



# Enhancing Conversational AI Model Performance and Explainability for Sinhala-English Bilingual Speakers

2022-056

Progress Presentation - I

# The Team



**Supervisor**  
Dr. Lakmini Abeywardhana



**Co – Supervisor**  
Ms. Dinuka Wijendra



Dissanayake D.M.I.M.  
IT19069432



Hameed M.S.  
IT19064932



Jayasinghe D.T.  
IT19075754



Sakalasooriya S.A.H.A.  
IT19051208

# Presentation Overview

1

Overall Research  
Progress

2

Individual Research  
Component Progress

3

Commercialization  
Plan

POST <http://localhost:5005/model/parse>

Params Authorization Headers (8) **Body** Pre-request Script Tests

none form-data x-www-form-urlencoded raw binary Graph

```

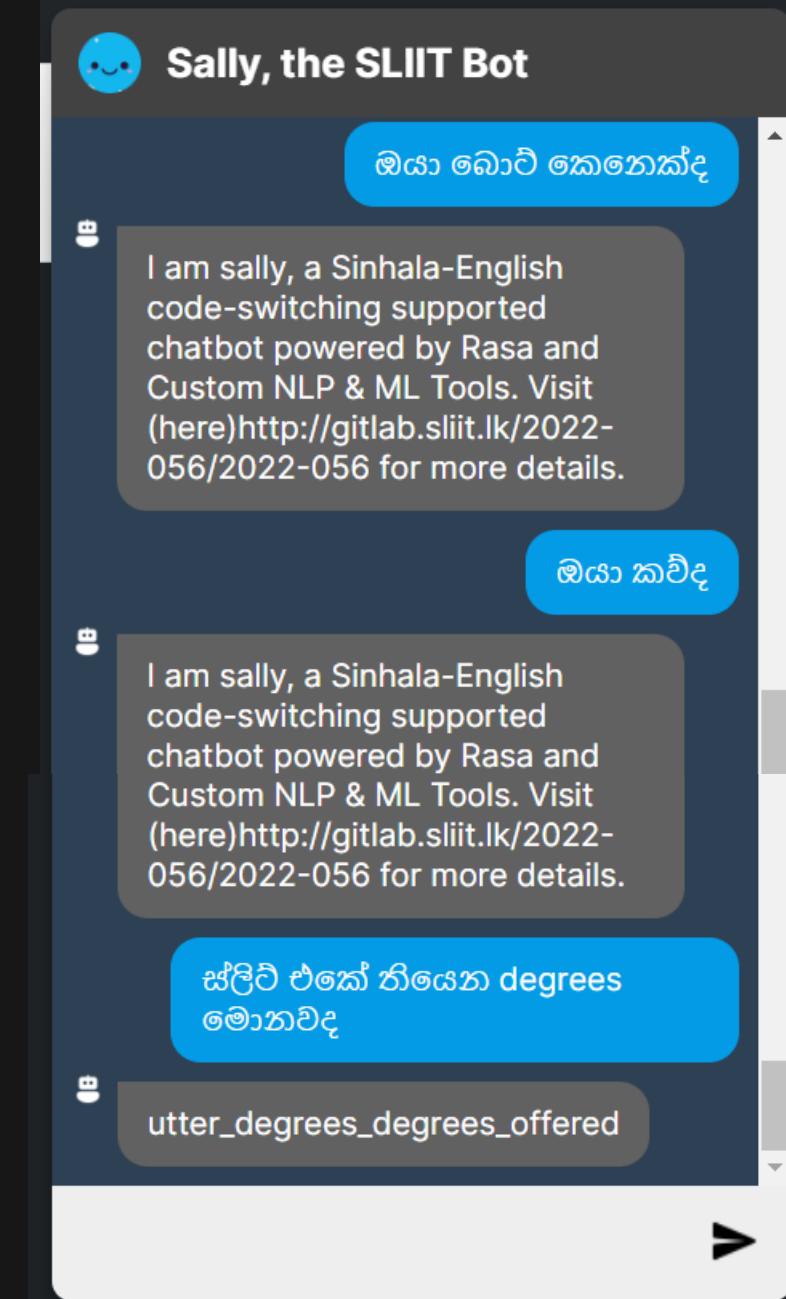
1 {
2   "text": "ඔයා කුමතිම කුම මොකක්ද"
3 }
```

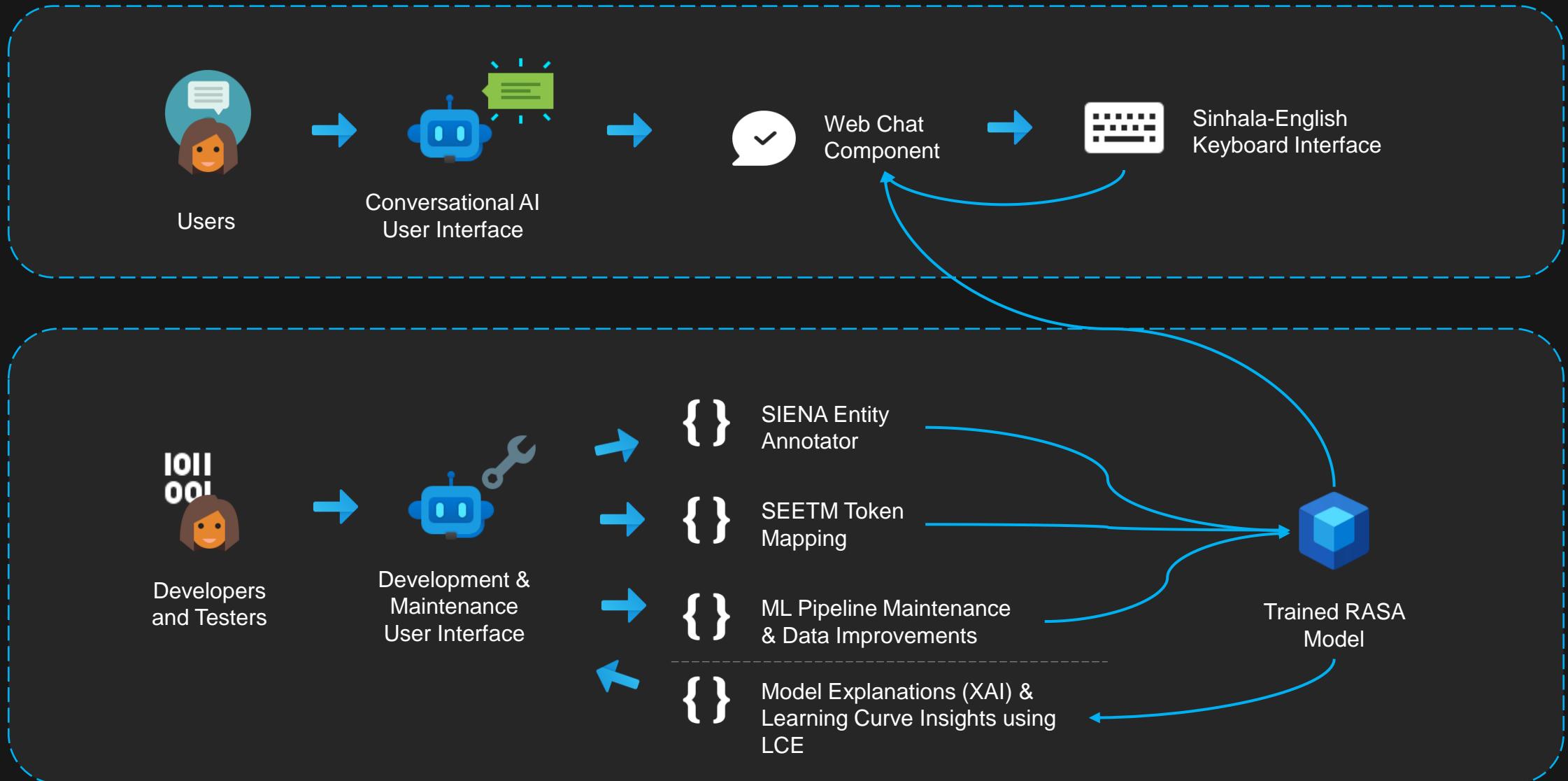
**Body** Cookies Headers (7) Test Results

Pretty Raw Preview Visualize JSON

```

1 {
2   "text": "ඔයා කුමතිම කුම මොකක්ද",
3   "intent": {
4     "id": 908698919044720841,
5     "name": "oos_food_preferences",
6     "confidence": 0.7912442088127136
7   },
8   "entities": [],
9   "intent_ranking": [
10    {
11      "id": 908698919044720841,
12      "name": "oos_food_preferences",
13      "confidence": 0.7912442088127136
14    },
15    {
16      "id": -8043840511142879659,
17      "name": "academic_scholarships",
```







# Technologies

## Backend Development



Python



TensorBoard



Rasa



pandas



spaCy



Flask



Gensim

## Frontend Development



React



Socket.IO



Docker



Nginx



GCP



Caddy

## NoSQL Database Development



MongoDB



Git



Teams

## Source-code and Project management

67%

# SIENA: Annotating entities using reverse-stemming & other techniques

to develop a data annotation tool for code-mixed text data for efficient  
custom entity tagging.

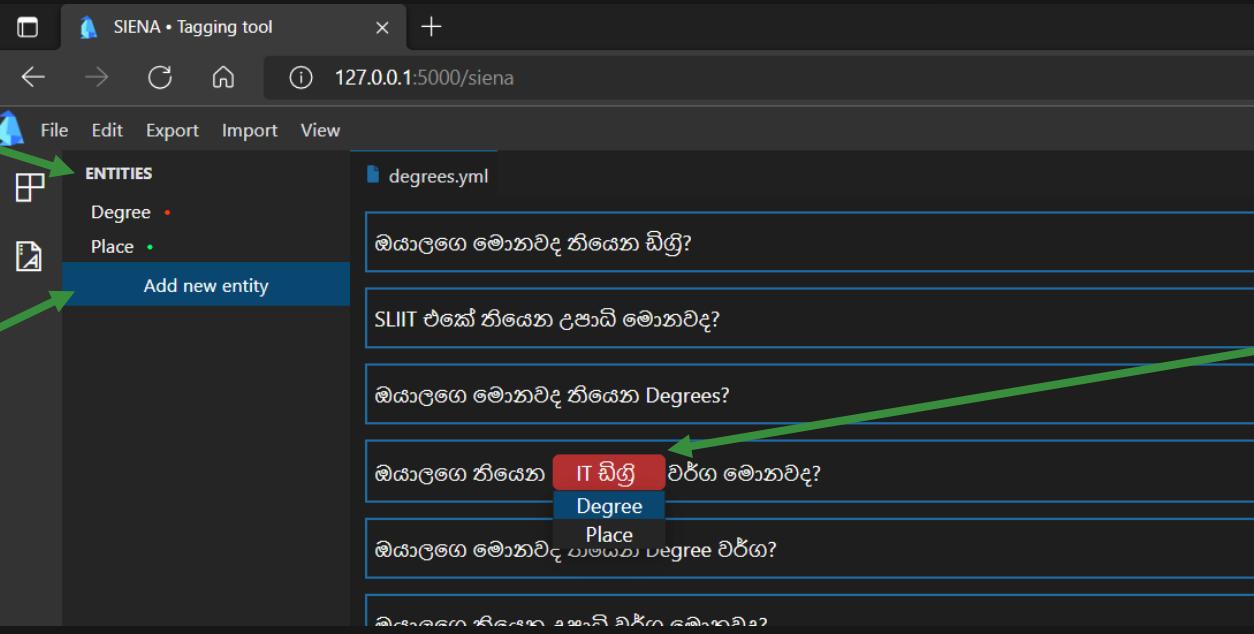


Sakalasooriya S.A.H.A.  
IT19051208  
Data Science

# Research Question

Why custom name entity tagging is very time consuming? 

Any solution?



The screenshot shows the SIENA Tagging tool interface. On the left, there's a sidebar with 'Edit', 'Export', and 'Import' buttons, and sections for 'ENTITIES', 'Degree', and 'Place'. A green arrow points from the 'Add new entity' button in the 'Place' section to the 'Add new entity' button in the main content area. Another green arrow points from the 'Degree' section in the sidebar to the 'Degree' section in the main content area. In the main content area, there's a search bar with Sinhala text 'ඒයාලගේ මොනවද තියෙන සිරුළු?' followed by a list of search results:

- එල්ලක් තියෙන උපාධි මොනවද?
- ඒයාලගේ මොනවද තියෙන Degrees?
- ඒයාලගේ තියෙන IT සිරුළු වර්ග මොනවද?
- ඒයාලගේ මොනවද යාපෘති පර්‍යාග වර්ග?

A red box highlights the 'IT සිරුළු' button in the third result. A green arrow points from this result to a circular callout on the right containing the Sinhala text 'IT සිරුළු' and its English equivalents 'Degree' and 'Place'.

# Functional Requirements



Recommend name entities



Import corpus into SINEA



Export annotated text from SIENA



Import portable knowledge base into SINEA



Export portable knowledge base from SINEA



# Non-functional Requirements



Able to handle large text corpus



Able to easily maintain



Easily install



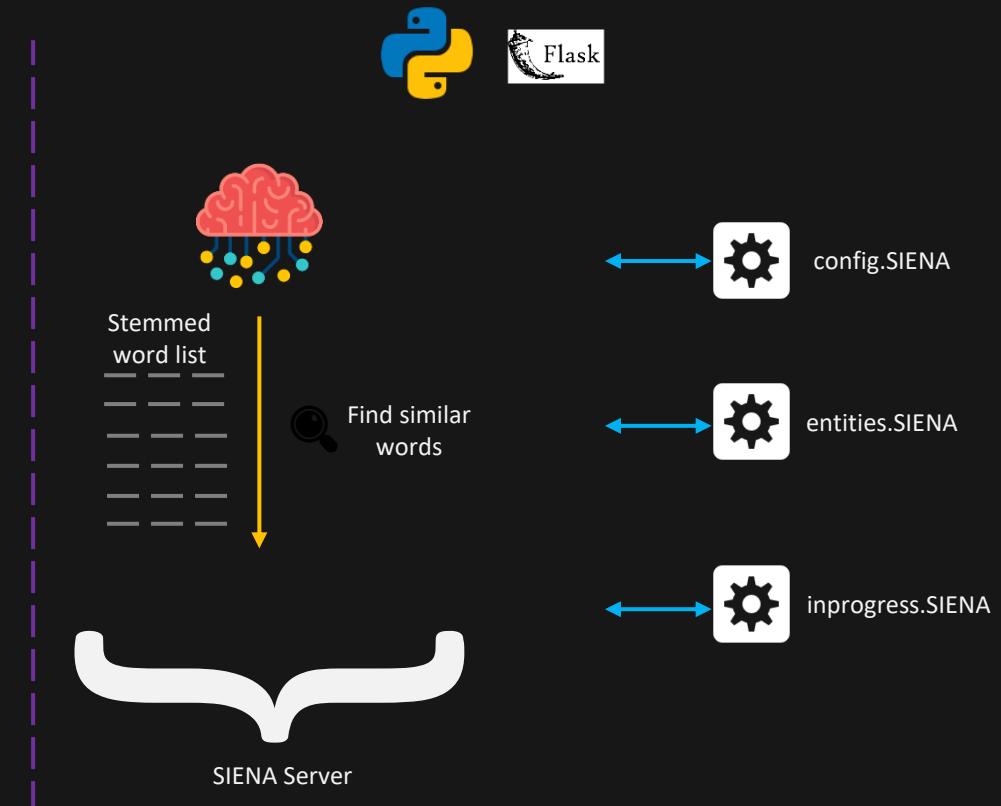
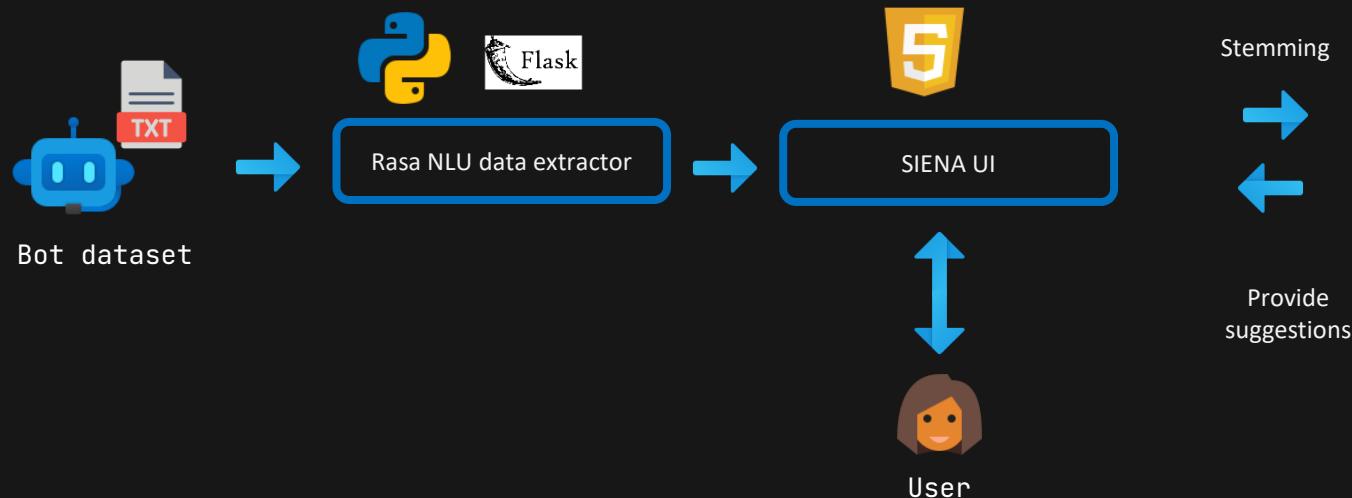
SIENA should be reliable



SIENA should be secure

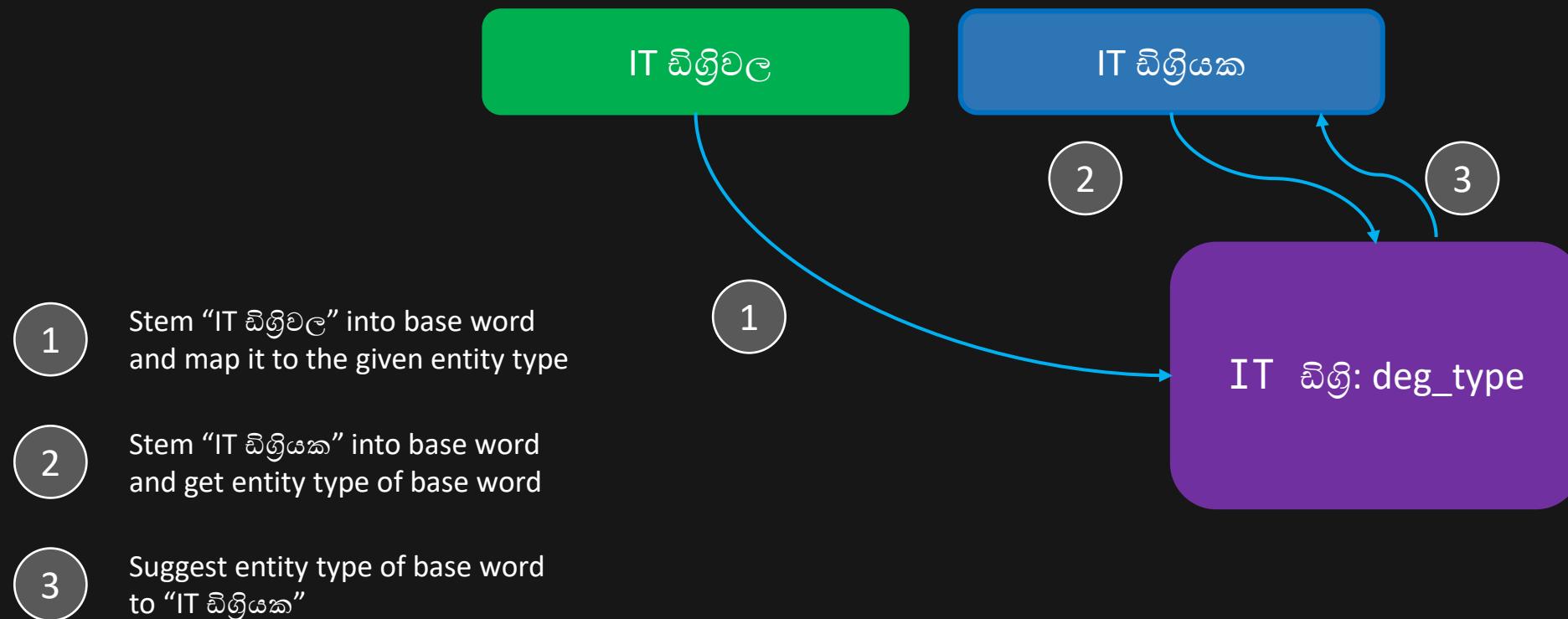
The screenshot shows the PyPI project page for 'siena'. The title is 'siena 0.0.1a1'. Below it is a command line interface (CLI) snippet: `pip install -i https://test.pypi.org/simple/ siena`. To the right, there's a green button labeled 'Latest version' with a checkmark. At the bottom right, it says 'Released: May 22, 2022'. On the left, there's a sidebar with navigation links: 'Project description' (which is highlighted in blue), 'Release history', 'Download files', and 'Project links'. The main content area has a heading 'SIENA tool.' and a 'Project description' section with the text: 'Code-Mixed Conversational AI - NLP Research | Custom Entity Recognition'. At the bottom, it says '@Author: Sakalasooriya S.A.H.A.'

# Solution Flow



# Methodology

## Reverse stemming



# Lemmatization vs stemming

## Stemming

Stemming is the process of producing morphological variants of a base word

## Lemmatization

In contrast to stemming, lemmatization looks beyond word reduction and considers a language's full vocabulary to apply a morphological analysis to word

# Finding relationship between slightly different base forms

```
1 sentence_list = ["IT ඩිග්‍රීයේ","IT ඩිග්‍රීයෝ","IT ඩිග්‍රීයකු","IT ඩිග්රී"]  
2 for selection in sentence_list:  
3     en_stemmed = en_stemmer_sentence(selection)  
4     si_stemmed = si_stemmer_sentence_custom(en_stemmed)  
5     #si_stemmed = si_stemmer_sentence(en_stemmed)  
6     print(si_stemmed)
```

```
IT ඩිග්‍රීර්දය්  
IT ඩිග්‍රීර්දය්  
IT ඩිග්‍රීර්දය්  
IT ඩිග්රී
```

Stemmed  
word list

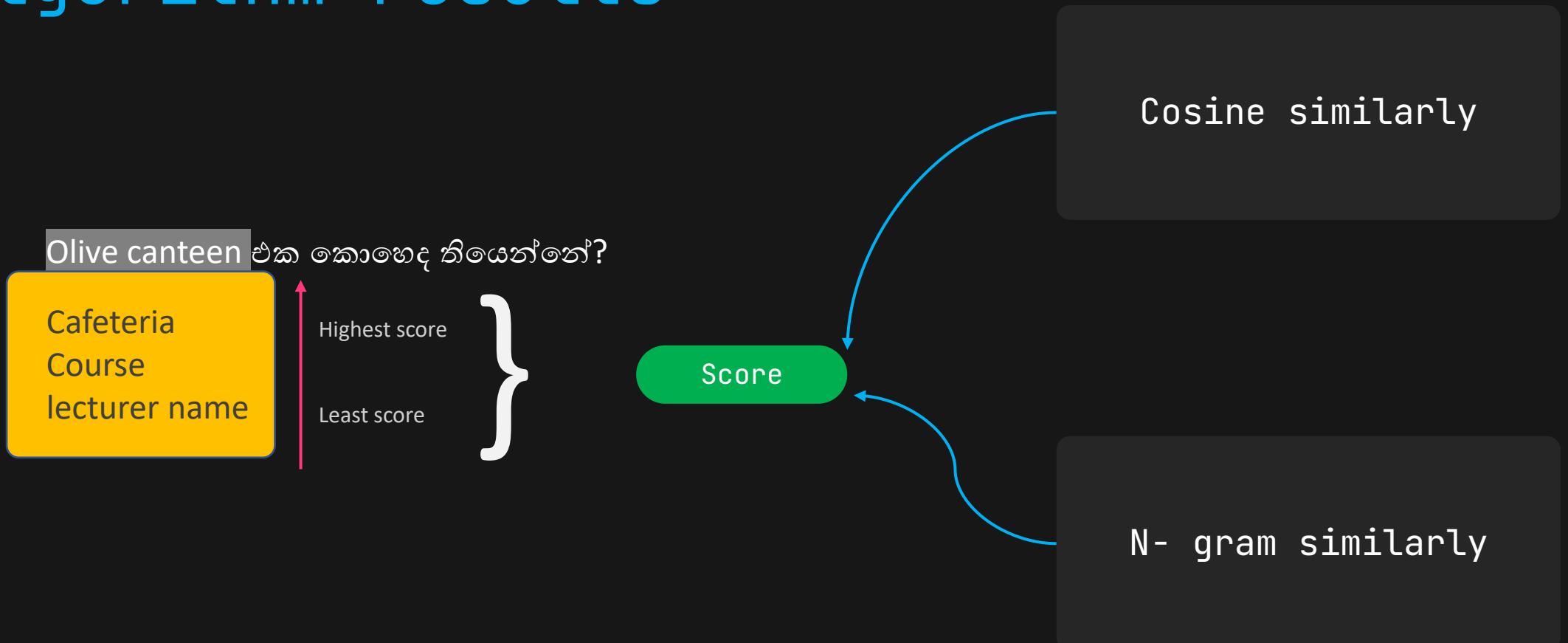


Find similar  
words

Slightly different base  
forms due to:

- Spelling mistakes
- Different Singlish typing patterns
- Limitations of stemming algorithm

# Order recommendations according to the algorithm results



# Approach 1

## Finding relationship between slightly different base forms

```
1 words = ["IT සිංහලුවා", "IT සිංහල්"]  
2 #counting chars  
3 result = []  
4 for word in words:  
5     wordVec = []  
6     for letter in vecSpace:  
7         wordVec.append(word.count(letter))  
8     squares=list(map(lambda x:pow(x,2),wordVec))  
9     sum_of_sq = sum(squares)  
10    norm = math.sqrt(sum_of_sq)  
11    norm_vec = [i/norm for i in wordVec]  
12    result.append(norm_vec)  
13  
14 result = np.array(result)
```

```
[55] 1 df = pd.DataFrame()  
2 i = 0  
3 for singleVec_R in result:  
4     col = []  
5     for singleVec_C in result:  
6         col.append(np.sum(singleVec_R*singleVec_C))  
7     df[i] = col  
8     i += 1  
9 print("Similarity =",df[0][1])
```

```
Similarity = 0.992573153576836
```



Cosine similarly approach

## Approach 2

# Finding relationship between slightly different base forms

```
[34] 1 word_1 = generate_ngrams("IT සිංහල", 2)
2 word_2 = generate_ngrams("IT සිංහල්තුය", 2)
3 long_word = ""
4 short_word = ""
5 count = 0
6 ans = 0
7 if len(word_1) > len(word_2):
8     long_word = word_1
9     short_word = word_2
10 else:
11     long_word = word_2
12     short_word = word_1
13 try:
14     for element in short_word:
15         if element in long_word:
16             count+=1
17     ans = count*100/(len(long_word))
18 except Exception as e:
19     pass
20
21 print("Similarity =",ans)
```

Similarity = 75.0



N-gram similarly approach

# Sub

# Objectives

Overall completion = objective completion \* objective weight

70% \* 0.4

Define the  
Recommendation  
hierarchy (algorithms)

70% \* 0.2

Make SIENA Compatible  
with Frameworks

10% \* 0.1

Make Knowledge base as  
Moduler Component

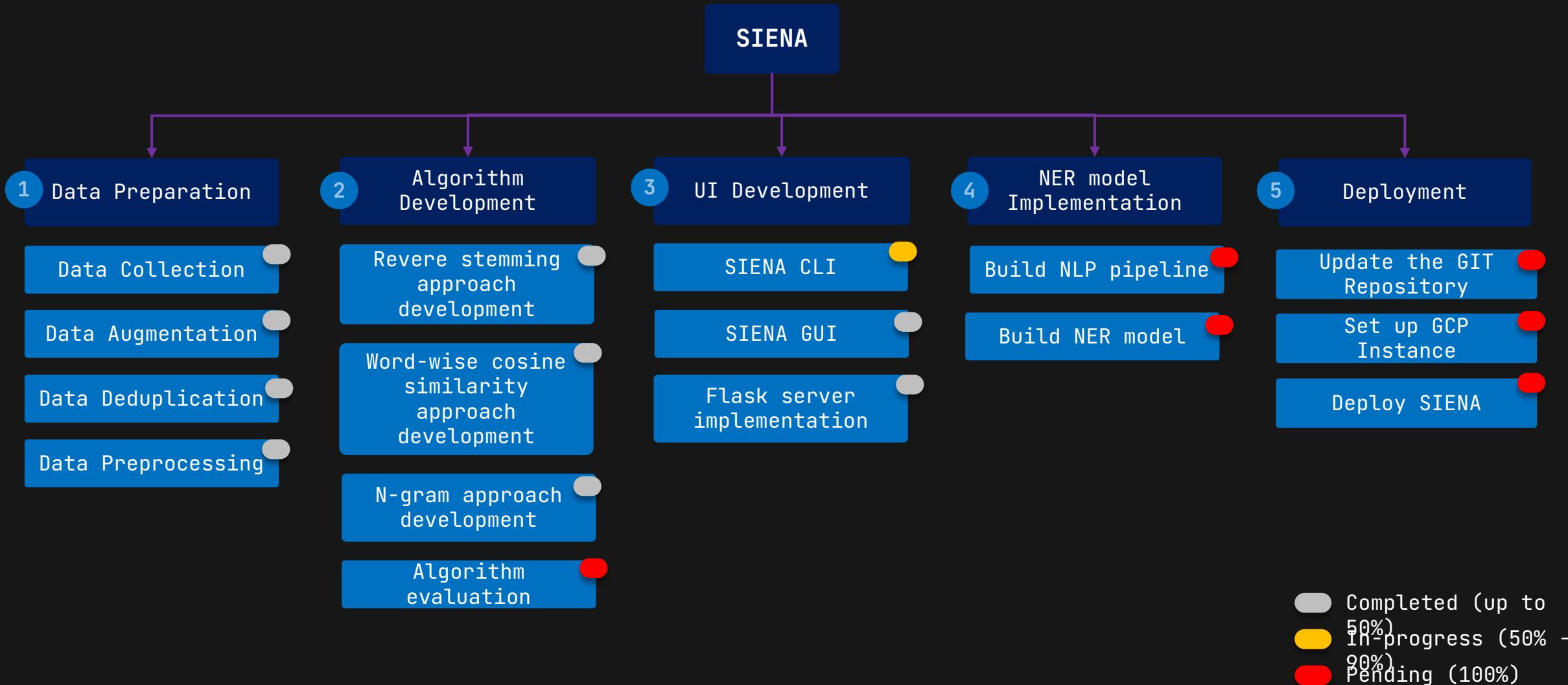
80% \* 0.3

Develop Visualizations  
Technique to Provide  
User Friendly  
Suggestions

\* Overall completion rate:

67%

# Work Breakdown Structure





# Technologies and best practices

Backend Development



Python



pandas



Rasa



Flask

Frontend Development



HTML

Project Management



GitLab  
GitHub

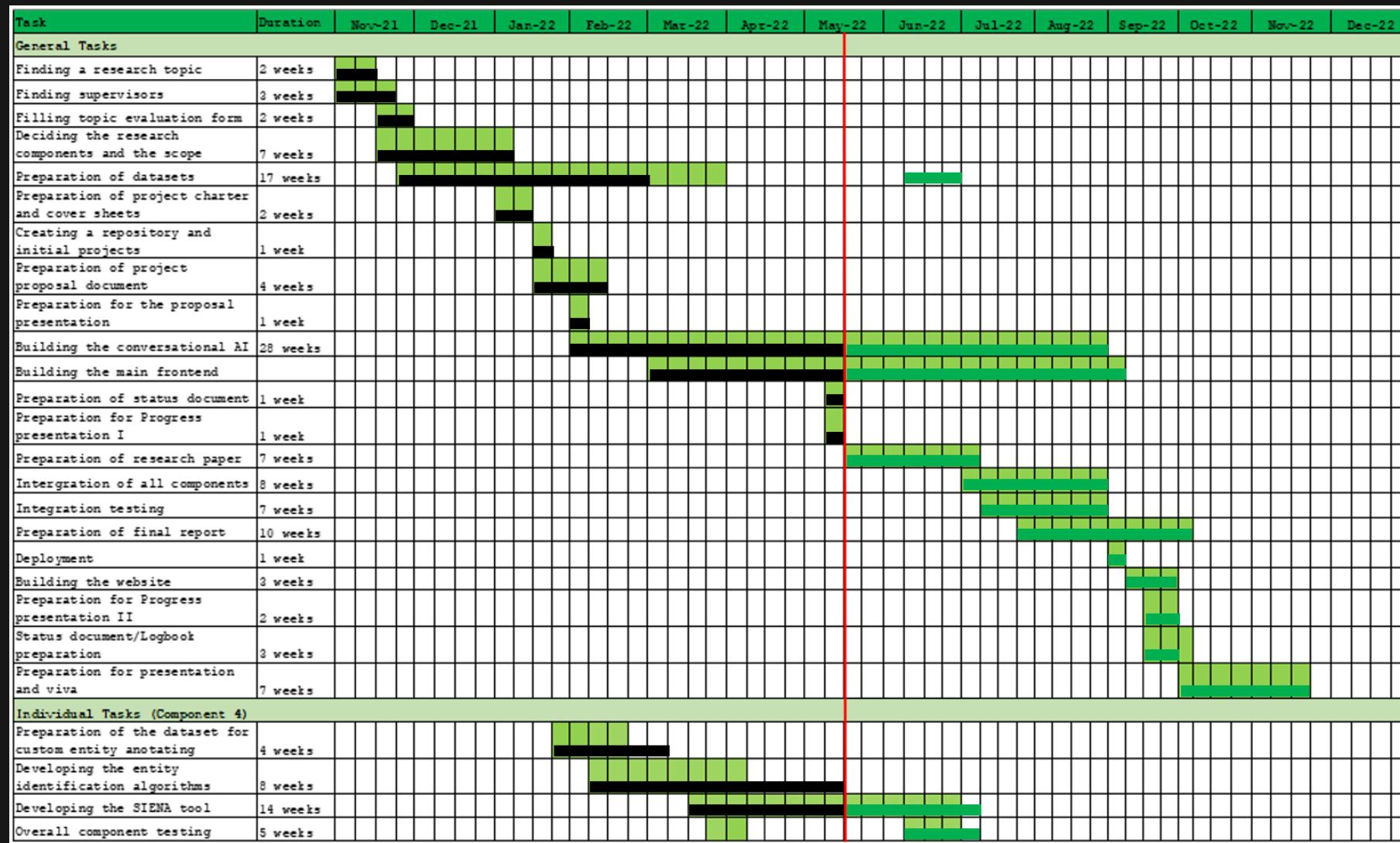


PyPI

Test PyPI

.SIENA files to save configuration data

# Gantt chart



# References

- [1] Anastasia Zhukova, Felix Hamborg, Bela Gipp, 'ANEA: Automated (Named) Entity Annotation for German Domain-Specific Texts' Available: <https://arxiv.org/pdf/2112.06724.pdf>
- [2] Pontus Stenetorp, Sampo Pyysalo, Goran Topic, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii, 'BRAT: a Web-based Tool for NLP-Assisted Text Annotation' Available: <https://aclanthology.org/E12-2021.pdf>
- [3] Kalina Bontcheva, Hamish Cunningham, Ian Roberts, Angus Roberts, Valentin Tablan, Niraj Aswani, Genevieve Gorrell, 'GATE Teamware: a web-based, collaborative text annotation framework', Available: <https://www.jstor.org/stable/42636386>
- [4] Jie Yang, Yue Zhang, Linwei Li, Xingxuan Li, 'YEDDA: A Lightweight Collaborative Text Span Annotation Tool', Available: <https://aclanthology.org/P18-4006.pdf>
- [5] J.B Dissanayake, Basaka mahima, ISBN: 9789556963656
- [6] "Spacy Styleguide",  
<https://spacy.io/styleguide>
- [7] "Spacy Data formats · spaCy API Documentation",  
<https://spacy.io/api/data-formats>
- [8] "Vector Icons and Stickers - PNG, SVG, EPS, PSD and CSS",  
<https://www.flaticon.com/>

54%

# Code-less Maintenance and Model Performance Evaluation

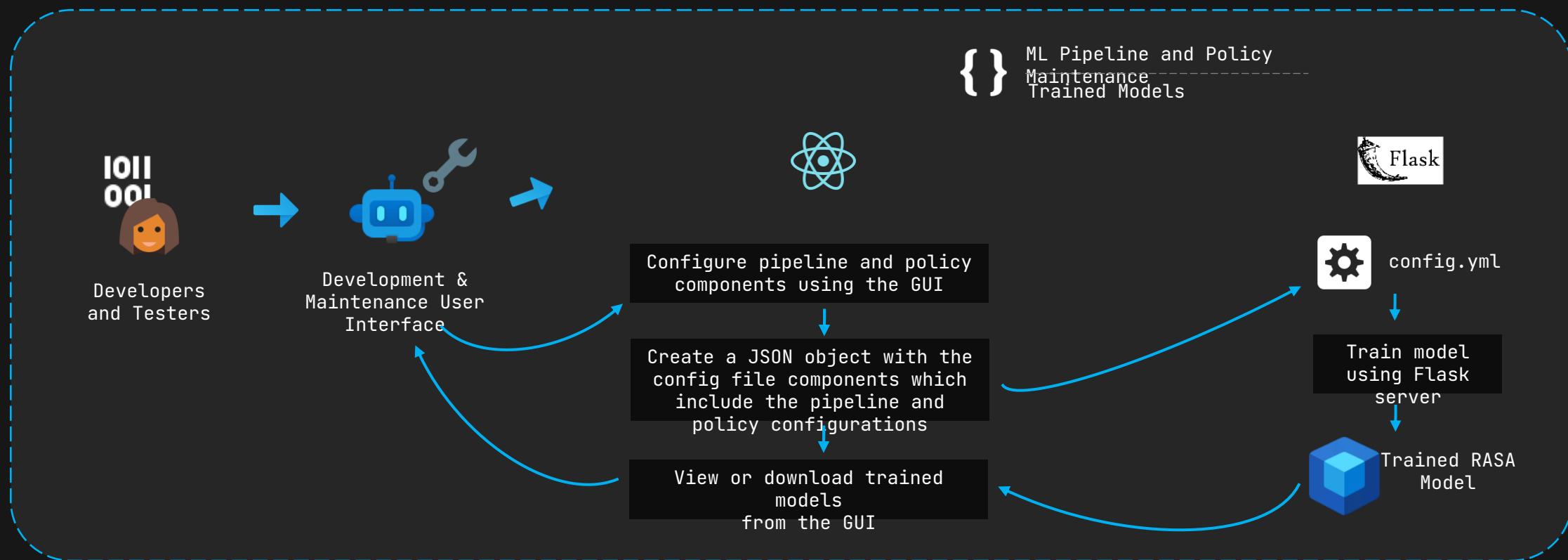
Enabling non-machine learning experts to effectively improve and evaluate conversational AI machine learning models



Hameed M.S.  
IT19064932  
Software Engineering



# Implemented Research Component Flow





# Data Improvement Existing Solution



# Rasa X Solution

The screenshot shows the Rasa X Configuration interface. On the left is a dark sidebar with icons for dashboard, logs, policies, stories, entities, and a message center. The main area is titled "Configuration" and contains a text editor. The text editor displays the following YAML code:

```
language: en
pipeline: supervised_embeddings
policies:
  - name: MemoizationPolicy
  - name: KerasPolicy
  - name: MappingPolicy
```

Below the code is a "Save" button. At the bottom left of the main area is a small circular icon with a white 'M'.



# Data Improvement Proposed Solution

## Modify Rasa NLU Configurations



Select an Intent Classifier:

DIET Classifier



Advanced Component Configurations:

Epochs

Ranking Length

Train

Window Help rasatest2.8.12 - config.yml

test-diet-exp.py nlu.yml domain.yml config.yml

```
language: en
pipeline:
  - name: WhitespaceTokenizer
  - name: RegexFeaturizer
  - name: LexicalSyntacticFeaturizer
  - name: CountVectorsFeaturizer
  - name: CountVectorsFeaturizer
    analyzer: char_wb
    min_ngram: 1
    max_ngram: 4
  - name: DIETClassifier
    ranking_length: 0
    epochs: 100
    constrain_similarities: true
  - name: EntitySynonymMapper
  - name: ResponseSelector
    epochs: 100
    constrain_similarities: true
  - name: FallbackClassifier
    threshold: 0.3
    ambiguity_threshold: 0.1
```



# Data Improvement Implemented Solution

## Pipeline Components

Language Models

- SpacyNLP

Tokenizers

- WhitespaceTokenizer
- SpacyTokenizer

Featurizers

- SpacyFeaturizer
- RegexFeaturizer
- CountVectorsFeaturizer
- LexicalSyntacticFeaturizer

Classifiers

- Keyword
- DIETClassifier
- Fallback

Extractors

- SpacyEntityExtractor
- CRFEntityExtractor
- RegexExtractor
- EntitySynonymMapper

### WhitespaceTokenizer

Intent Tokenization Flag:  False  Intent Split Symbol:

Token Pattern:  None

### RegexFeaturizer

Case Sensitive:  True  Use Word Boundaries:  True

Number of Additional Patterns:

### Policies

Policies

- TEDPolicy
- UnexpectTEDIntentPolicy
- MemoizationPolicy
- AugmentedMemoizationPolicy
- RulePolicy

#### UnexpectTEDIntentPolicy

Epochs:

Max History:

#### MemoizationPolicy

Max History:

Train Model

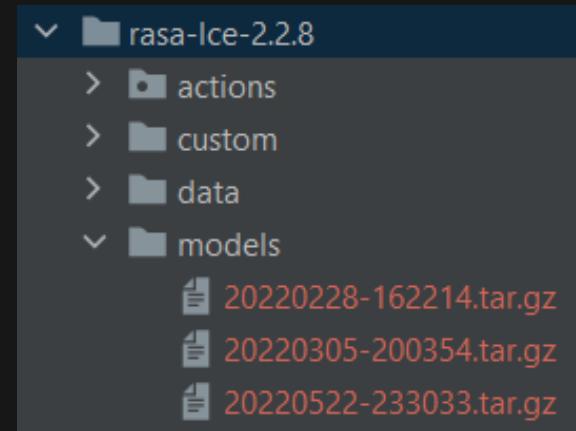
```

language: en
pipeline:
  - name: WhitespaceTokenizer
    intent_tokenization_flag: false
    intent_split_symbol: -
    token_pattern: ' '
  - name: RegexFeaturizer
    case_sensitive: true
    use_word_boundaries: true
  - name: CountVectorsFeaturizer
    analyzer: word
    min_ngram: 1
    max_ngram: 1
    OOV_token: None
    use_shared_vocab: false
    additional_vocabulary_size:
      text: 1000
      response: 1000
      action_text: 1000
  - name: DIETClassifier
    epochs: 30
    entity_recognition: true
    intent_classification: true
policies:
  - name: UnexpectTEDIntentPolicy
    epochs: 20
    max_history: 8
  - name: MemoizationPolicy
    max_history: 3

```



# Trained Models



SALLY

View all your trained models here!

Model Status	Model Name	Actions
● (Red)	20220228-162214.tar.gz	<a href="#">Download Model</a> <a href="#">Delete Model</a>
● (Red)	20220305-200354.tar.gz	<a href="#">Download Model</a> <a href="#">Delete Model</a>
● (Green)	20220522-233033.tar.gz	<a href="#">Download Model</a> <a href="#">Delete Model</a>



# Data Improvement Requirements

## Functional Requirements

Should be able to choose pipeline and policy components and train models without knowing what dependencies they require and the required order



Should be able to view trained models



Should be able to use conversations users have had with the conversational AI, to improve the machine learning model (CDD)



## Non-Functional Requirements

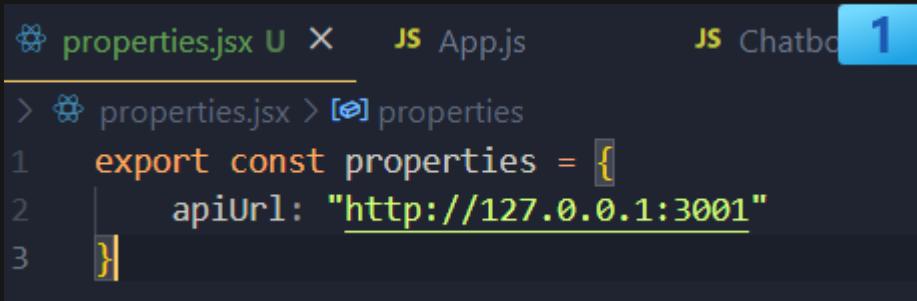
Should increase the efficiency of training a model by reducing errors using the UI compared to manually typing the pipeline and policy components



Should reduce the time consumed to create the pipeline and policy components by reducing the need to type by increasing clickable components

# Best Practices

Maintain a properties file for the front-end



properties.jsx 1

```
> properties.jsx > properties
1 export const properties = [
2   apiUrl: "http://127.0.0.1:3001"
3 ]
```



Model.js 2

```
rc > pages > Model.js > ...
30
31 import { properties } from '../properties';
32
33 const API_URL_BASE = properties.apiUrl;
```

3

```
const getModels = () => {
  axios.get(` ${API_URL_BASE}/config/update` , {
    headers: {
      'Content-Type': 'application/json'
    }
  }).then((res) => {
    setTrainedModels(res.data["Model List"]);
  });

  axios.get(` ${API_URL_BASE}/model/latest` , {
    headers: {
      'Content-Type': 'application/json'
    }
  }).then((res) => {
    setLatestModel(res.data["Latest Model"]);
  });
}
```

# Best Practices

Maintain a config file for the back-end

```
testFlaskServer.py × config.yml × .gitignore × responses.yml ×  
28  
29     def _get_dir_file_list(  
30         dir_path: Text = settingsData['DEFAULT_MODEL_PATH'],  
31         file_suffixes: List = settingsData['MODEL_EXTENSIONS']  
32     ) -> Optional[List]:  
33         file_paths = list()  
34  
35         for (dir_path, dir_names, file_names) in os.walk(dir_path):  
36             if file_suffixes:  
37                 file_paths += [file for file in file_names if str(file).endswith(tuple(file_suffixes))]  
38             else:  
39                 file_paths += [file for file in file_names]  
40         return file_paths
```

```
settings.json × config.yml × .gitignore × responses.yml × 1  
1 {  
2     "DEFAULT_MODEL_PATH": "./models",  
3     "MODEL_EXTENSIONS": [".tar.gz"],  
4     "RASA_MODEL_EXTENSIONS": [".tar.gz"],  
5     "RASA_288_MODEL_REGEX": "^(\\d{8})-(\\d{6}).tar.gz$"  
6 }
```

```
testFlaskServer.py × config.yml × .gitignore × 2  
19  
20     jsonSettings = open('settings.json')  
21     settingsData = json.load(jsonSettings)  
22
```

3



# Test Classe S

```
✓ _test_
  ✓ _snapshots_
    Chatbot.test.js.snap
    Model.test.js.snap
    Pipeline.test.js.snap
  Chatbot.test.js
  Model.test.js
  Pipeline.test.js
```

The screenshot shows the VS Code interface with several tabs at the top: Pipeline.test.js (active), Chatbot.test.js, Model.test.js, Page.js, package.json, jsconfig.json, and DashboardSidebar.js. The Pipeline.test.js file contains code for testing a 'Pipeline' component. The terminal below shows test results:

```
TERMINAL      GITLENS      PROBLEMS      OUTPUT      DEBUG CONSOLE
> 1 snapshot updated from 1 test suite.

Test Suites: 2 passed, 2 total
Tests:     8 passed, 8 total
Snapshots: 1 updated, 1 passed, 2 total
Time:      17.365 s, estimated 56 s
Ran all test suites related to changed files.

Watch Usage: Press w to show more. □
```



# Specific Objective

Develop an efficient and code-less approach to improve and evaluate conversational AI machine learning models for non-machine learning experts



# Sub Objectives

Overall completion = objective completion \* objective weight

0% \* 0.1

Developing an interface to allow making improvements to model training data without any coding knowledge.

Developing a solution for non-technical users to configure and efficiently retrain machine learning models.

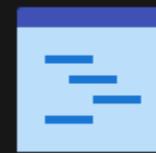
90% \* 0.6

0% \* 0.3

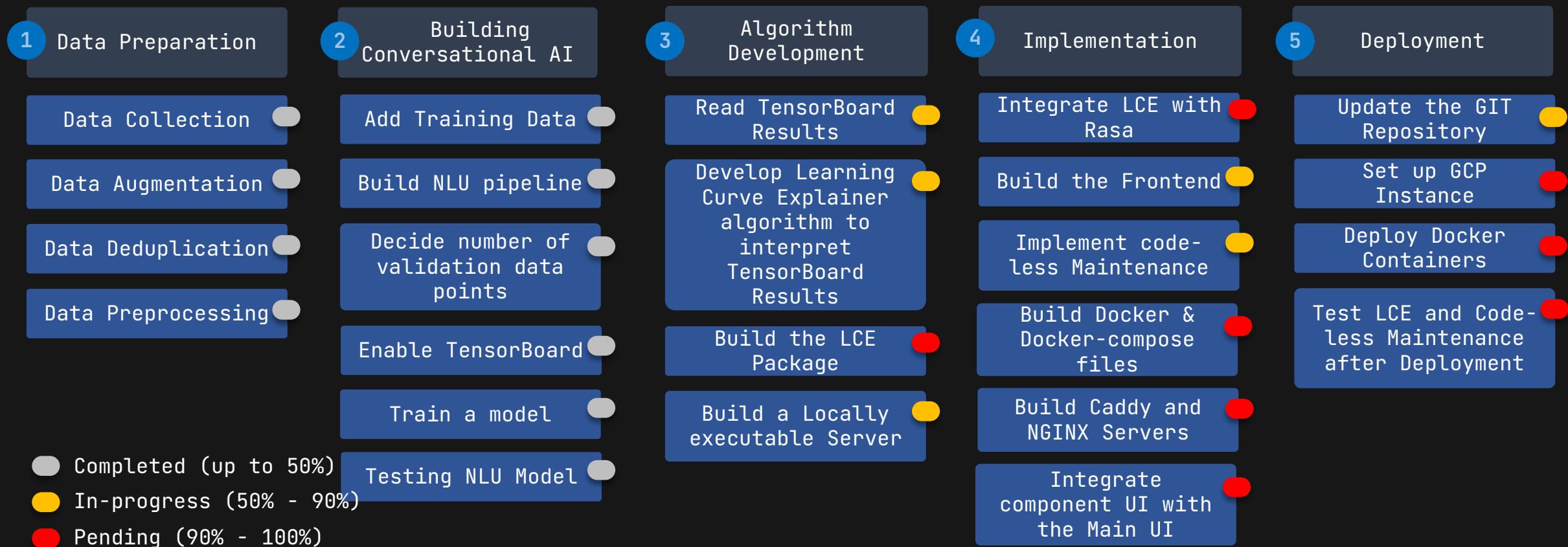
Developing an algorithm to identify any overfittings or underfittings in a model and indicate it in the frontend.

\* Overall completion rate:

54%



# Work Breakdown Structure





# Technologies

## Frontend Development



React

## Backend Development



Python



TensorBoard



Rasa



Flask

## Project Management



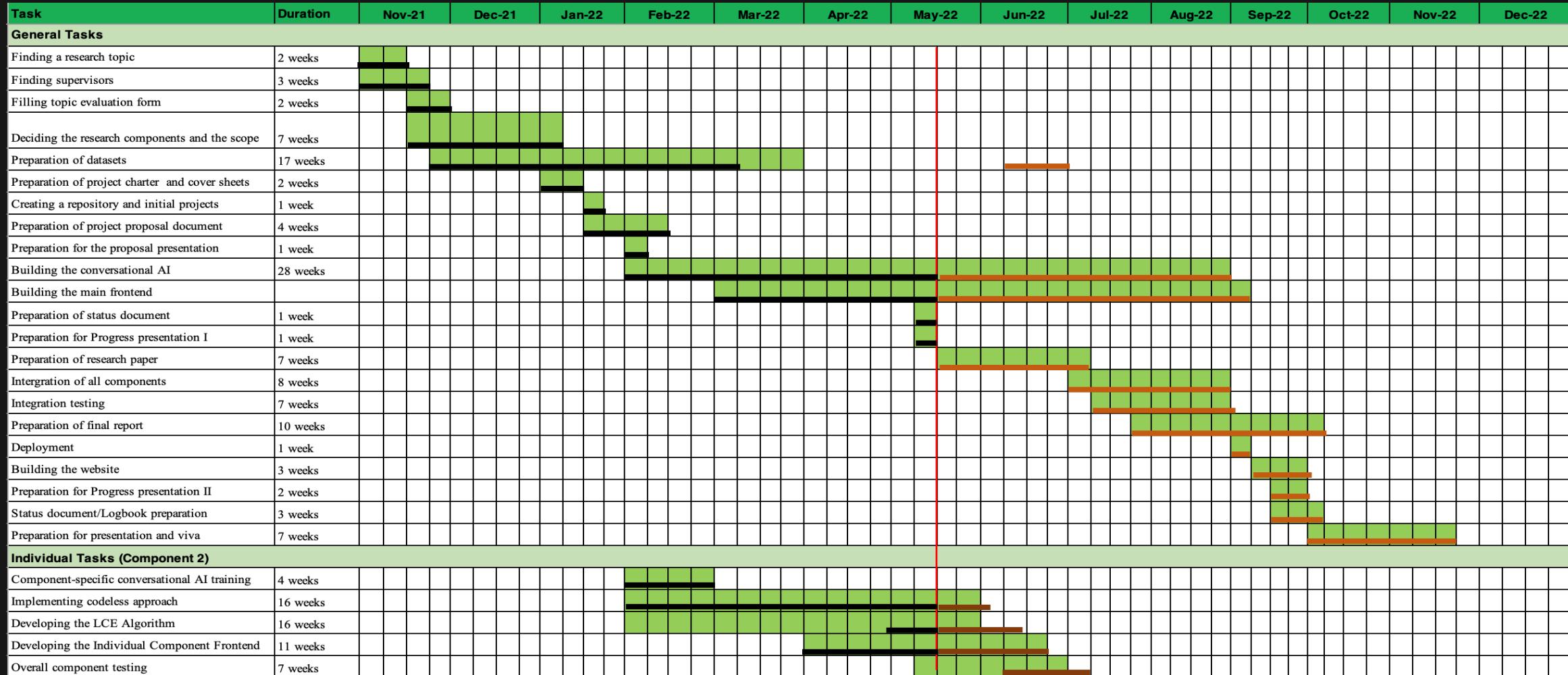
Git

GitLab  
GitHub





# Individual Gantt Chart



 Predicted Time

 Actual Time Spent

 Remaining time for general tasks

 Remaining time for individual tasks



# References

- [1]. T. Bocklisch, J. Faulkner, N. Pawłowski, en A. Nichol, "Rasa: Open source language understanding and dialogue management", *arXiv preprint arXiv:1712. 05181*, 2017.
  
- [2]. "Introduction to rasa X," *Open source conversational AI*, 10-Dec-2021. [Online]. Available: <https://rasa.com/docs/rasa-x/>. [Accessed: 22-Jan-2022].

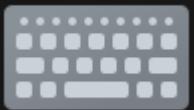
52%

# SEETM: Sinhala-English Equivalent Token Mapper

Developing rule-based approaches to process code-mixed textural data and make word embeddings models lightweight using token mapping.



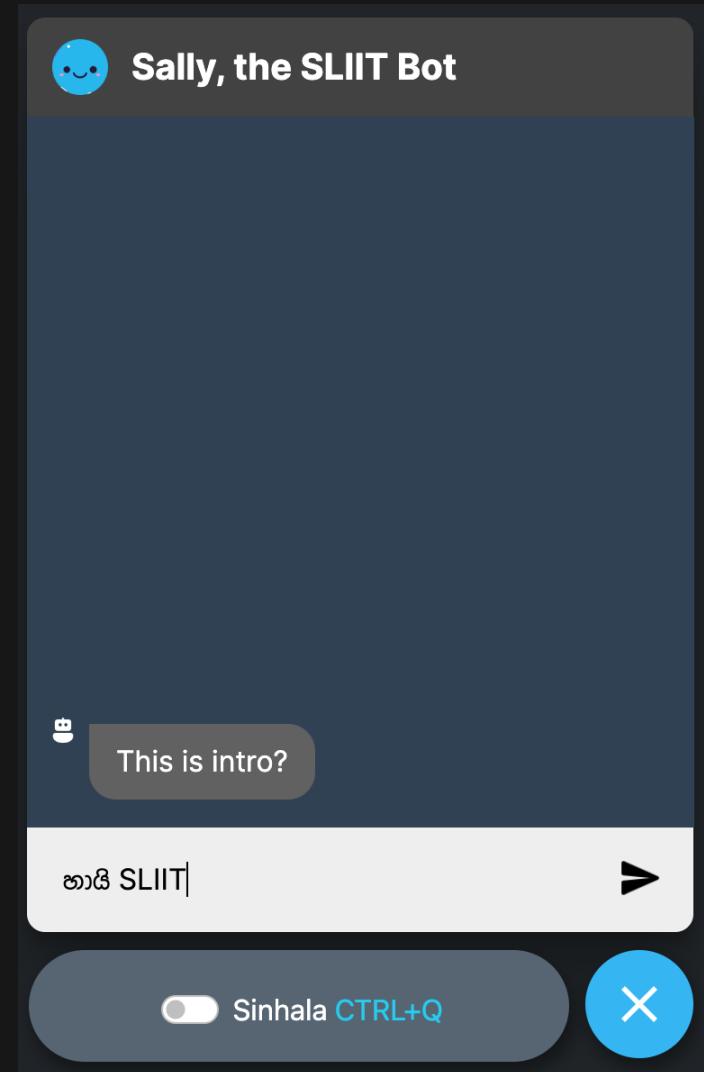
Jayasinghe D.T.  
IT19075754  
Data Science



# Sinhala-English code-switchable Keyboard Interface

Code-Switching

Library Membership  
ඒකට apply  
කරන්නේ කොහොමද?





# Functional Requirements



Ability to type in Sinhala using an English keyboard.



Ability to switch between typing Sinhala and English



Ability to type emojis if required



SEETM should handle out-of-vocabulary words in Word2Vec models when at least one of the representations of equivalent tokens are present in training data.



# Non-Functional Requirements



Efficiency



Reliability



Modularity



Usability



Scalability

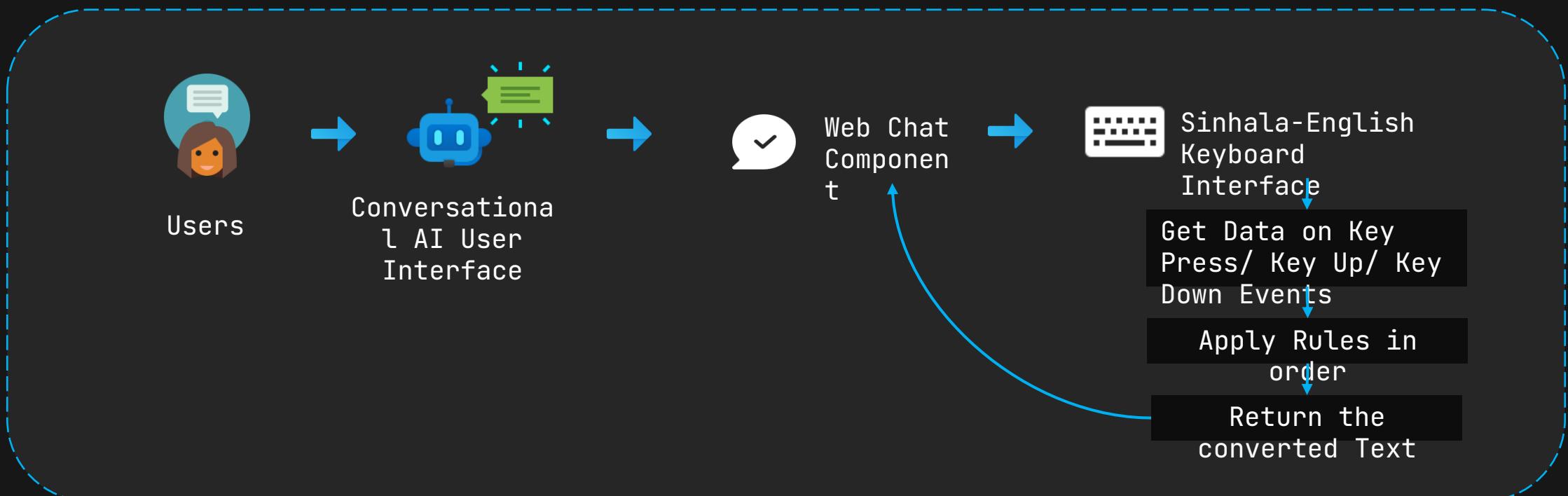


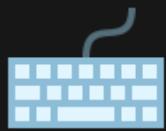
Sinhala **CTRL+Q**





# Research Flow





# Keyboards Comparison

Keyboard Name	Default (Misleading character Map)	Changes
Helakuru Keyboard (හෙලකරු)	<ul style="list-style-type: none"><li>Backspace + a → අ</li><li>c → ක්</li><li>u → උ (when re-enter)</li><li>ou → ඕ</li><li>Q/q → ස</li></ul>	<ul style="list-style-type: none"><li>Backspace + a → ඇ</li><li>c → ඩ</li><li>u → ඊ</li><li>au → එ</li><li>Q/q → (Shortcut) (dh → ස)</li></ul>



# Keyboards Comparison

Keyboard Name	Default (Misleading character Map)	Changes
UCSC Unicode Real-Time Font Converter	<ul style="list-style-type: none"><li>• c → c</li><li>• q → ක්‍රු</li><li>• Q → Q</li><li>• U → U</li><li>• V → V</li><li>• E → E</li><li>• F → F</li><li>• H → H</li><li>• M → M</li><li>• O → O</li><li>• S → S</li><li>• W → W</li><li>• x, X → x</li><li>• z → z</li></ul>	<ul style="list-style-type: none"><li>• c → ස</li><li>• Q/q → (Shortcut)</li><li>• U → උ</li><li>• V → අ</li><li>• E → එ</li><li>• F → ඌ</li><li>• H → ඕ</li><li>• M → ම</li><li>• O → මො</li><li>• S → ස</li><li>• X → ඇ</li><li>• x → ඕ.</li><li>• W → අ</li><li>• z → ග</li></ul>



# Emoji Mapping

```
var en_emoji_map = {  
    ':sun:': '☀️', ':moon:': '🌙', ':happy:': '😊', ':sad:': '😢', ':angry:': '😡',  
    ':bye:': '👋', ':worship:': '🙏', ':ok:': '👍', ':alien:': '👽'  
};
```

```
var si_emoji_map = {  
    ':ira:': '☀️', ':hadha:': '🌙', ':sathutu:': '😊', ':duka:': '😢', ':tharaha:': '😡',  
    ':baayi:': '👋', ':wadhinawaa:': '🙏', ':haa:': '👍'  
};
```



# Character Mapping

```
var sinhala_vowel_pure_map = {  
    'uu': 'ශු', 'oo': 'ශෝ', 'oe': 'ශෝ', 'aa': 'ශප', 'AA': 'ශපි', 'Aa': 'ශපි', 'ae': 'ශපි',  
    'ii': 'ශඩ', 'ie': 'ශඩ', 'ee': 'ශඩ', 'ea': 'ශඩ', 'ei': 'ශඩ', 'uu': 'ශු', 'au': 'ශඩු',  
    'O': 'ශඩු', 'a': 'ශප', 'A': 'ශපි', 'i': 'ශඩ', 'e': 'ශඩ', 'u': 'ශප', 'U': 'ශපි', 'o': 'ශඩු', 'E': 'ශඩු', 'I': 'ශඩු'  
};
```

```
var sinhala_vowel_suffix_map = {  
    'uu': 'ශු', 'oo': 'ශුෂ', 'oe': 'ශුෂ', 'aa': 'ශප', 'AA': 'ශප්', 'Aa': 'ශප්', 'ae': 'ශප්',  
    'ii': 'ශඩ', 'ie': 'ශඩ්', 'ee': 'ශඩ්', 'ea': 'ශඩ්', 'ei': 'ශඩ්', 'uu': 'ශු', 'au': 'ශඩු',  
    'O': 'ශඩු', 'a': 'ශප', 'A': 'ශප්', 'i': 'ශඩ', 'e': 'ශඩ්', 'u': 'ශප', 'U': 'ශප්', 'o': 'ශඩු', 'E': 'ශඩු', 'I': 'ශඩු'  
};
```

```
var sinhala_nonjoining_map = {  
    'RR': 'ශාෂා',  
    'z': 'ශ' + '\u200D',  
    'x': 'ශෝ',  
    'H': 'ශප්',  
    'R': 'ශා'  
};
```



# Character Mapping

```
var sinhala_consonant_map = {  
    'nndh': 'ඥ', 'nnd': 'ඦ', 'nng': 'ඤ', 'mmb': 'ට',  
    'GN': 'ඇ', 'KN': 'උ', 'Lu': 'බ', 'Th': 'ඪ', 'Dh': 'ඩ', 'gh': 'භ', 'Ch': 'ඵ', 'ph': '඲', 'kh': 'බ', 'bh': 'ප',  
    'Sh': 'ජ', 'sh': 'ඣ', 'dh': 'ඟ', 'ch': 'ච', 'th': 'ඤ',  
    'N': '඙', 'L': 'අ', 'K': 'ආ', 'G': 'ඇ', 'T': '඄', 'D': 'ං', 'P': 'ඉ', 'B': 'ඊ', 'C': 'ඃ', 'X': 'එ', 'J': 'උ',  
    't': 'ත', 'k': 'ක', 'd': 'ඩ', 'n': 'න', 'p': 'ප', 'b': 'බ', 'm': 'ම', 'M': 'ම',  
    'Y': 'ය', 'y': 'ය', 'j': 'ජ', 'l': 'ල', 'v': 'ව', 'w': 'ව', 'V': 'ඩ', 'W': 'ඩ',  
    's': 'ස', 'S': 'ස', 'h': 'හ', 'f': 'ෆ', 'F': 'ෆ', 'g': 'ග', 'c': 'ච',  
    'r': 'ර'  
};
```

```
var sinhala_special_map = {  
    'ruu': 'ශා',  
    'ru': 'ශ'  
};
```

# 1 ↓ Keyboard Interface

## 2 ↓ Ruleset

Input	aakramaNaya	saxskruthiya	:alien:	
Rule #0	aakramaNaya	sa x skruthiya	😊	Emoji
Rule #1	aakramaNaya	sa ○ s kru thiya	😊	Non-joining [සා, ○, ○:, සා]
Rule #2	aa kra maNaya	sa○s කා thiya Click to add text	😊	Special [○aa, ○a, ]
Rule #3	aa ක ma Na ya	sa ○.s කා thi ya	😊	Rakaransha ya [උර]
Rule #4	aa ක ම ගය	ස ○. S කා නිය	😊	Consonants + Vowel [ස, ක, ම.. ]
Rule #5	aa ක ම ගය	ස. ස කානිය	😊	ශ්‍රීන්දේශීලිස් + Hal [ස්, ක්, ම.. ]
Rule #6	අ කුමණය	ස.ස්කෘතිය	😊	Pure Vowels [අ, අ, ..]
	අ කුමණය	ස.ස්කෘතිය	😊	

# Rule Activation

```
✓ [105] convert("aakramaNaya")
0s

Rule 3.1 [Detected]: aakramaNaya
Rule 3.1 [Replaced]: aaந்தமாNaya
Rule 4 [Detected]: aaந்தமாNaya
Rule 4 [Replaced]: aaந்தமாஞ்யா
Rule 4 [Detected]: aaந்தமாஞ்யா
Rule 4 [Replaced]: ஆந்தமாஞ்யா
Rule 4 [Detected]: ஆந்தமாஞ்யா
Rule 4 [Replaced]: ஆந்தமாஞ்யா
Rule 6 [Detected]: ஆந்தமாஞ்யா
Rule 6 [Replaced]: அந்தமாஞ்யா
'அந்த\u200dஏஞ்யா'
```

ଆந்தமாஞ்ய

```
✓ [106] convert("saxskruthiya")
0s
▶ convert("saxskruthiya")

⇒ Rule 1 [Detected]: saxskruthiya
Rule 1 [Replaced]: saஃskruthiya
Rule 2 [Detected]: saஃskruthiya
Rule 2 [Replaced]: saஃகஷ்டhiya
Rule 4 [Detected]: saஃகஷ்டhiya
Rule 4 [Replaced]: saஃகஷ்டிya
Rule 4 [Detected]: saஃகஷ்டிya
Rule 4 [Replaced]: சஃகஷ்டிya
Rule 4 [Detected]: சஃகஷ்டிya
Rule 4 [Replaced]: சஃகஷ்டிya
Rule 5 [Detected]: சஃகஷ்டிya
Rule 5 [Replaced]: சஃக்ஷப்திya
'சஃக்ஷ\u200dபதிய'
```

சுங்கீகாந்தி

```
✓ [109] convert(":alien:")
0s

Rule 0.1 [Detected]: :alien:
Rule 0.1 [Replaced]: 😶
'😶\u200b'
```

😶

# HTML Script tag...

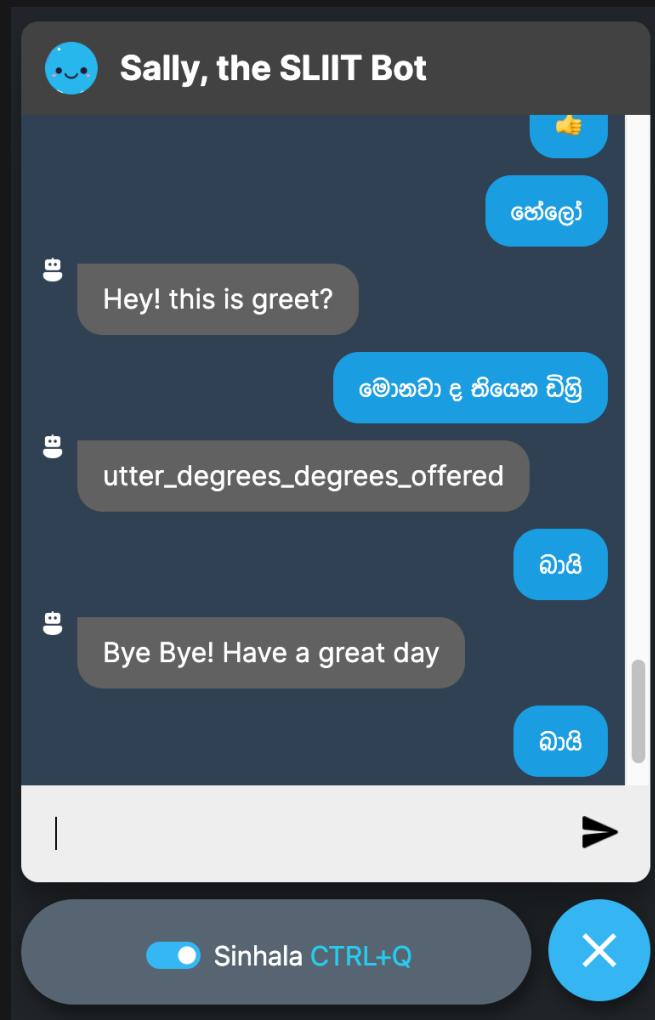
webchat\_2.0.2.html X

D: > NotCopied > Docs > Institutions > SLIIT > 5\_DS > Y4S0Research > GitLab > dev-ishara > 2022-056 > component-token-mapping

```
72
73     <script>!(function () {
74         let e = document.createElement("script"),
75             t = document.head || document.getElementsByTagName("head")[0];
76         (e.src = "static/kbi-2.0.2.js"),
77         (e.async = !0),
78         (e.onload = () => {
79             KeyboardInterfaceInit(
80                 {
81                     defaultLanguage: "SI", //EN/SI
82                     textMessageElement: ".rw-new-message", // "#tb1", //
83                     // sendMessageElement: "", //OPTIONAL
84                     switchColor: "#29B6F6",
85                     switchBarColor: "#566573",
86                     switchTextColor: "#e68d29",
87                     switchBarPosition: {
88                         hpos: "bottom",
89                         hval: "18px",
90                         vpos: "right",
91                         vval: "90px",
92                         zIndex: "10000",
93                     },
94                 },
95             );
96     });
97 
```

Attachable script  
tag for keyboard  
interface

Keyboard works on  
chatbot





# Best Practices

## PSE Keyboard v2.0.2rc Pre-release

Pre-release



### ⚡ Features

- Support language between Sinhala and English.
- New keystroke mappings.
- Fully portable.
- Fully integrable with rasa-webchat.
- Support customizing the rasa-webchat component.

### ✖ Limitations and Issues

- Only attachable with one text field within a web page.

### 🍪 What's Cooking?

- We are constantly working on new updates for the latest version.

### ▶ Assets 2



1

Release a  
Pre-release

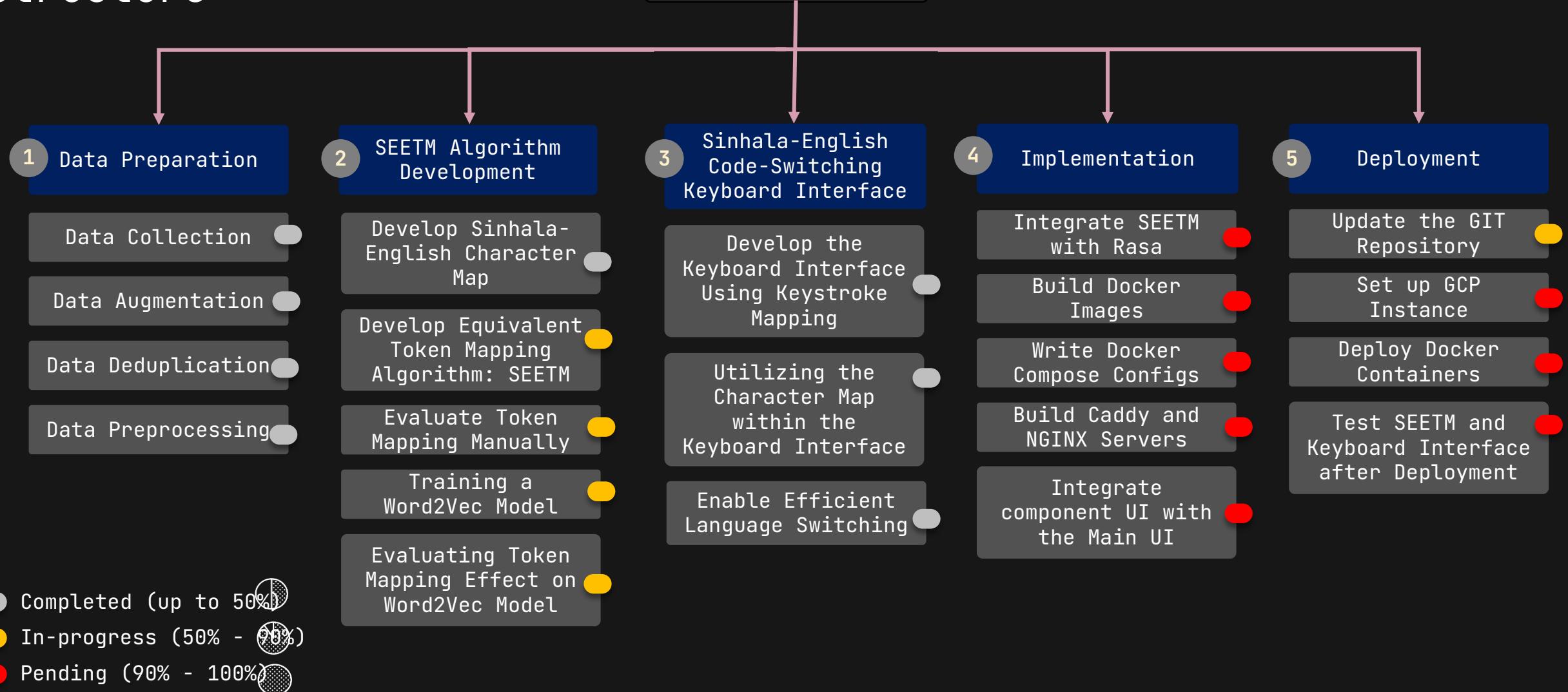
2

Implemented  
versions

```
component-token-mapping
  keyboard_interface
    media
  static
    bootstrap
      jquery-3.6.0.min.js
      kbi-0.0.1.js
      kbi-0.0.2.js
      kbi-1.0.0.js
      kbi-1.0.1.js
      kbi-2.0.0.js
      kbi-2.0.1.js
      kbi-2.0.2.js
      kbi_changelog.md
      kbi_themesets.md
      rasa_webchat.js
      rasa_webchat_100.js
      rasa_webchat_101.js
      toast-builder-1.0.0.js
    webchat.html
    webchat_2.0.2.html
```

# Work Breakdown Structure

## SEETM





# Specific Objective

Assigning the same word vector to equivalent words in a Sinhala-English code-switched text corpus

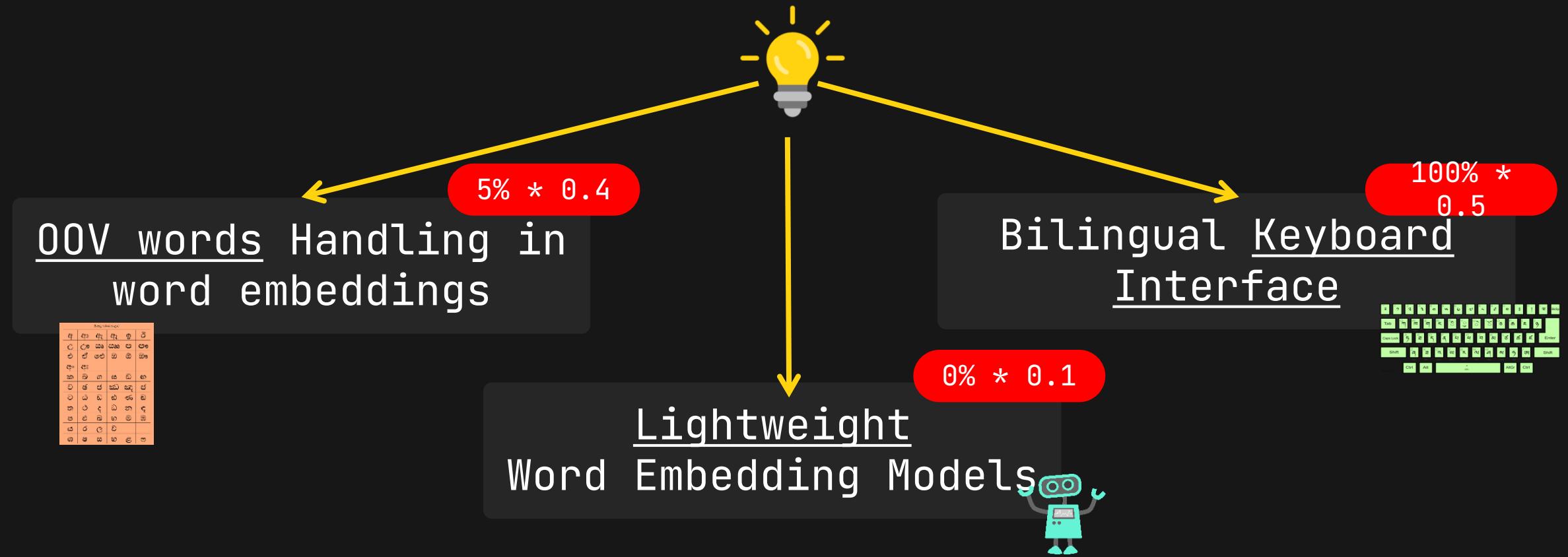


**SEETM**



# Sub Objectives

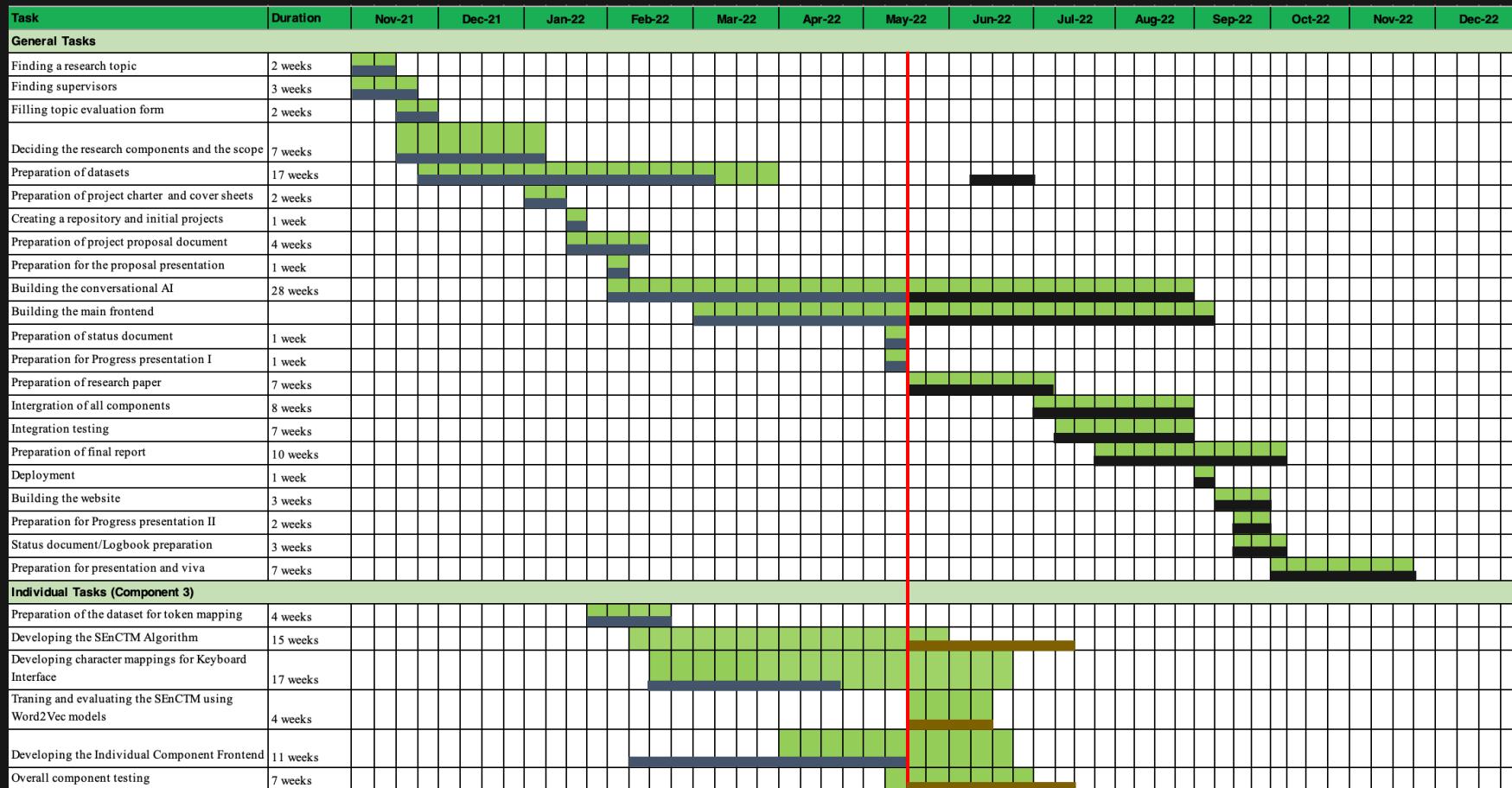
Overall completion = objective completion \* objective weight



\*Overall completion rate:  
**52%**



# Gantt Chart



● Predicted Time

● Actual Time Spent

● Remaining time for General Tasks

● Remaining time for Individual Tasks



# References

- [1]. T. KasthuriArachchi and E. Y. A. Charles, "Deep Learning Approach to Detect Plagiarism in Sinhala Text," *2019 14th Conference on Industrial and Information Systems (ICIIS)*, 2019, pp. 314-319, doi: [10.1109/ICIIS47346.2019.9063299](https://doi.org/10.1109/ICIIS47346.2019.9063299).
- [2]. A. Kugathasan and S. Sumathipala, "Standardizing Sinhala Code-Mixed Text using Dictionary based Approach," *2020 International Conference on Image Processing and Robotics (ICIP)*, 2020, pp. 1-6, doi: [10.1109/ICIP48927.2020.9367353](https://doi.org/10.1109/ICIP48927.2020.9367353).

69%

# DIME: Dual Interpretable Model-Agnostic Explanations

Using global explanations to generate local interpretations  
in intent classification models using explainable AI



Dissanayake D.M.I.M.  
IT19069432  
Data Science



# DIME Python Package

The image shows a split-screen view of a GitHub interface. On the left, the GitHub dashboard for the repository `thisisishara/dime-xai` is displayed. The repository is private and has 1 unwatched star. On the right, the detailed release page for version `v0.0.3a5` is shown. The release is a pre-release, indicated by a yellow button. It was created by `thisisishara` 5 days ago and includes a commit hash `v0.0.3a5 -o 54a6525`. The release notes state: "Pre-release of DIME (Dual Interpretable Model-agnostic Explanations) Models." Below this, a section titled "Features" lists the following capabilities:

- Explain RASA DIET Classifiers using feature importance
- Generate dual feature importance scores
- No Surrogate models, thus efficient
- Total confidence drop as the feature importance score
- Explain both local and REST models
- Easy to use DIME CLI

The right side of the image shows the GitHub release page for `dime-xai 0.0.3a5`. The page includes a search bar, navigation links for "Project description", "Release history", and "Download files", and a "Release history" section showing two entries: "0.0.3a5 PRE-RELEASE May 17, 2022" and "0.0.3a4 PRE-RELEASE". A "Latest version" button is also present.



# Functional Requirements

- DIME should provide methods to calculate global and local explanations
- DIME should provide a local server as a visualization tool
- DIME should be applicable to any text classification model that outputs confidence scores for predictions
- DIME should provide adjustable configurations required for explanations
- DIME should utilize caching to optimize calculations



# Non-functional Requirements



Efficient Calculations



Reliable Explanations



Simple & Interpretable Visualizations



Modular package



# DIME Interfaces



CLI Interface  
(terminal)



GUI Interface  
(Web)



Python API  
(Python/Jupyter)

```
Command Prompt - "C:\Program" + 
(rasa288) D:\dime_test\test>dime
usage: dime [-h] [-v] {server,explain,visualize,init} ...
starts DIME CLI

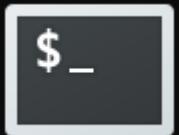
positional arguments:
  {server,explain,visualize,init}
    desired DIME interface to run [cli/server]
    server          run DIME server, a web-based visualization tool for
                    DIME.
    explain         run DIME CLI explainer, a terminal-based explainer
                    tool for DIME.
    visualize      run DIME CLI visualizer, a terminal-based
                    visualization tool for already generated DIME
                    explanations.
    init           create init dir structure for a new explanation
                    process.
```

```
CO DIME_Notebook_v0.0.3a5.ipynb ☆
File Edit View Insert Runtime Tools Help All changes saved
+ Code + Text
[ ] from dime_xai.core.rasa_dime_explainer import RasaDIMEExplainer
instances = ["degree Good Morning!"]

[ ] %pwd
'/content/drive/MyDrive/DIME/v0.0.3a5'

[ ] exp = RasaDIMEExplainer(
  models_path="/content/drive/MyDrive/DIME/v0.0.3a5/models/",
  testing_data_path="/content/drive/MyDrive/DIME/v0.0.3a5/data/",
  model_mode="local",
  model_name="20220508-124916.tar.gz",
  data_instances=instances,
  ranking_length=10,
)

Parsing all instances: 100%|██████████| 869/869 [00:10<00:00, 85.11it/s]
```





# DIME Project

```
dime_config.yml X
D: > dime_test > test > dime_config.yml > ...
8   dime_base_configs:
9     - languages:
10       - 'en'
11       - 'si'
12     - data_path: ./data/
13     - models_path: ./models/
14     - model_name: latest
15     - model_type: diet
16     - rasa_version: 2.8.8
17     - model_mode: local
18     - url_endpoint: http://localhost:5005
19     - data_instance:
20       - "SLIIT එකේ තියෙන උපාධි මොනවද?"
21       - "First class degree එකක් ගන්න GPA එක කොට්ටර ඕනෑද"
22     - ranking_length: 10
23     - ngrams: False
24     - min_ngrams: 1
25     - max_ngrams: 2
26     - case_sensitive: True
27     - metric: confidence
```

```
33   dime_server_configs:
34     - host: localhost
35     - port: 6067
36     - output_mode: dual
37
38   # Configurations that applies to DIME CLI
39   # If no config is provided following are the default values
40   # Please adjust them accordingly.
41   # See https://docs link here) for more information
42   dime_cli_configs:
43     - output_mode: dual
44
```

```
(rasa288) D:\dime>dime init
👋 Hi there! Welcome to DIME.
In which directory do you want to initialize DIME? [Default: Current Directory]:
2022-05-23 21:24:25,613 INFO    dime_xai.utils.scaffold - Initialized a new dime project.

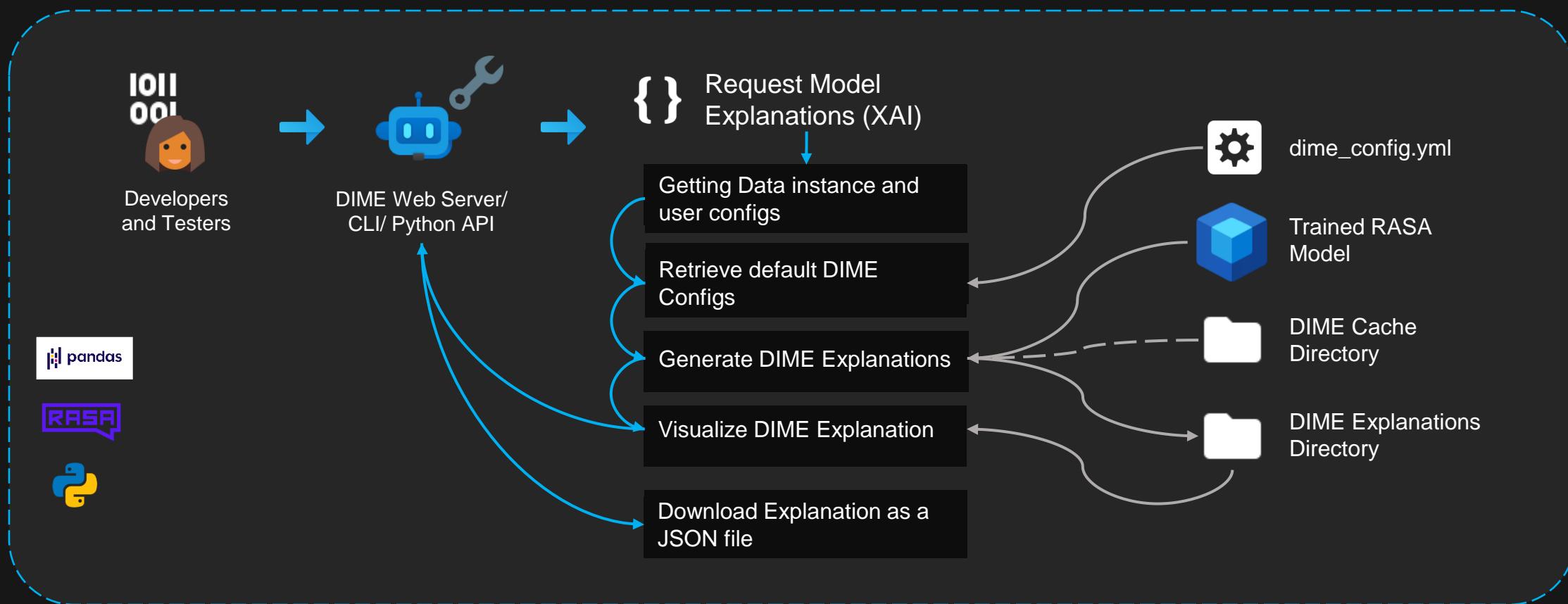
(rasa288) D:\dime>
```

This PC > Local Disk (D:) > dime

Name
.dime_cache
data
dime_explanations
models
dime_config.yml
dime_instructions.md

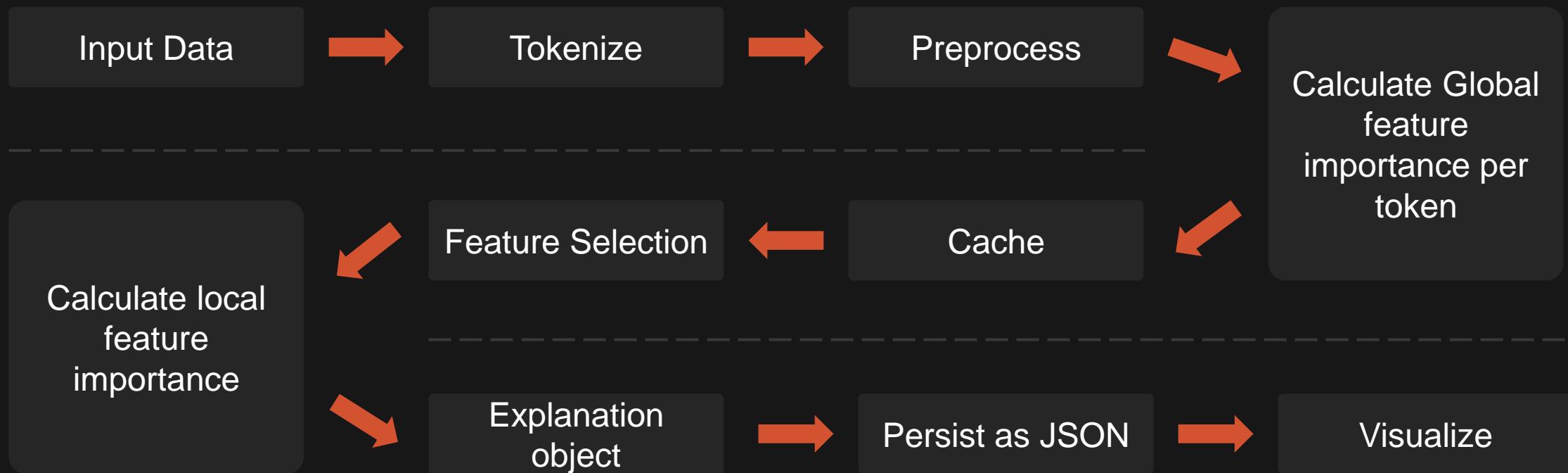
1 + 1

# Generating DIME Explanations





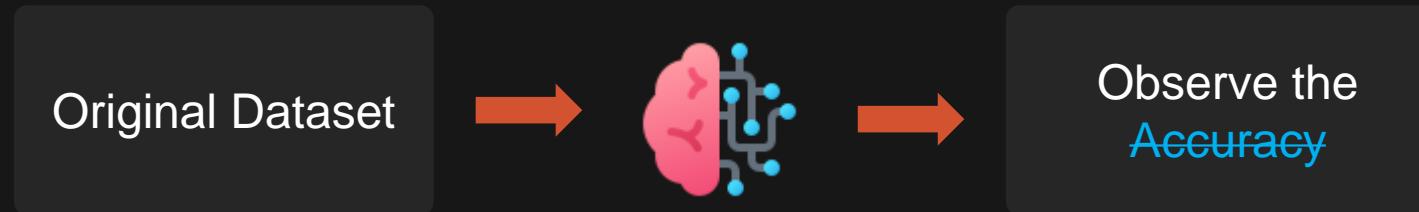
# DIME Algorithm



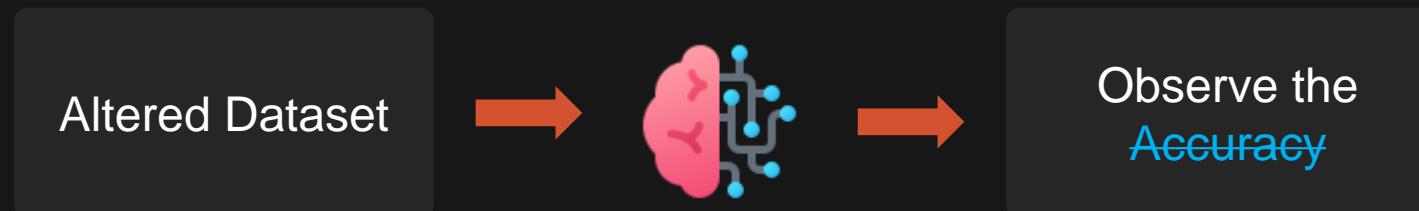


# Dual Feature Importance Metric

Finding Global Feature Importance for “මොනවදා”:



Removing “මොනවදා”



Global Feature Importance of “මොනවදා” = Accuracy Difference

→ Confidence Difference

It is recommended to use model confidence as the metric

Why?

Confidence is less sparse

DIME supports Accuracy and F1-Score, however they are highly sparse

## Accuracy

```
"instance": "SLIIT එක් තියෙක උපාධි ඔහුවද?",  
"global": {  
    "feature_importance": {  
        "ඔහුවද": 0.004602991944764101,  
        "තියෙක": 0.0011507479861909697,  
        "SLIIT": 0.0,  
        "ශ්‍රාපාධි": 0.0,  
        "එක්": 0.0  
    },  
    "feature_selection": {  
        "ඔහුවද": 0.004602991944764101,  
        "තියෙක": 0.0011507479861909697,  
        "SLIIT": 0.0,  
        "ශ්‍රාපාධි": 0.0,  
        "එක්": 0.0  
    },  
    "dual": {  
        "feature_importance": {  
            "ඔහුවද": 0.004433035850524902,  
            "තියෙක": -0.002359926700592041  
        },  
        "normalized_scores": {  
            "ඔහුවද": 1.0,  
            "තියෙක": 0.0  
        },  
        "probability_scores": {  
            "ඔහුවද": 1.0,  
            "තියෙක": 0.0  
        },  
    }  
},
```

Feature Selection when metric is “Accuracy”, or “F1-Score” is biased since ordering tokens with the same score is always the same.

## F1-score

```
"instance": "SLIIT එක් තියෙක උපාධි ඔහුවද?",  
"global": {  
    "feature_importance": {  
        "ඔහුවද": 0.003601354038019422,  
        "තියෙක": 0.0005901271724055857,  
        "SLIIT": 0.0,  
        "ශ්‍රාපාධි": 0.0,  
        "එක්": 0.0  
    },  
    "feature_selection": {  
        "ඔහුවද": 0.003601354038019422,  
        "තියෙක": 0.0005901271724055857,  
        "SLIIT": 0.0,  
        "ශ්‍රාපාධි": 0.0,  
        "එක්": 0.0  
    },  
    "dual": {  
        "feature_importance": {  
            "ඔහුවද": 0.004433035850524902,  
            "තියෙක": -0.002359926700592041  
        },  
        "normalized_scores": {  
            "ඔහුවද": 1.0,  
            "තියෙක": 0.0  
        },  
        "probability_scores": {  
            "ඔහුවද": 1.0,  
            "තියෙක": 0.0  
        },  
    }  
},
```

## Model Confidence

```
"instance": "SLIIT එක් තියෙක උපාධි ඔහුවද?",  
"global": {  
    "feature_importance": {  
        "ඔහුවද": 4.373067244887352,  
        "තියෙක": 1.5122178196907043,  
        "එක්": 0.4100315570831299,  
        "ශ්‍රාපාධි": 0.15255939960479736,  
        "SLIIT": 0.14157140254974365  
    },  
    "feature_selection": {  
        "ඔහුවද": 4.373067244887352,  
        "තියෙක": 1.5122178196907043,  
        "එක්": 0.4100315570831299,  
        "ශ්‍රාපාධි": 0.15255939960479736,  
        "SLIIT": 0.14157140254974365  
    },  
    "dual": {  
        "feature_importance": {  
            "SLIIT": 0.006376385688781738,  
            "ශ්‍රාපාධි": 0.004443705081939697,  
            "ඔහුවද": 0.004433035850524902,  
            "එක්": 0.0008720159530639648,  
            "තියෙක": -0.002359926700592041  
        },  
        "normalized_scores": {  
            "SLIIT": 1.0,  
            "ශ්‍රාපාධි": 0.69690029725738,  
            "ඔහුවද": 0.6952270560302118,  
            "එක්": 0.13675709024285368,  
            "තියෙක": 0.0  
        },  
        "probability_scores": {  
            "SLIIT": 0.3954312750660728,  
            "ශ්‍රාපාධි": 0.27557617313841093,  
            "ඔහුවද": 0.2749145212264587,  
            "එක්": 0.054078030569057606,  
            "තියෙක": 0.0  
        },  
    }  
},
```



# Dual Feature Importance Probabilities

Problem:  
Exp norm Softmax Scores



```
"dual": {  
    "feature_importance": {  
        "SLIIT": 0.006376385688781738,  
        "උපාධි": 0.004443705081939697,  
        "මහත්වය": 0.004433035850524902,  
        "ලක්": 0.0008720159530639648,  
        "නියෙක": -0.002359926700592041  
    },  
    "normalized_scores": {  
        "SLIIT": 1.0,  
        "උපාධි": 0.69690029725738,  
        "මහත්වය": 0.6952270560302118,  
        "ලක්": 0.13675709024285368,  
        "නියෙක": 0.0  
    },  
    "softmax_scores": {  
        "SLIIT": 0.2007250101894944,  
        "උපාධි": 0.20033744749308377,  
        "මහත්වය": 0.20033531005789784,  
        "ලක්": 0.19962318073849,  
        "නියෙක": 0.19897905152103398  
    }  
},
```

Unrealistic  
Importance  
Scores



Solution:  
Probability Scores



```
"dual": {  
    "feature_importance": {  
        "SLIIT": 0.006376385688781738,  
        "උපාධි": 0.004443705081939697,  
        "මහත්වය": 0.004433035850524902,  
        "ලක්": 0.0008720159530639648,  
        "නියෙක": -0.002359926700592041  
    },  
    "normalized_scores": {  
        "SLIIT": 1.0,  
        "උපාධි": 0.69690029725738,  
        "මහත්වය": 0.6952270560302118,  
        "ලක්": 0.13675709024285368,  
        "නියෙක": 0.0  
    },  
    "probability_scores": {  
        "SLIIT": 0.3954312750660728,  
        "උපාධි": 0.27557617313841093,  
        "මහත්වය": 0.2749145212264587,  
        "ලක්": 0.054078030569057606,  
        "නියෙක": 0.0  
    }  
},
```



# DIME Algorithm cont.

```
2022-05-22 23:21:17,315 INFO dime_xai.shared.explanation - DIME explanations were persisted in dime_explanations directory under dime_results_20220522_232117.json
```

DUAL FEATURE IMPORTANCE SCORES

DATA INSTANCE: SLIIT එක තීයෙක උපාධි ඔහුවද?

PREDICTED INTENT: degrees\_degrees\_offered

CONFIDENCE: 0.979936957359314

Explanation Type: dual

Case Sensitive: True

Global Metric: confidence

N-grams: -

Ranking Length: 10

Global feature importance scores (Raw):     ඔහුවද=4.373067244887352, තීයෙක=1.5122178196907043, එක=0.4100315570831299, උපාධි=0.15255939960479736, SLIIT=0.14157140254974365

Selected token list based on global score:     ඔහුවද, තීයෙක, එක, උපාධි, SLIIT

Global feature importance scores (Normalized):     ඔහුවද=1.0, තීයෙක=0.34580255345002325, එක=0.09376292065083305, උපාධි=0.034886131646650956, SLIIT=0.03237347944174832

Global feature importance probability scores:     ඔහුවද=0.6636470349671681, තීයෙක=0.2294908392811836, එක=0.06222548427978721, උපාධි=0.0231520778287742, SLIIT=0.021484563643086846

වෛත්‍යවදී: [redacted] 0.6636470349671681

තීයෙක: [redacted] 0.2294908392811836

එක: [redacted] 0.06222548427978721

උපාධි: [redacted] 0.0231520778287742

SLIIT: [redacted] 0.021484563643086846

Dual feature importance scores (Raw):     SLIIT=0.006376385688781738, උපාධි=0.004443705081939697, ඔහුවද=0.004433035850524902, එක=0.0008720159530639648, තීයෙක=-0.002359926700592041

Dual feature importance scores (Normalized):     SLIIT=1.0, උපාධි=0.69690029725738, ඔහුවද=0.6952270560302118, එක=0.13675709024285368, තීයෙක=0.0

Dual feature importance probability scores:     SLIIT=0.3954312750660728, උපාධි=0.27557617313841093, ඔහුවද=0.2749145212264587, එක=0.054078030569057606, තීයෙක=0.0

SLIIT: [redacted] 0.3954312750660728

උපාධි: [redacted] 0.27557617313841093

වෛත්‍යවදී: [redacted] 0.2749145212264587

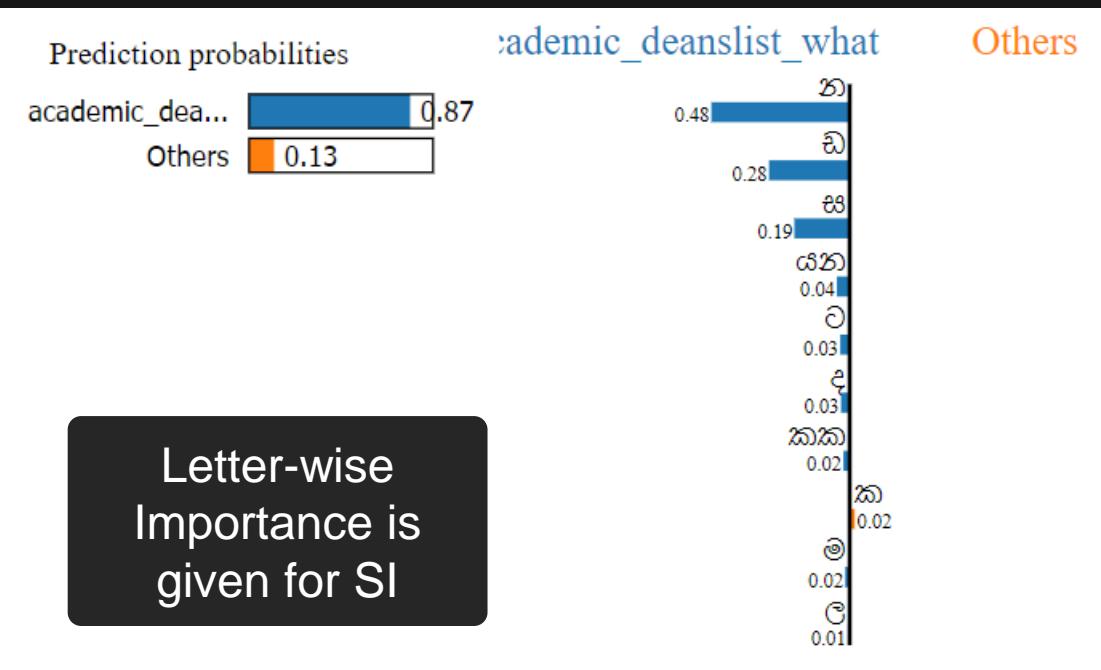
එක: [redacted] 0.054078030569057606

තීයෙක: 0.0



# Comparison LIME vs DIME

Problem with LIME



DIME



Dual feature importance scores (Raw):  
Dual feature importance scores (Normalized):  
Dual feature importance probability scores:

SLIIT:	0.2007250101894944
අපාධි :	0.20033744749308377
මෙහවද :	0.20033531005789784
මෙක :	0.19962318073849
නීයෙන :	0.19897905152103398

(rasa288) D:\dime\_test\test>



# Best Practices

The Code

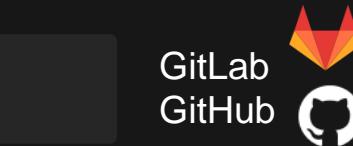
Threading

Docstrings

PEP8

Releases

Git



PyPI

Test PyPI

## Docstrings & PEP8

```
def get_all_tokens(
    instances: Union[Text, List],
    merge: bool = False
) -> Optional[Union[List, Dict]]:
    """
    Returns all tokens present in a single string instance
    or a list of string instances

    Args:
        instances: single string instance or a list of strings
        merge: if True, merges a list of strings passed as
               arguments into one list of tokens

    Returns:
        for a single string, returns the list of tokens.
        for a list of unmerged strings, returns the list
        of tokens per each string instance as a dictionary
    """

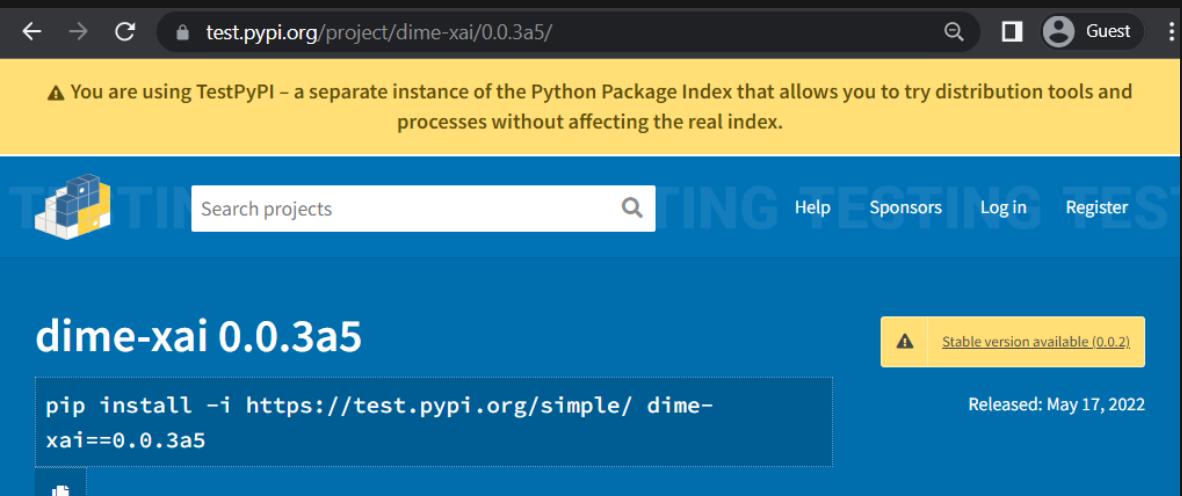
    instances_copy = instances
    if not instances_copy:
        return None

    if merge and isinstance(instances_copy, List):
```

## Threading for parsing RASA model

```
try:
    if is_supervised:
        tqdm_dataset = tqdm(dataset.items())
        tqdm_dataset.set_description(f" {description}")
        for intent, examples in tqdm_dataset:
            with ThreadPoolExecutor(max_workers=os.cpu_count() * 5) as executor:
                for example in examples:
                    future = executor.submit(
                        self._parse_rest,
                        example,
                    )
                    rasa_responses.append([example, intent, future])
    return self._process_supervised_batch_output(rasa_responses)
```

## Test PyPI alpha version release





# Specific Objective

Develop DIME, an Explainable AI approach to deliver local model explanations with the help of global feature importance.



# Sub Objectives

**80% \* 0.4**

Find Global and Local Explanations logically

**80% \* 0.15**

Develop a python package for DIME

**100% \* 0.05**

Modify DIET intent classifier to get all confidence scores

**40% \* 0.3**

Visualize explanations in an interpretable manner

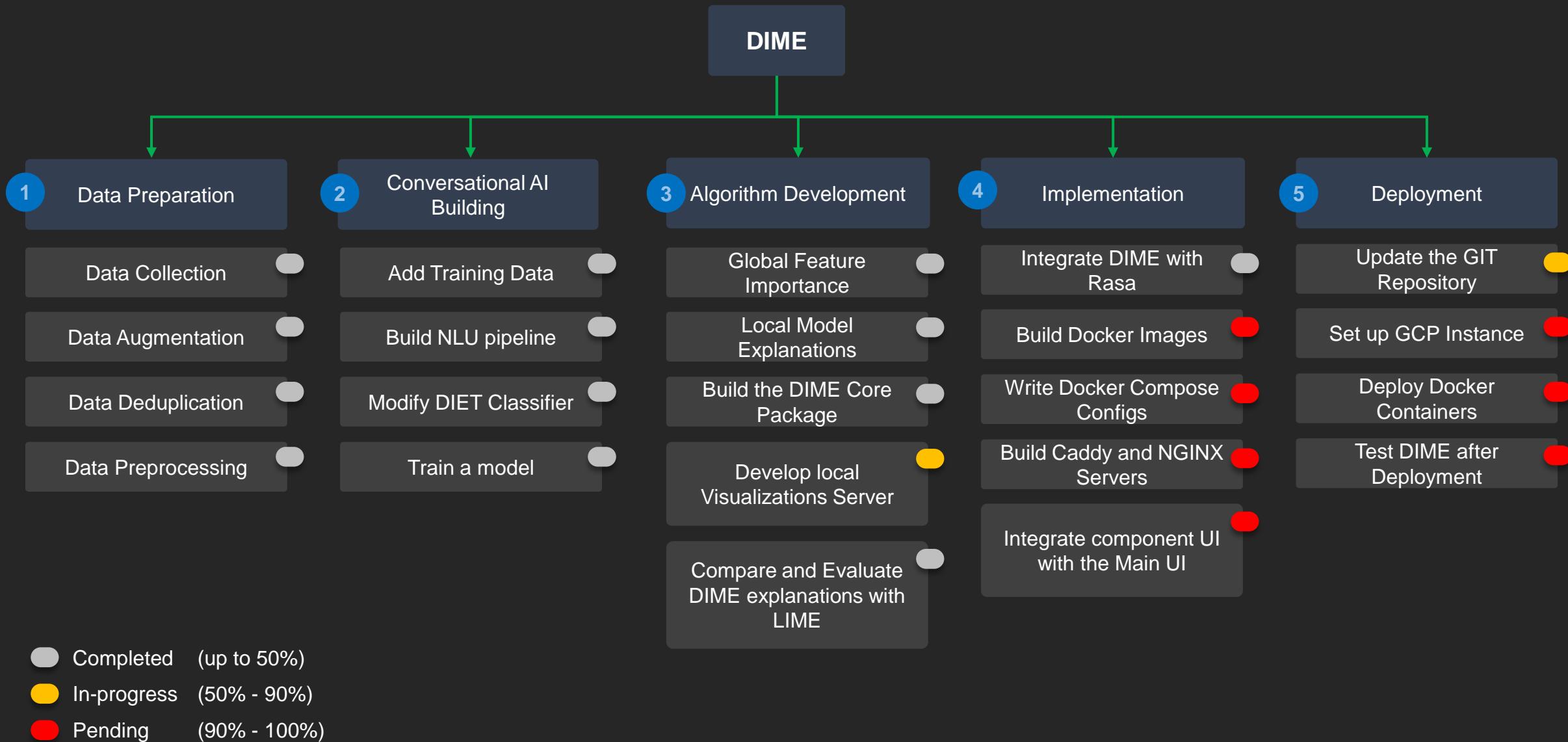
**80% \* 0.1**

Integrate DIME with Rasa seamlessly

Overall completion = (objective completion) x (objective weight) = **69%**

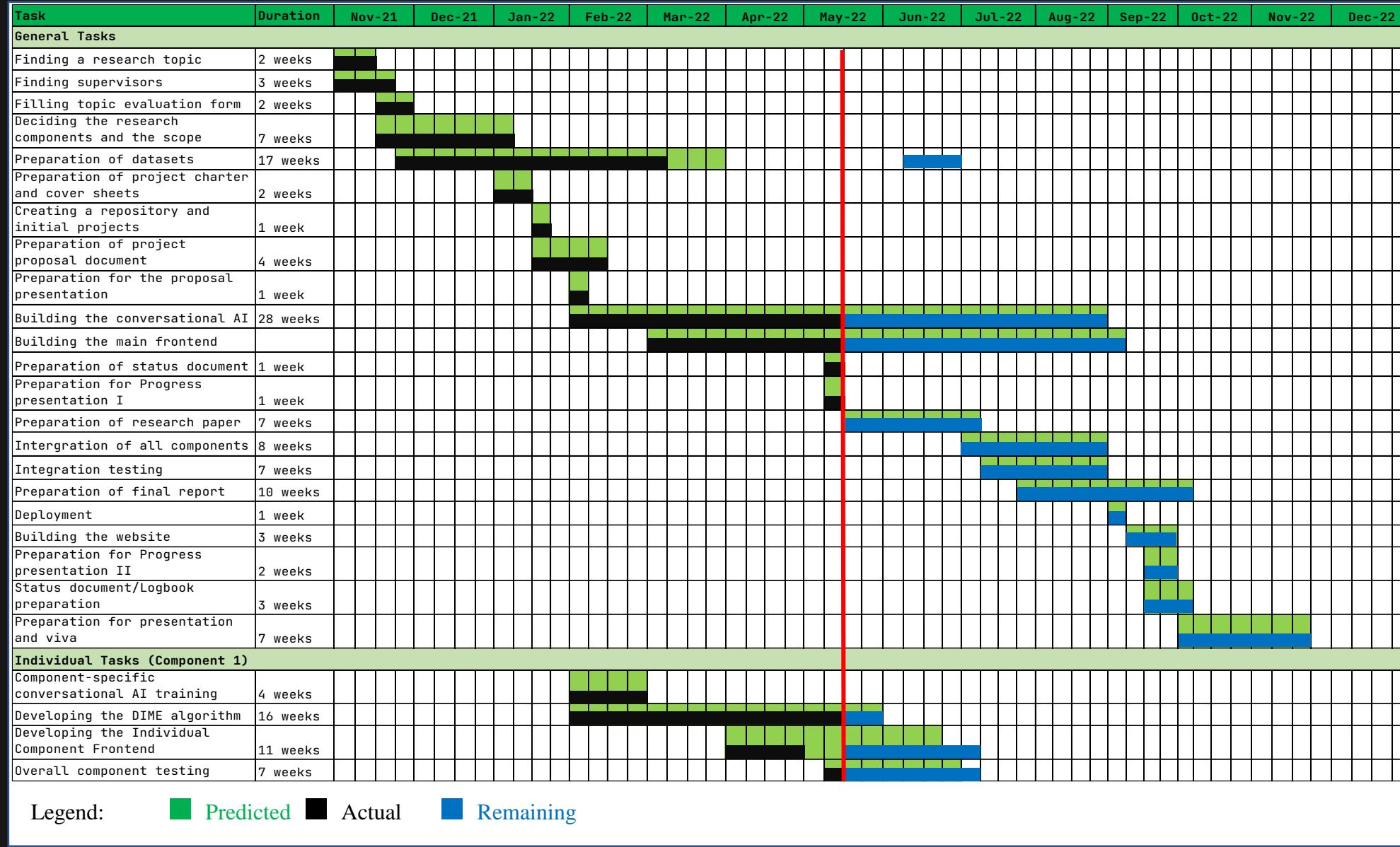


# Work Breakdown Structure





# Individual Gantt Chart





# References

- [1]. M. T. Ribeiro, S. Singh, C. Guestrin, “why should I trust you?’: Explaining the predictions of any classifier,” 16 Feb 2016. [Online]. Available: <https://arxiv.org/abs/1602.04938>
- [2]. S. Lundberg, S. Lee, “A Unified Approach to Interpreting Model Predictions,” 2017. [Online]. Available: <https://arxiv.org/abs/1705.07874>
- [3]. T. Bunk, D. Varshneya, V. Vlasov, A. Nichol, “DIET: Lightweight Language Understanding for Dialogue Systems,” 2020. [Online]. Available: <https://arxiv.org/abs/2004.09936>



# Commercialization Plan



## DEMO PACKAGE

## CaaS PACKAGES

Free for 1 Month

10 Intents/Question Categories  
2 API Integrations  
Bot Analytics Included  
Unlimited CDD Improvements



## ON PREM PACKAGE

\$ 199.99/One Time

2 Free Maintenance.  
(\$9.99 per additional call)  
Bot Analytics + CDD

### STARTER

\$ 9.99/Month

20 Intents/Question Categories  
2 API Integrations  
No Bot Analytics  
Unlimited CDD Improvements

### PRO

\$ 34.99/Month

180 Intents  
110 API Integrations  
Bot Analytics Included  
CDD + Sinhala Entity Annotating



### GENIUS

\$ 49.99/Month

400 Intents  
200 API Integrations  
Bot Analytics Included  
CDD + ML Packages



# Budget Plan

Component Name	Individual Item Price (LKR)	Number of Items	Duration	Total Item Price (LKR)
Domain Name	2148.43 /year	1	1 year	2148.43
GCP Instance	10683.84 /month	1	6 months	64,103.06
Reference Book: Basaka Mahima by J.B. Dissanayake	1250.00	1	-	1250.00
Research Paper Publication	25000.00	-	-	25000.00
Grand Total				<b><u>92,501.49</u></b>



# Thank You



# Demonstration



# Any Questions