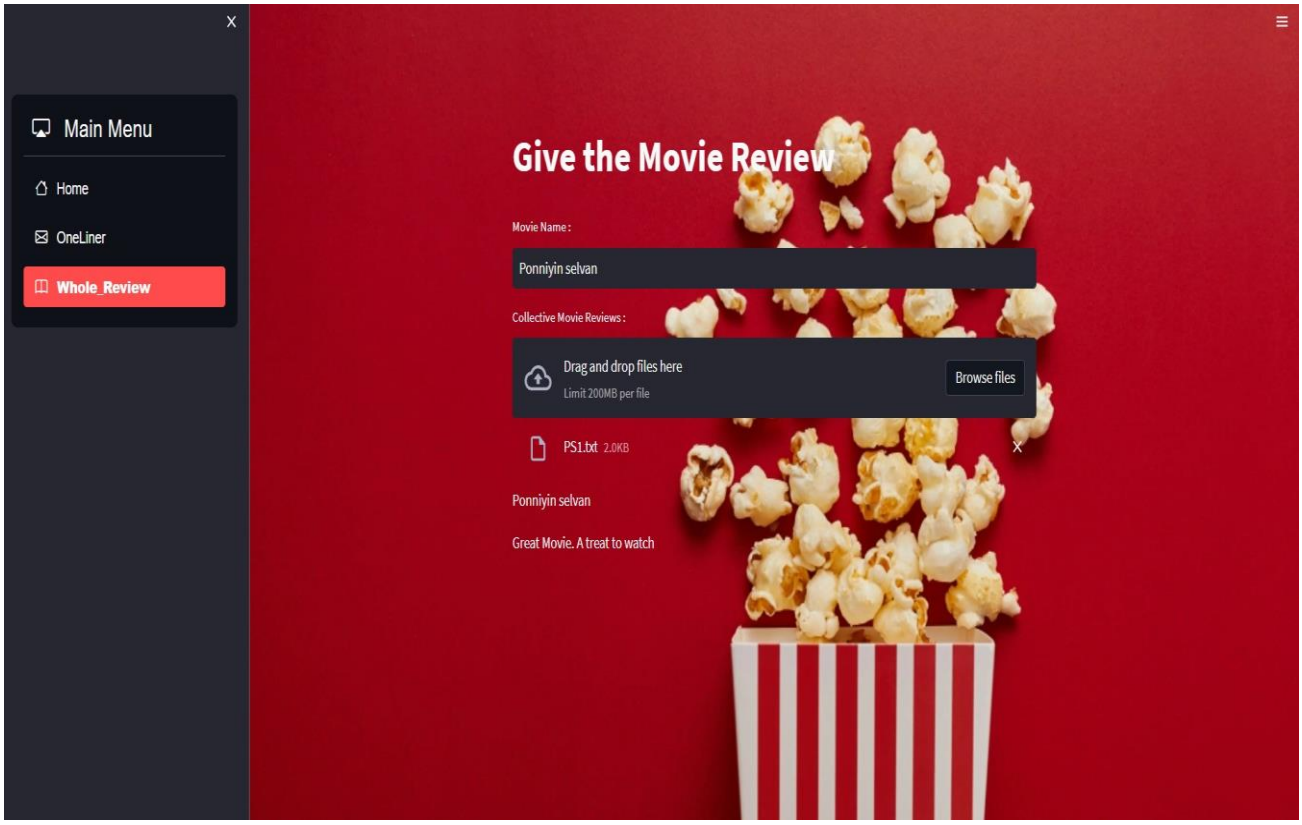


Fig 9.8 Sentiment Analysis on Collection of Sentence

9.3 APPENDIX—C: INTEGRATION TESTING

IAA Score

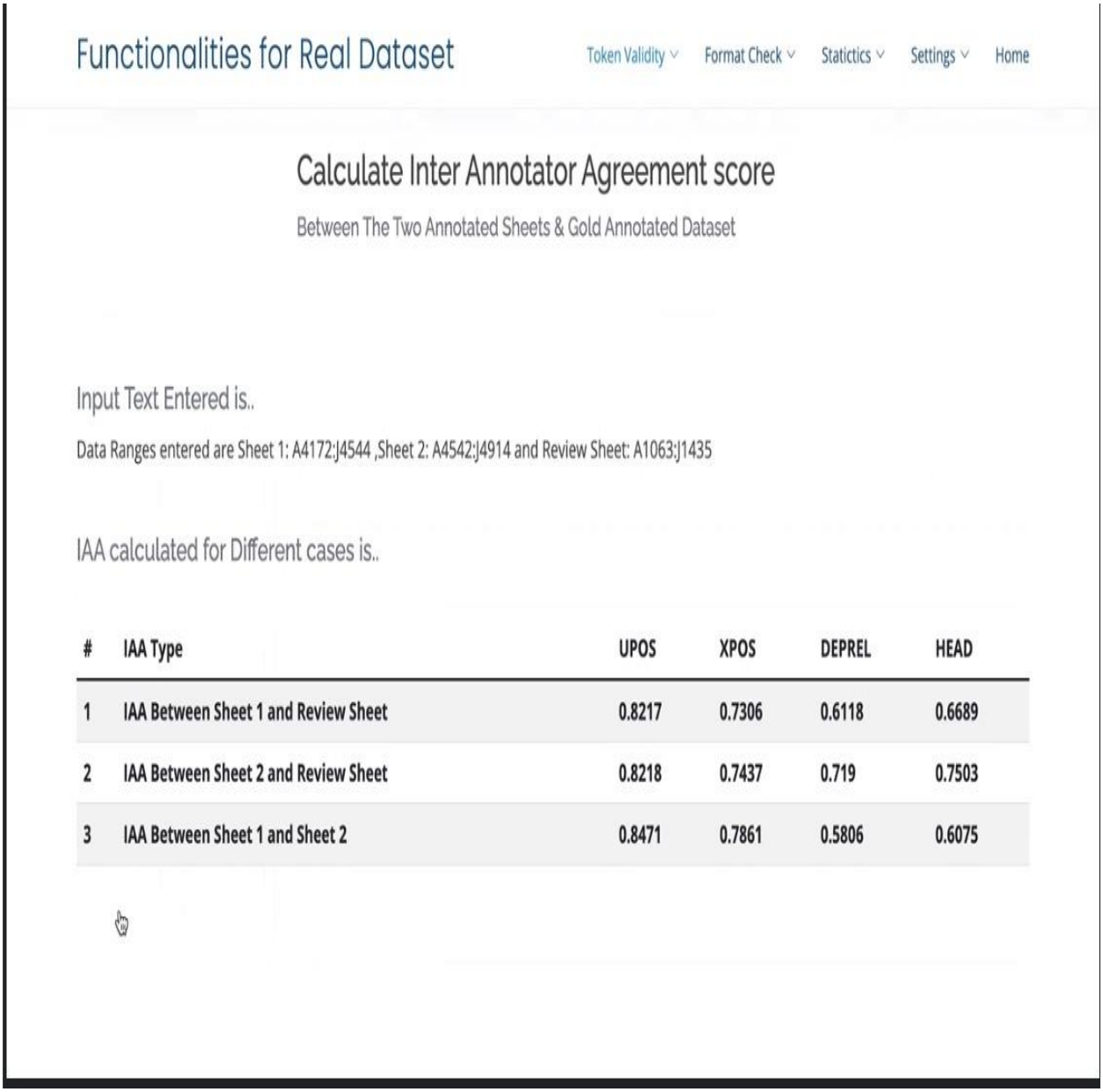


Fig 9.9 IAA Score Before reviewing

Input Text Entered is..

Data Ranges entered are Sheet 1: A4172:J4544 ,Sheet 2: A4542:J4914 and Review Sheet: A1063:J1435

IAA calculated for Different cases is..

#	IAA Type	UPOS	XPOS	DEPREL	HEAD
1	IAA Between Sheet 1 and Review Sheet	0.8411	0.7498	0.6118	0.6689
2	IAA Between Sheet 2 and Review Sheet	0.8413	0.7629	0.7311	0.744
3	IAA Between Sheet 1 and Sheet 2	0.8471	0.7861	0.5986	0.6136

Fig 9.10 IAA Score After review