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

2022-056

Progress Presentation - II

The Team

Supervisor
Dr. Lakmini Abeywardhana

Co – Supervisor
Ms. Dinuka Wijendra

Dissanayake D.M.I.M.
IT19069432
(Explainable AI)

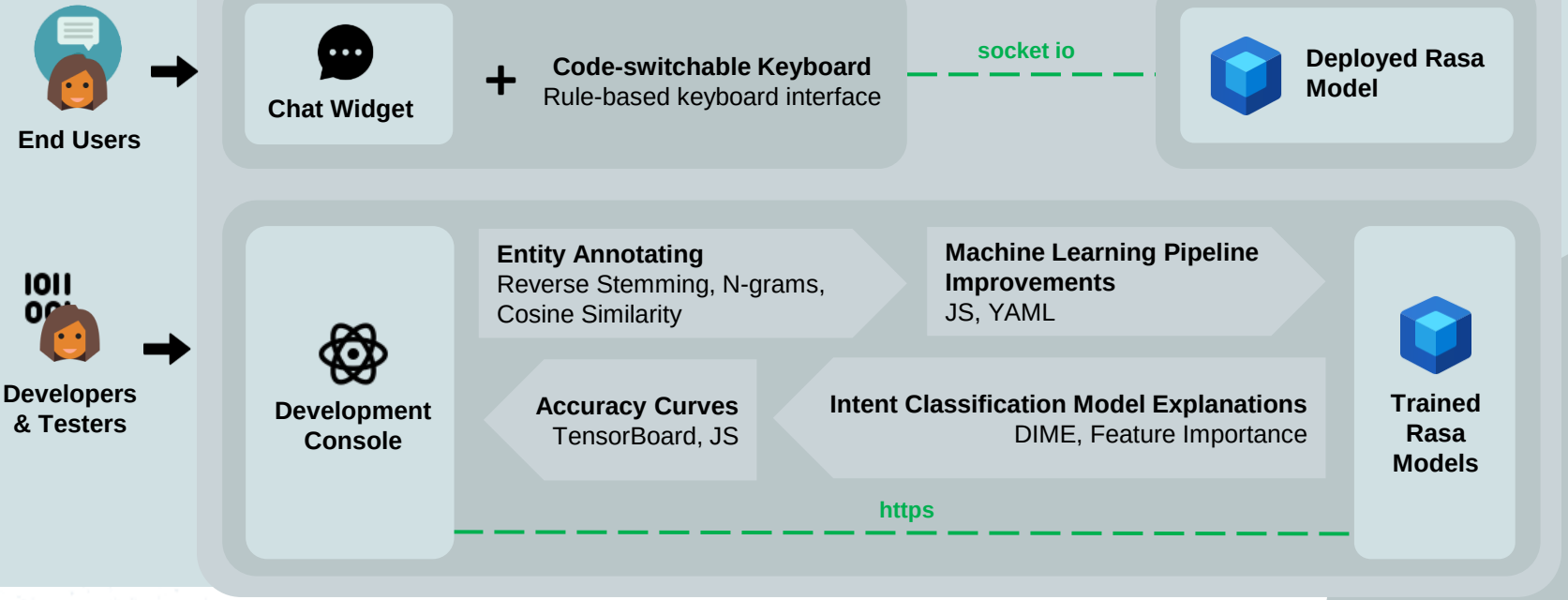
Hameed M.S.
IT19064932
(Codeless Interface)

Jayasinghe D.T.
IT19075754
(Language & Interactions)

Sakalasooriya S.A.H.A.
IT19051208
(Entity Annotation)

kolloqe

Sysyem Architecture



98%

SIENA: Annotating entities using reverse-stemming & other techniques

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

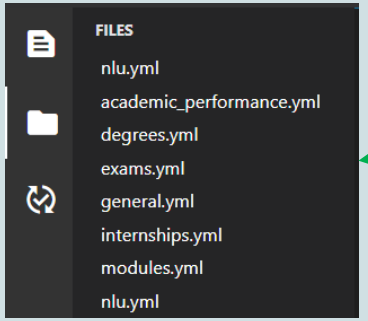
Sakalasooriya S.A.H.A.
IT19051208
Data Science

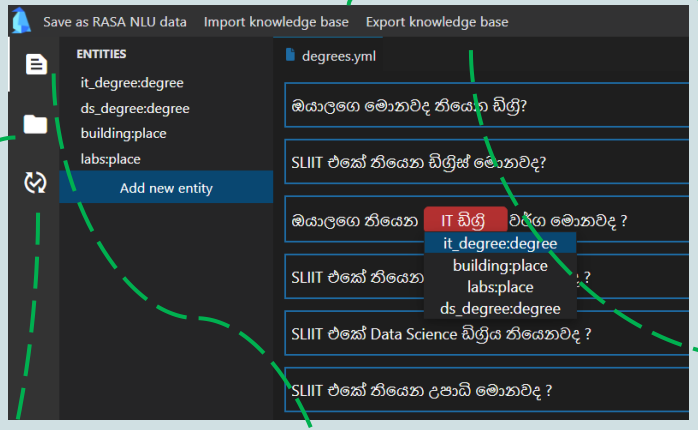
Problem

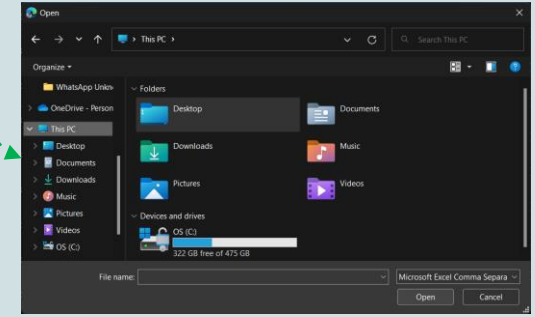
Why custom name entity tagging is very time consuming? 💤 🥱

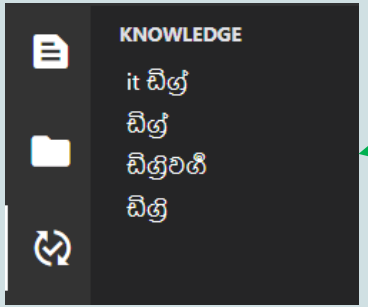
Any solution? 🤔

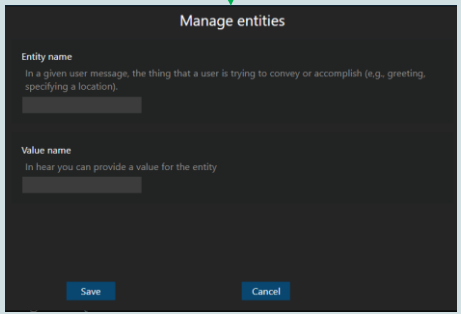
Solution

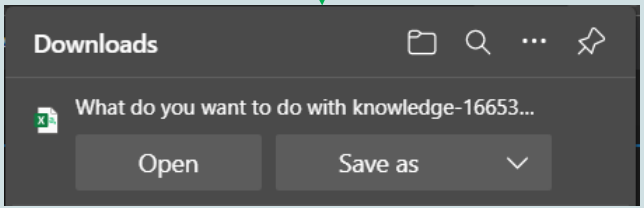

FILES
nlu.yml
academic_performance.yml
degrees.yml
exams.yml
general.yml
internships.yml
modules.yml
nlu.yml


Save as RASA NLU data
Import knowledge base
Export knowledge base
ENTITIES
it_degree:degree
ds_degree:degree
building:place
labs:place
Add new entity
degrees.yml
ඔයාලගෙ මොනවද තියෙන ඩිග්‍රී?
SLIIT එකේ තියෙන ඩිග්‍රිස් මොනවද?
ඔයාලගෙ තියෙන IT ඩිග්‍රී වර්ග මොනවද ?
it_degree:degree
building:place
labs:place
ds_degree:degree
SLIIT එකේ Data Science ඩිග්‍රිය තියෙනවද ?
SLIIT එකේ තියෙන උපාධි මොනවද ?


Open
This PC
Organize
Folders
Desktop
Documents
Downloads
Music
Pictures
Videos
Devices and drives
OS (C:)
322 GB free of 475 GB
File name:
Microsoft Excel Comma Separa
Open
Cancel


KNOWLEDGE
it ඩිග්
ඩිග්
ඩිග්‍රීවගී
ඩිග්


Manage entities
Entity name
In a given user message, the thing that a user is trying to convey or accomplish (e.g., greeting, specifying a location).
Value name
In hear you can provide a value for the entity
Save
Cancel


Downloads
What do you want to do with knowledge-16653...
Open
Save as

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

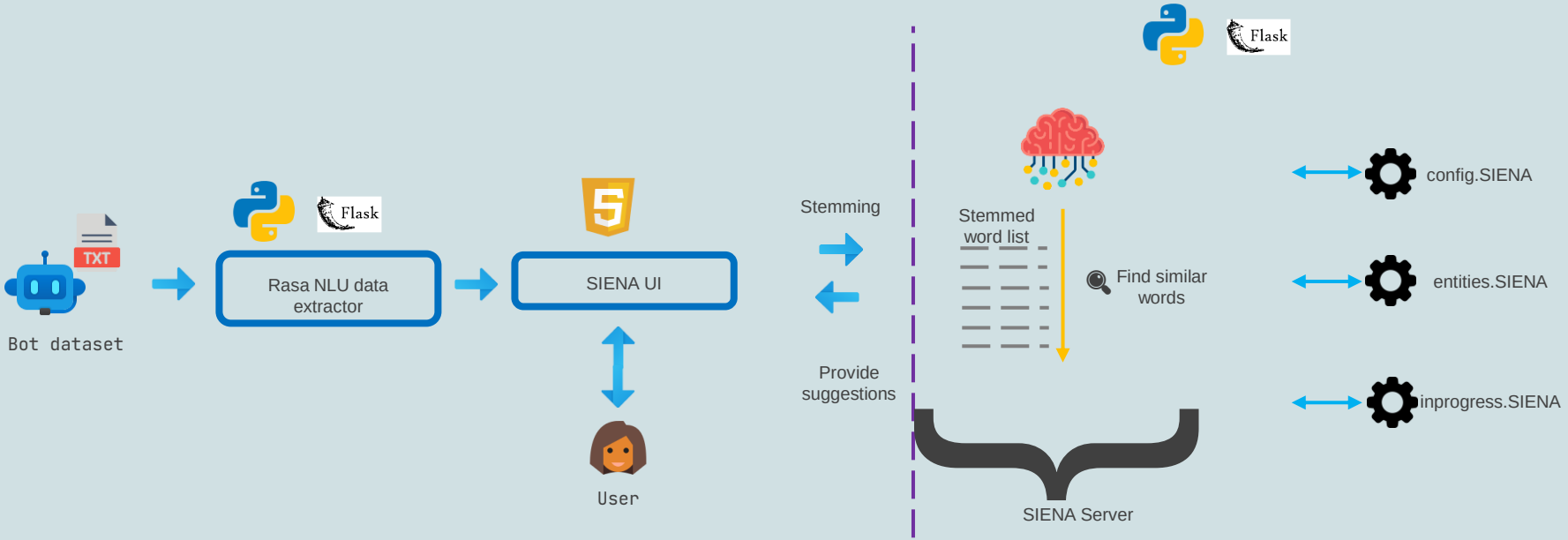
RASA spaCy

Non-functional

Requirements

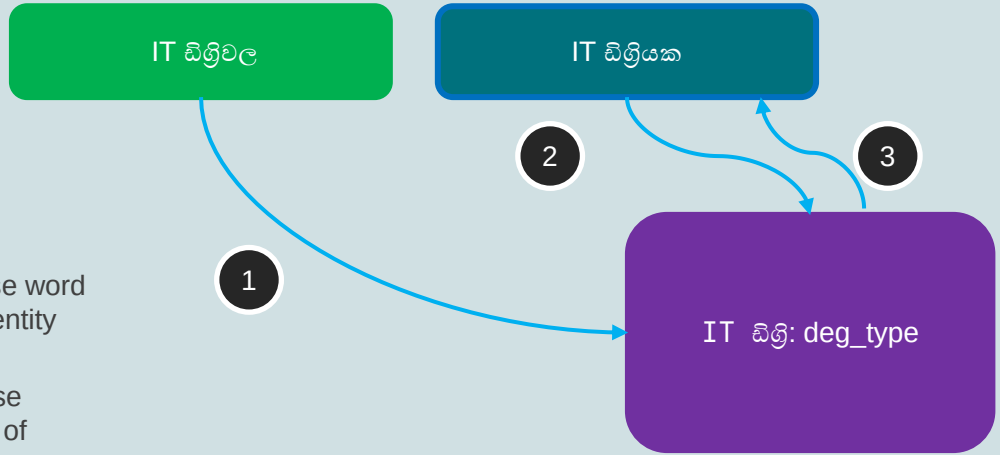
- Make as a SIENA secure software
- Able to easily maintain
- Easily install
- SIENA should be reliable
- Compatibility

Solution
Flow



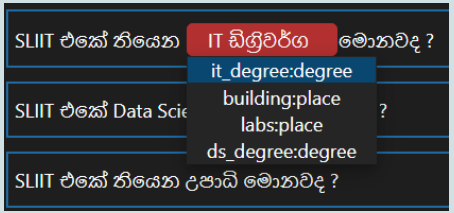
Methodology

Reverse stemming



1. Stem “IT ඩිග්‍රීවල” into base word and map it to the given entity type
2. Stem “IT ඩිග්‍රියක” into base word and get entity type of base word
3. Suggest entity type of base word to “IT ඩිග්‍රියක”

Sort recommendations according to the similarly



Highest score

Least score

Base word list

Score

Cosine similarly

N- gram similarly

Slightly different base forms due to:

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

Sub

Objectives

Define the Recommendation hierarchy (algorithms) — 100% * 0.4

Make SIENA Compatible with Frameworks — 100% * 0.2

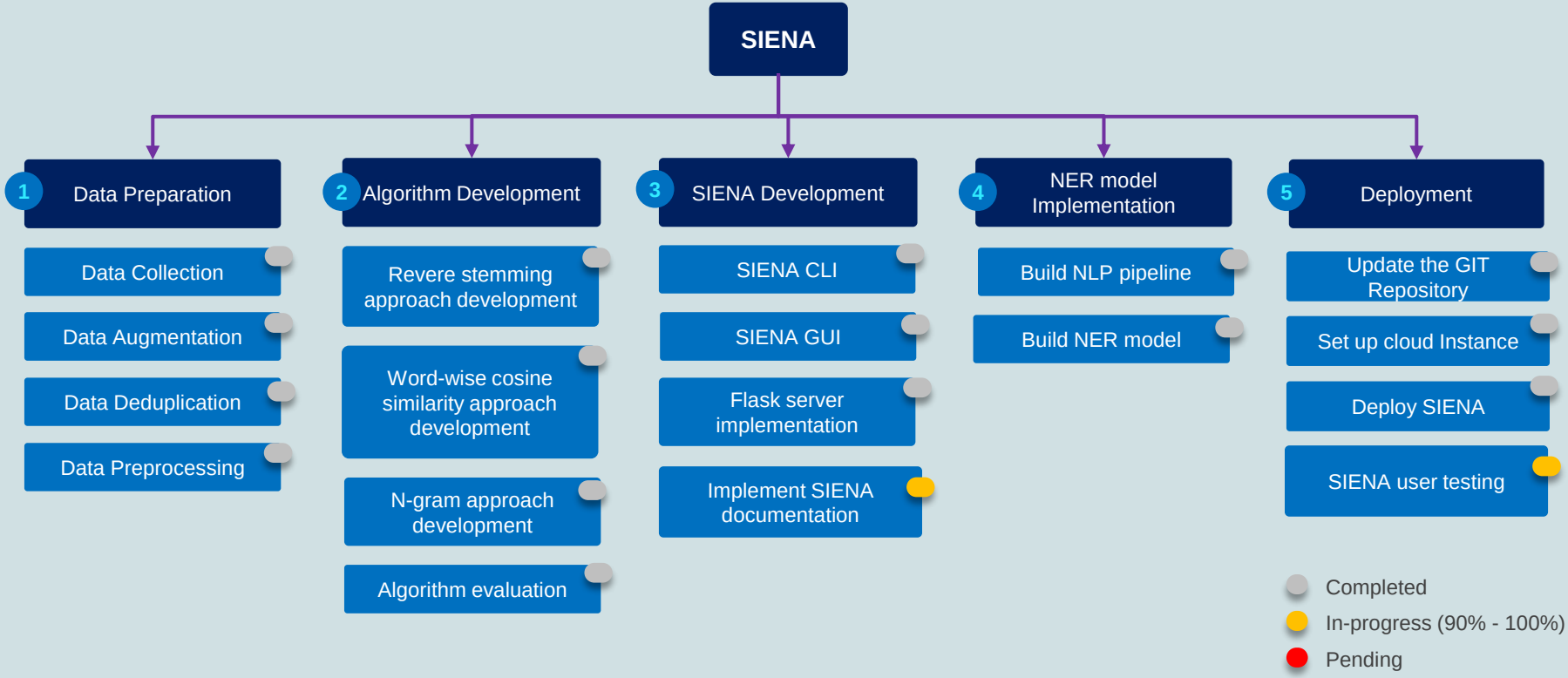
Make Knowledge base as Moduler component — 100% * 0.1

Develop Visualizations Technique to Provide User Friendly Suggestions — 94% * 0.3

Overall completion = (objective completion) x (objective weight) = 98%

Work

Breakdown Structure



Technologies

Backend Development

- Python
- pandas
- Rasa
- Flask
- Secure flask server
- .SIENA files to save configuration data

Frontend Development

- HTML

Project Management

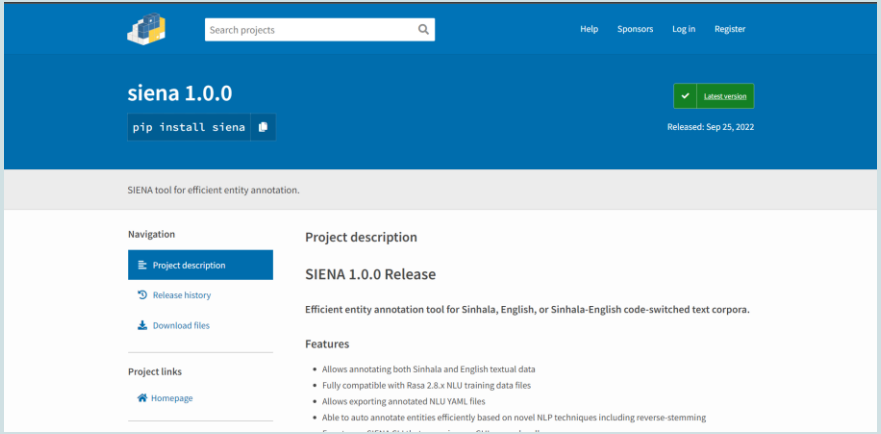
- Git
- GitLab
- GitHub
- PyPI
- Test PyPI

Best practices

Docstrings & PEP-8 Style

```python
def init_project(uploads: bool = True, exports: bool = True, cache: bool = True) -> bool:
    r"""Creates mandatory file and folders for SIENA
    """
    if uploads:
        Path("uploads").mkdir(parents=True, exist_ok=True)
    if exports:
        Path("exports").mkdir(parents=True, exist_ok=True)
    if cache:
        Path("siena_cache").mkdir(parents=True, exist_ok=True)

    fle = Path(SIENA_CONFIG_PATH)
    fle.touch(exist_ok=True)

    fle = Path(SIENA_ENTITIES_PATH)
    fle.touch(exist_ok=True)
    fle = Path(SIENA_IN_PROGRESS_PATH)
    fle.touch(exist_ok=True)

    return True
```

Available on pypi



Testing

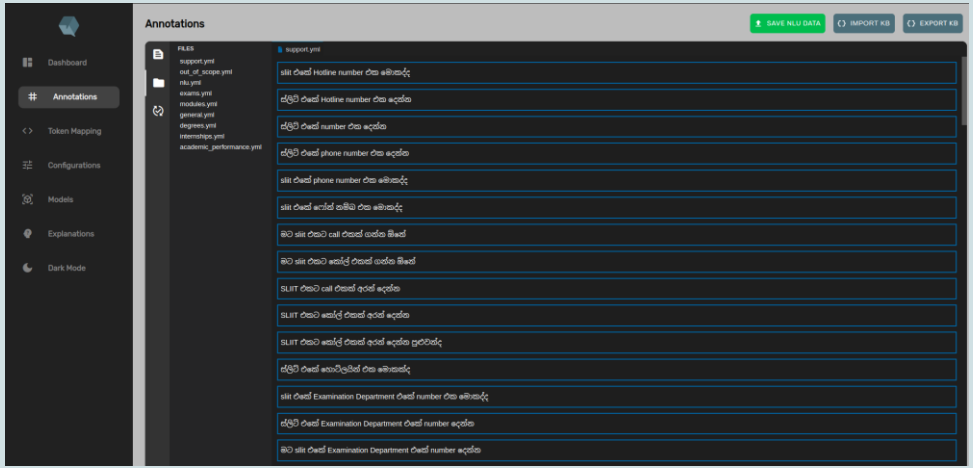
Unit testing

Integration testing

siena > test > functions.py > test_stem_word

You, 14 hours ago | 1 author (You)

```python
from siena.core.actions import (
allowed_file_nlu,
allowed_file_knowledge,
)
from siena.core.similarity import(
    si_stemmer_sentence_custom
)

def test_yaml_file_ext():
    assert allowed_file_nlu("test.yaml") == True, "It should be True"
    assert allowed_file_nlu("test.exe") == False, "It should be False"

def test_csv_file_ext():
    assert allowed_file_knowledge("sample.csv") == True, "It should be True"
    assert allowed_file_knowledge("sample.ccv") == False, "It should be False"

def test_stem_word():
    assert si_stemmer_sentence_custom("අශ්වයන්ට") == "අශ්වයි", "It should be අශ්වයි"
```



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/

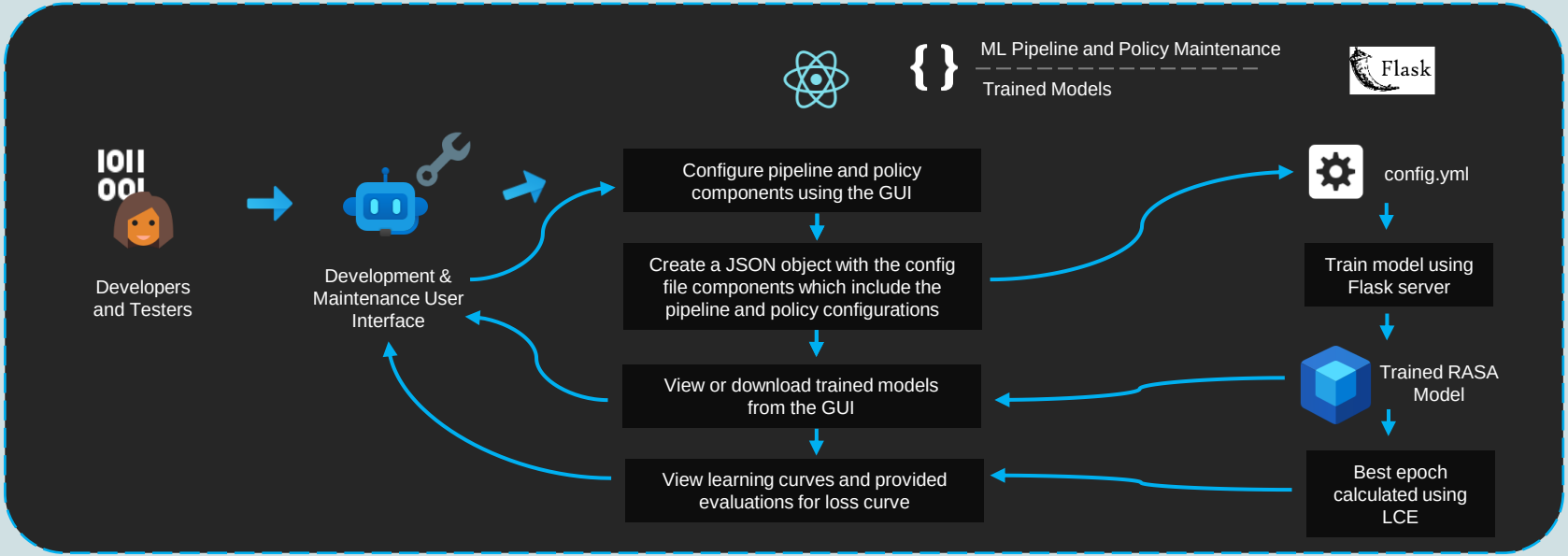
99%

Code-less Maintenance and Model Performance Evaluation

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

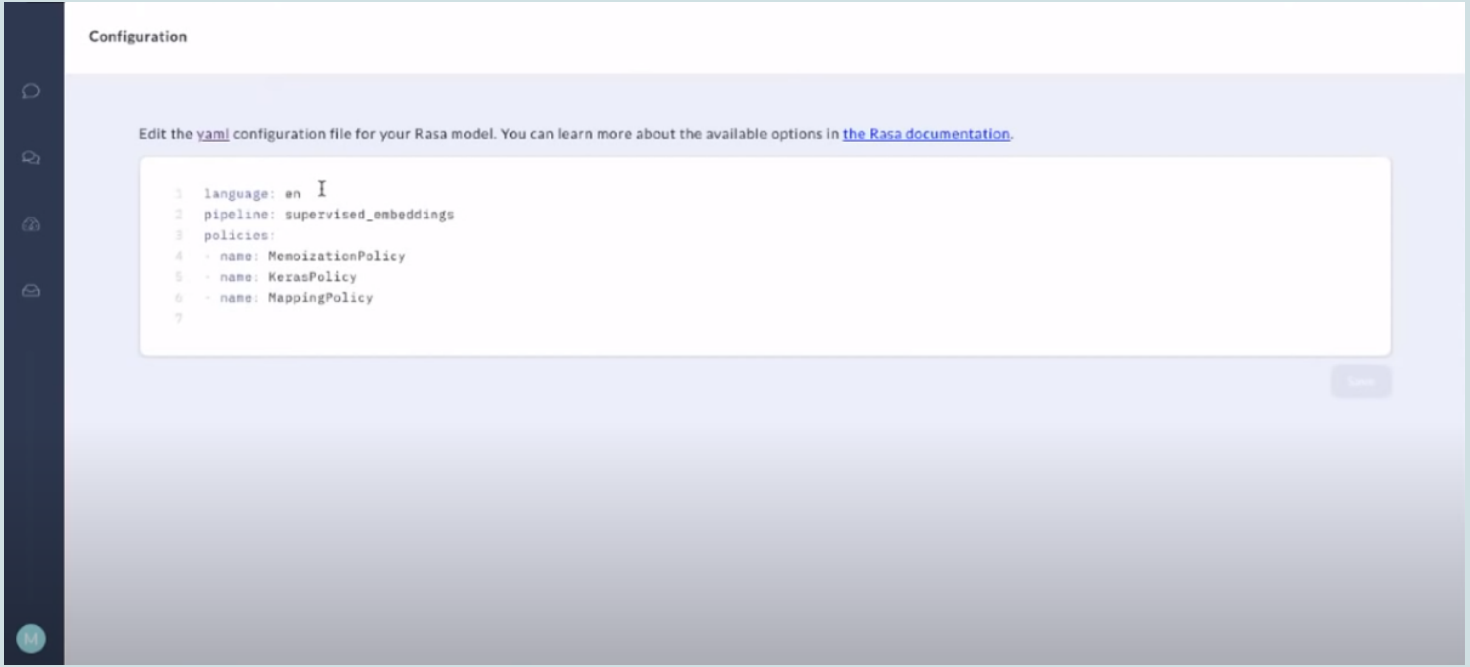
Hameed M.S.
IT19064932
Software Engineering

Implemented Research
Component Flow

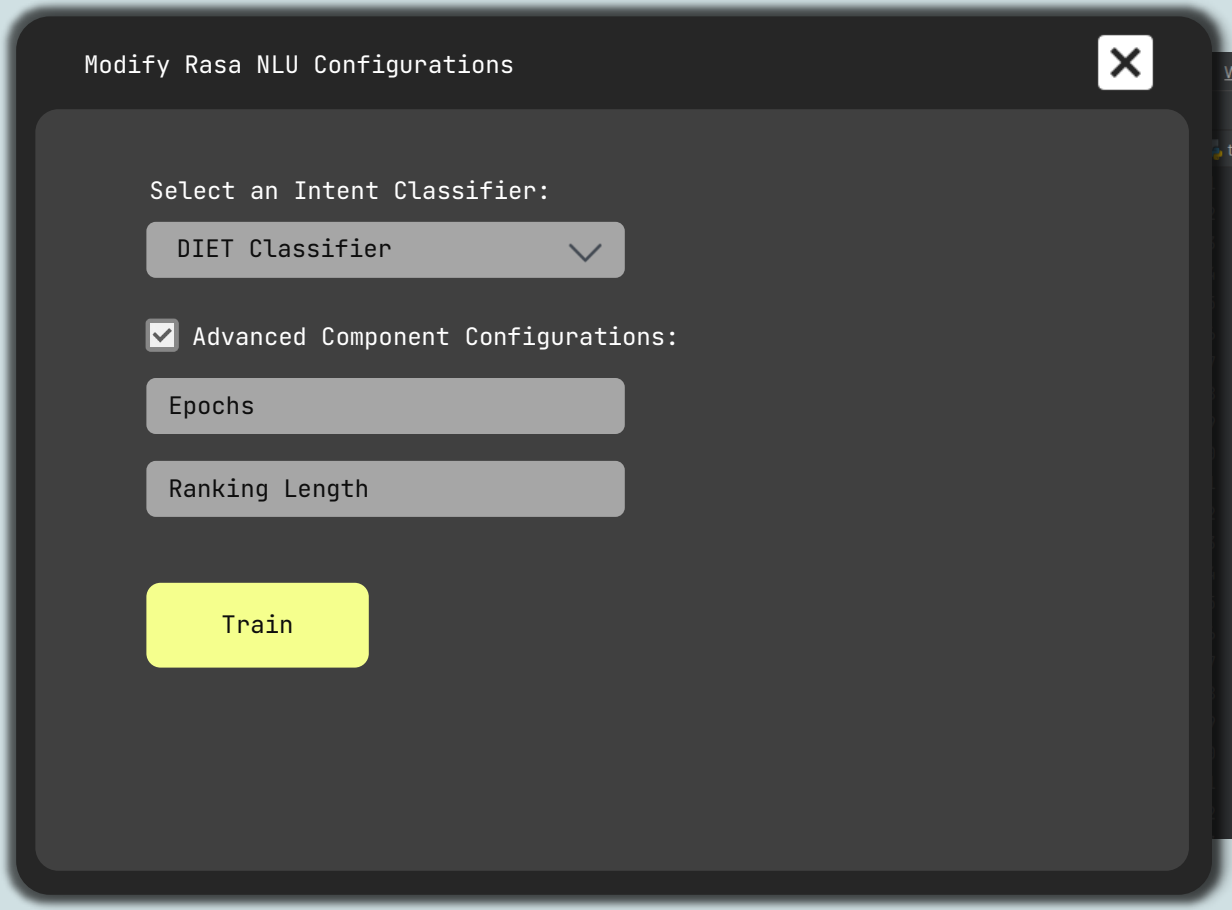


Model Improvement Existing Solution

Rasa X
Solution



Model Improvement Proposed Solution

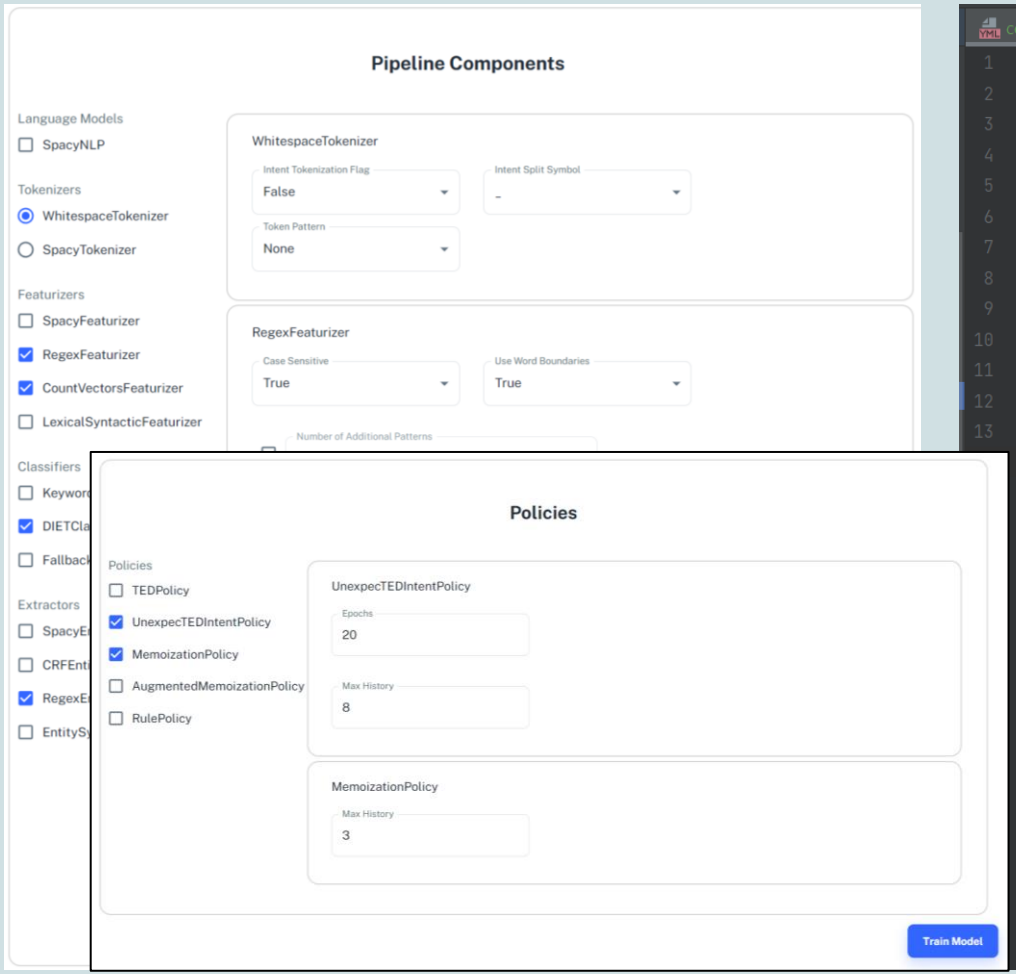


Window Help rasatest2.8.12 - config.yml

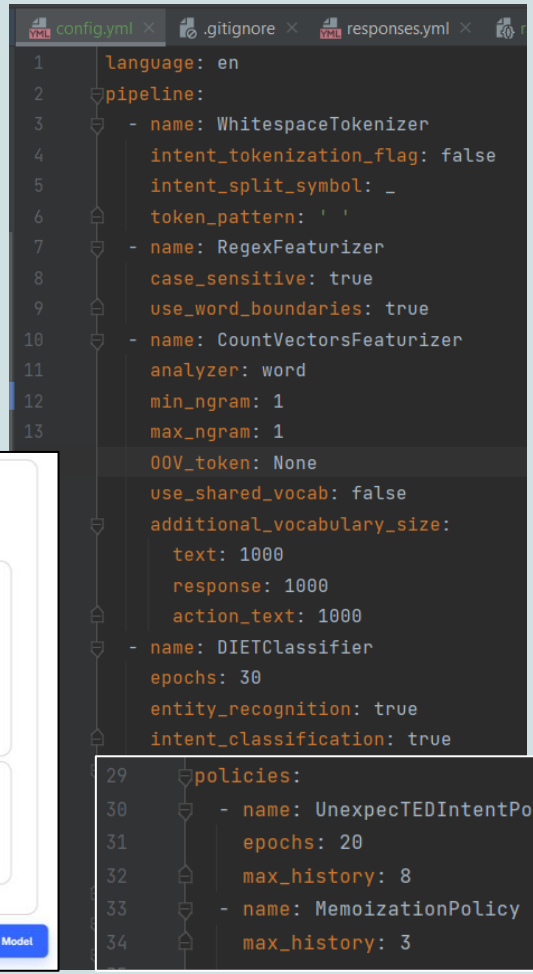
test-diet-exp.py | nlu.yml | domain.yml | config.yml

```yaml
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
```

Model Improvement Implemented Solution


Pipeline Components
Language Models
SpacyNLP
Tokenizers
WhitespaceTokenizer
SpacyTokenizer
Featurizers
SpacyFeaturizer
RegexFeaturizer
CountVectorsFeaturizer
LexicalSyntacticFeaturizer
Classifiers
Extractors
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
UnexpecTEDIntentPolicy
MemoizationPolicy
AugmentedMemoizationPolicy
RulePolicy
UnexpecTEDIntentPolicy
Epochs
20
Max History
8
MemoizationPolicy
Max History
3
Train Model


config.yml
.gitignore
responses.yml
language: en
pipeline:
- name: WhitespaceTokenizer
intent_tokenization_flag: false
intent_split_symbol: _
token_pattern: ' '
- name: RegexFfeaturizer
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

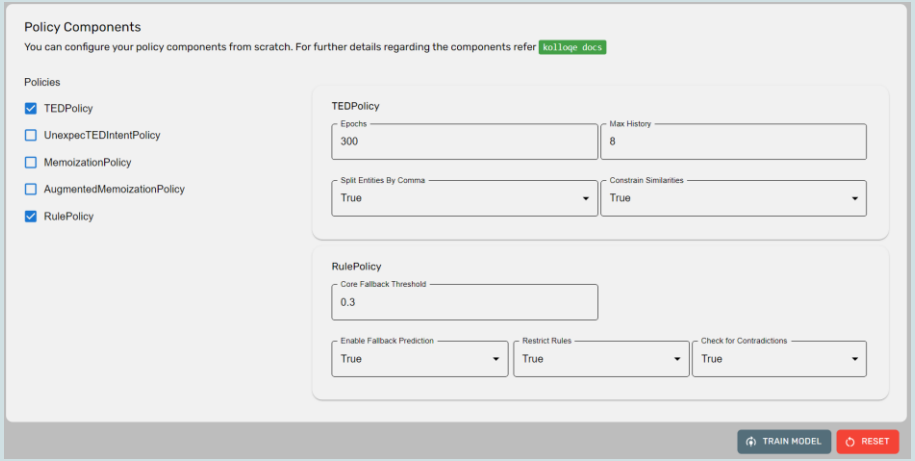
policies:
- name: UnexpecTEDIntentPolicy
epochs: 20
max_history: 8
- name: MemoizationPolicy
max_history: 3

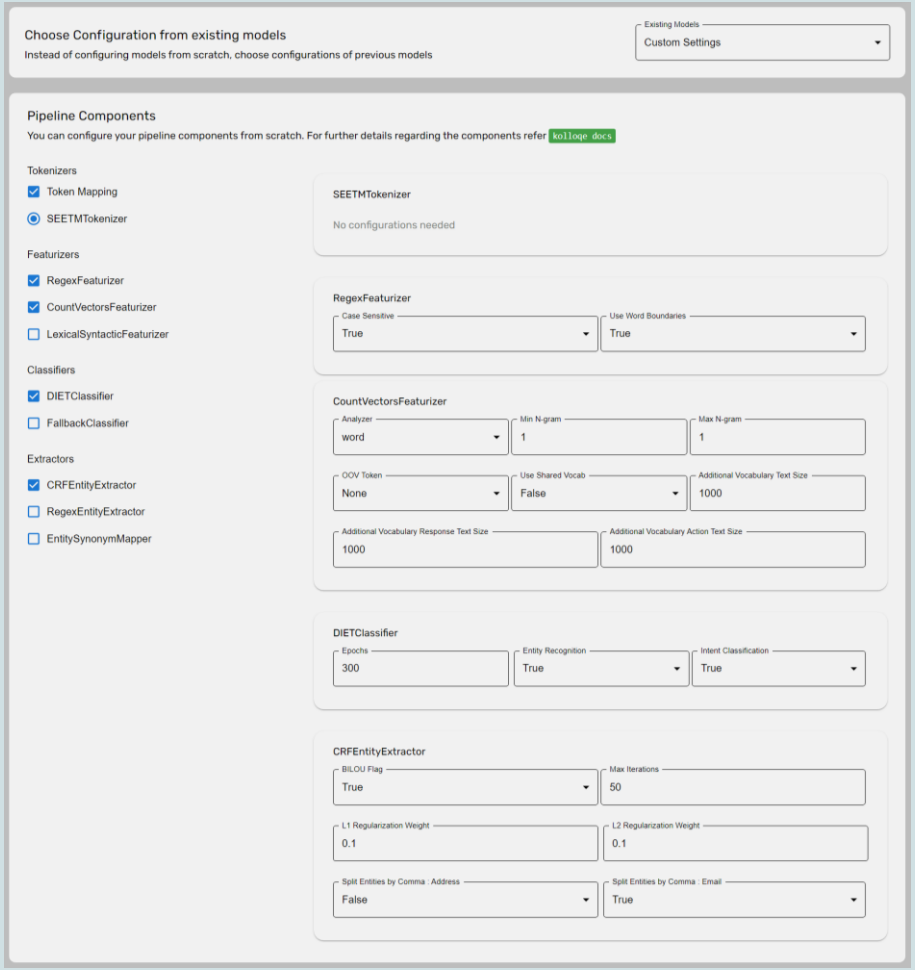
Version 2.0

Provides the option to set the pipeline and policy configurations using configurations from existing models

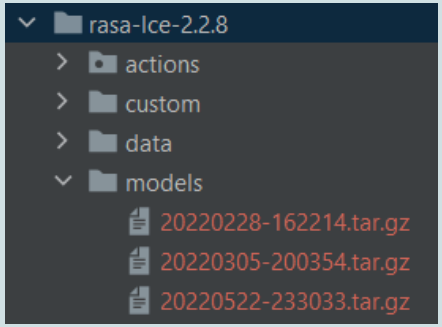
Added a custom tokenizer(SEETM) for token mapping

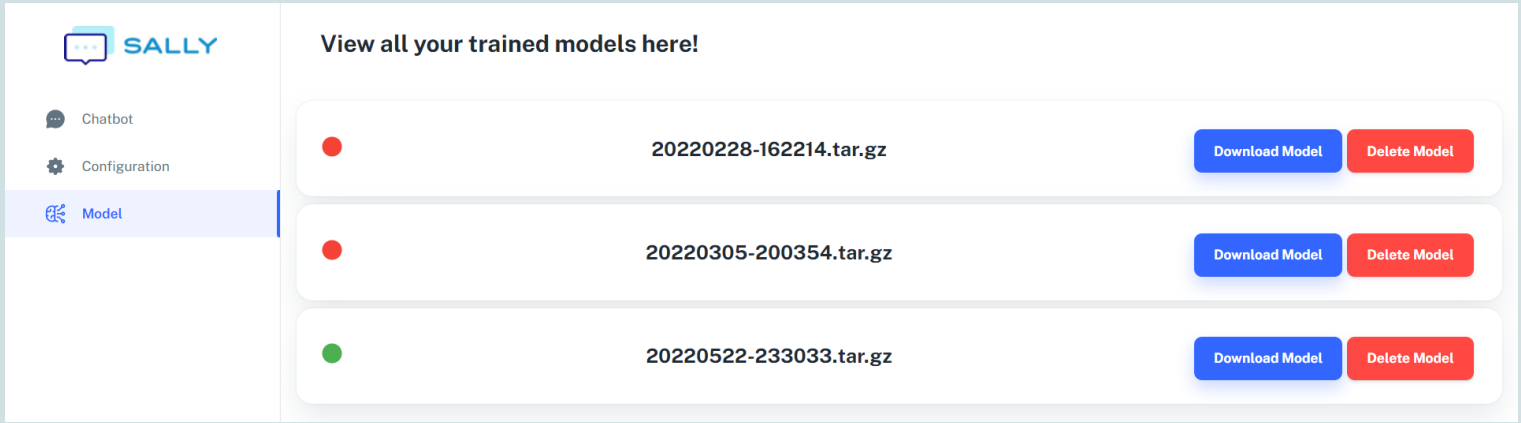
Added links to kolloqe documentation containing explanations for provided components





Trained Models

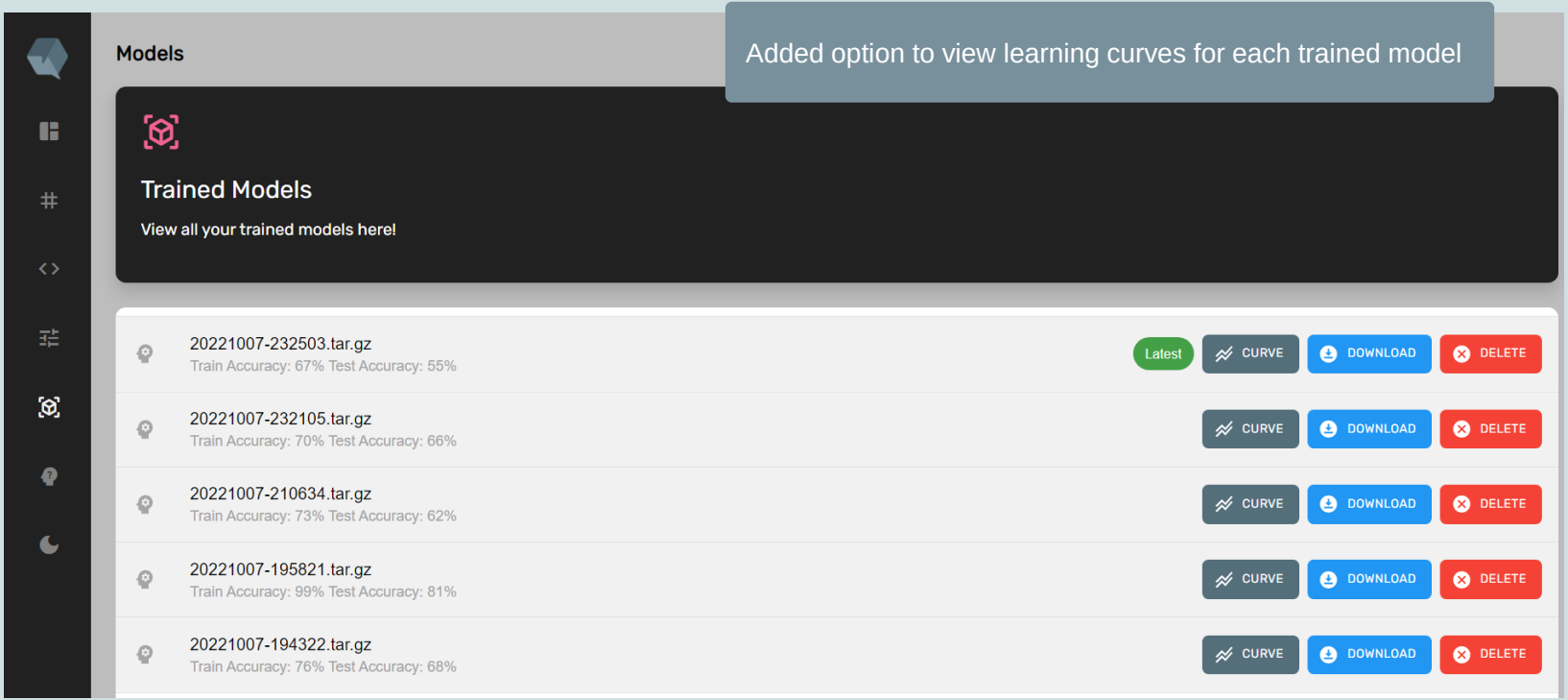




Trained Models
Version 2.0

Train and Test accuracy of each model is displayed

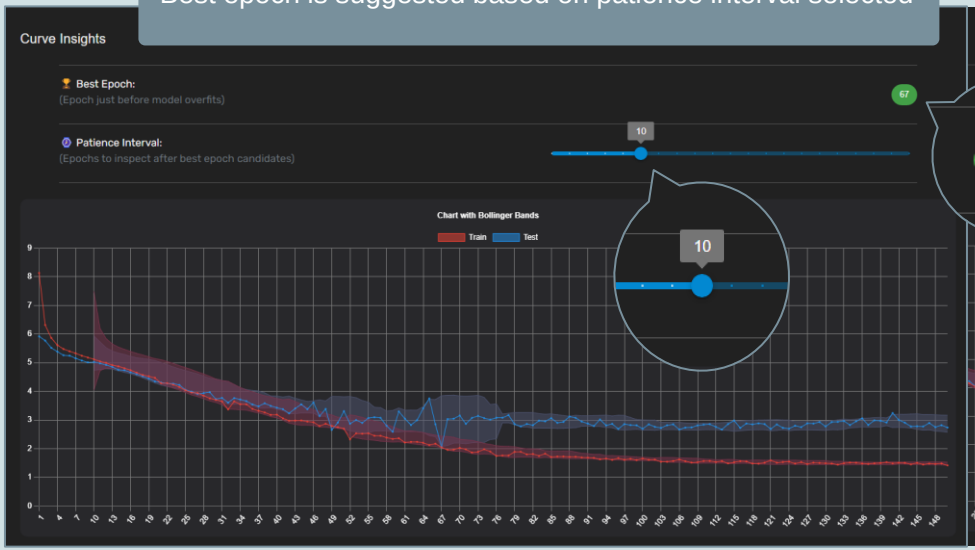
Added option to view learning curves for each trained model

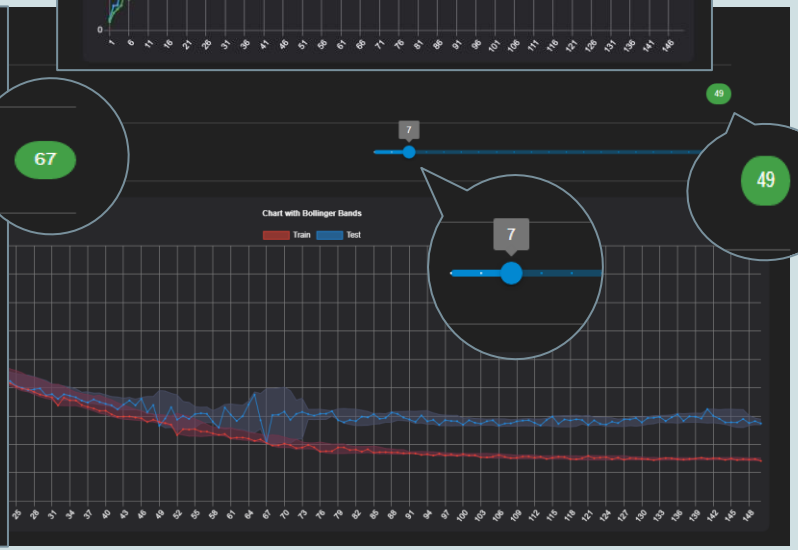


Learing Curves

Option to view accuracy and loss curve for evaluations

Best epoch is suggested based on patience interval selected





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 and evaluate learning curves of trained models
- Should be able to suggest the best epoch to train a model after evaluating the loss curve of a trained model.

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

1

Configuration.jsx | configs.js | Curve.jsx | Model.jsx | Dashboard.jsx

src > configs.js > configs

You, 3 days ago | 2 authors (Ishara Dissanayake and others)

```
let api = ""; //"http://localhost:6070";

let kolloqeDocsHost = "https://docs.kolloqe.com";

export const configs = {
  api: api,
  snackbarVerticalPosition: "bottom",
  snackbarHorizontalPostion: "left",

  // DIME
  explainEndpoint: `${api}/api/dime/explain`,
  abortExplainEndpoint: `${api}/api/dime/abort`,
  dimeStatsEndpoint: `${api}/api/dime/stats`,
  modelEndpoint: `${api}/api/dime/model`,
  explanationEndpoint: `${api}/api/dime/explanation`,
  visualizationEndpoint: `${api}/api/dime/explanation/visualize`,
  dimeConfigEndpoint: `${api}/api/dime/configs`,

  // RASAC
  trainModelEndpoint: `${api}/api/rasac/bot/train`,
  abortTrainEndpoint: `${api}/api/rasac/bot/abort`,
  getModelListEnpoint: `${api}/api/rasac/botstore/models`,
  getModelCurveDatapointsEndpoint: `${api}/api/rasac/botstore/curve/`,
```

2

```
axios
  .get(`${configs.getModelListEnpoint}`, {
    headers: {
      "Content-Type": "application/json",
    },
  })
  .then(async (res) => {
    const data = res.data;

    if (Object.hasOwn(data, "status")) {
      // error has occcured
      openGetModelFailAlert(true);
    } else {
      // no error
      if (data["model_list"] === null) {
        openGetModelFailAlert(true);
      } else {
        setTrainedModels(data["model_list"]);
        setLatestModel(data["latest_model"]);
```

Best Practices

Maintain a constants file for the back-end

1

requirements-kolloqe.txt | kolloqe.py | constants.py | KOLLOQEBUILD.md | .env | local

```python
import ...

PACKAGE_NAME = "kolloqe"
PACKAGE_VERSION = "0.0.1a4"
LOGGING_FORMAT_STR = '%(asctime)s\t%(levelname)s\t%(name)s - %(message)s'


# server env
class ServerEnv:
    STRICT_LOCAL = "strict_local"  # triggers server explanations without subprocess
    DEV = "dev"  # enables server debugging
    PROD = "prod"  # disables server debugging
    ENV_LIST = [DEV, PROD, STRICT_LOCAL, "Dev", "StrictLocal", "Prod"]


class Theme:
    DARK = "dark"
    LIGHT = "light"
    THEME_LIST = [DARK, LIGHT, "Dark", "Light"]
```

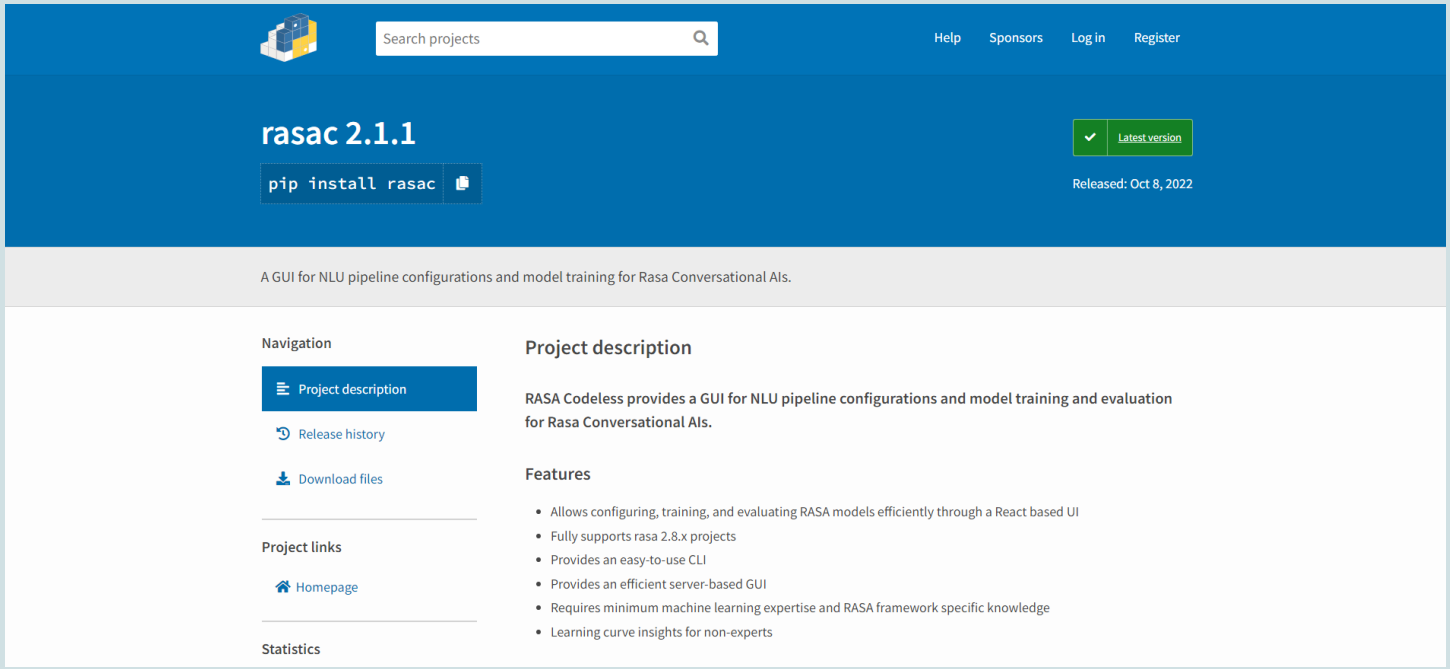
2

requirements-kolloqe.txt | kolloqe.py | constants.py | KOLLOQEBUILD.md | .env | local_botstore.py

```python
    def _initialize_env(self) -> NoReturn:
        # primary env vars
        try:
            host = os.environ.get("KOLLOQE_HOST")
            port = os.environ.get("KOLLOQE_PORT")
            theme = os.environ.get("APP_THEME")
            sinhala = BoolStr.to_bool(value=str(os.environ.get("SINHALA_ENABLED")).lower())
            env = os.environ.get("APP_ENV")
            secure_url = BoolStr.to_bool(value=str(os.environ.get("SECURE_BOT_URL")).lower())

            self.host = str(host).lower() if host else KOLLOQE_HOST
            self.port = int(port) if port else KOLLOQE_PORT
            self.theme = str(theme).lower() if theme in Theme.THEME_LIST else self.theme
            self.sinhala = sinhala
            self.env = str(env).lower() if env in ServerEnv.ENV_LIST else None
            self.secure_url = secure_url

            if not self.env or not self.host or not self.port:
                raise KolloqeEnvException()
```

Additional Work

Publishing individual component PyPi



Test Classes


```
// Unit Testing
test("curveExplainer", () => {
  const curveExplainer = jest.fn((patienceInter, train_loss, test_lo
    var trainUpper = 0;
    var trainLower = 0;
    var testUpper = 0;
    var testLower = 0;

    const trainBollinger = boll(train_loss, patienceInter, 2);
    trainUpper = trainBollinger.upper;
    trainLower = trainBollinger.lower;

    const testBollinger = boll(test_loss, patienceInter, 2);
    testUpper = testBollinger.upper;
    testLower = testBollinger.lower;

    let tempArray = [];
    let minLoss = test_loss[0];
    let bestEpoch = 1;

    for (let i = 0; i < train_loss.length; i++) {
      if (i < patienceInter - 1) {
        if (test_loss[i] < minLoss) {      You, 3 days ago • added
          minLoss = test_loss[i];
          bestEpoch = i + 1;
        }
      } else {
        if (test_loss[i] < trainBollinger.upper[i]) {
          if (test_loss[i] < minLoss) {
            minLoss = test_loss[i];
            bestEpoch = i + 1;
          }
```

OffCanvasExplanation.jsx | Model.jsx | Curve.jsx | Curve.test.js M | package.json | app_styles.c

src > components > curve > Curve.test.js > test("curveExplainer") callback > curveExplainer > jest.fn() callback

```
304             break;
305           }      You, 2 days ago • added test for curve explainer
306         }
307       }
308
309       if (test_loss[i] < minLoss) {
310         minLoss = test_loss[i];
311         bestEpoch = i + 1;
312       }
313     }
314
315     return bestEpoch;
316   });
317
318   // testing for the best epoch on 1st dataset with varying patience intervals of 5, 9, and 21 with
319   expect(curveExplainer(5, train_loss_1, test_loss_1)).toBe(31);
320   expect(curveExplainer(9, train_loss_1, test_loss_1)).toBe(31);
321   expect(curveExplainer(21, train_loss_1, test_loss_1)).toBe(47);
322   expect(curveExplainer(25, train_loss_1, test_loss_1)).toBe(47);
323 });
324
```

TERMINAL | GITLENS | PROBLEMS | OUTPUT | DEBUG CONSOLE | JUPYTER

```
PASS src/components/curve/Curve.test.js
  √ curveExplainer (14 ms)

Test Suites: 1 passed, 1 total
Tests:       1 passed, 1 total
Snapshots:   0 total
Time:        3.027 s
Ran all test suites related to changed files.

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

Sub Objectives

95% * 0.1

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

100% * 0.6

Developing a solution for non-technical users to view all train models and evaluate them

100% * 0.3

Developing an algorithm to identify what the best epoch is to train a model and suggesting it in the UI.

Overall completion = (objective completion) x (objective weight) = 99%

Work Breakdown Structure

1 Data Preparation

- Data Collection
- Data Augmentation
- Data Deduplication
- Data Preprocessing

2 Building Conversational AI

- Add Training Data
- Build NLU pipeline
- Decide number of validation data points
- Enable TensorBoard
- Train a model
- Testing NLU Model

3 Algorithm Development

- Read TensorBoard Results
- Develop Learning Curve Explainer algorithm to interpret TensorBoard Results
- Build the LCE Package
- Build a Locally executable Server

4 Implementation

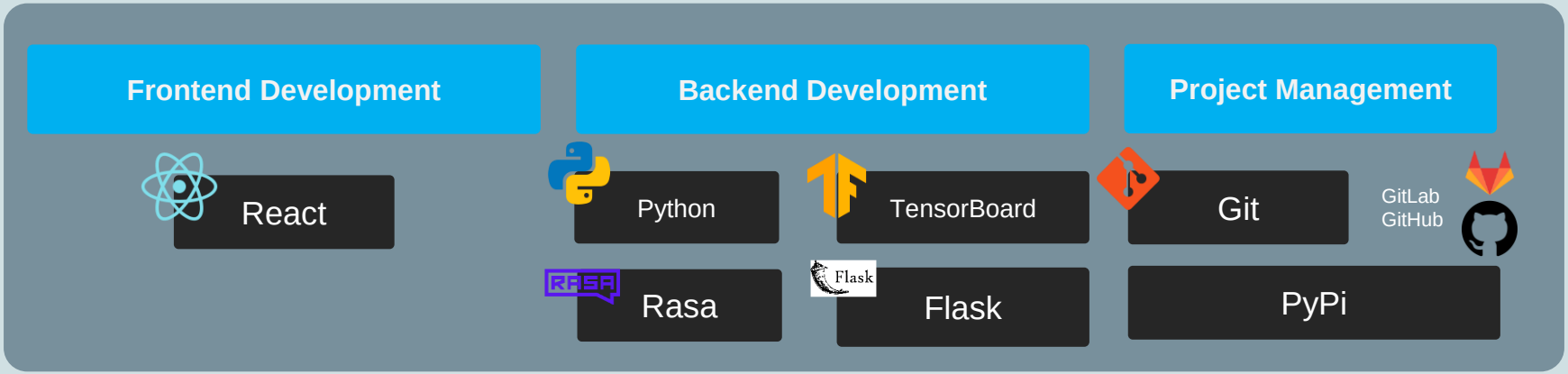
- Integrate LCE with Rasa
- Build the Frontend
- Implement code-less Maintenance
- Build Docker & Docker-compose files
- Build Caddy and NGINX Servers
- Integrate component UI with the Main UI
- Implement kolloqe documentation

5 Deployment

- Update the GIT Repository
- Set up Cloud Instance
- Deploy Docker Containers
- Test LCE and Code-less Maintenance after Deployment

- Completed
- In-progress (90% - 100%)
- Pending

Technologies



References

[1]. T. Bocklisch, J. Faulkner, N. Pawlowski, en A. Nichol, “Rasa: Open source language understanding and dialogue management”, *arXiv preprint arXiv:1712. 05181*, 2017.

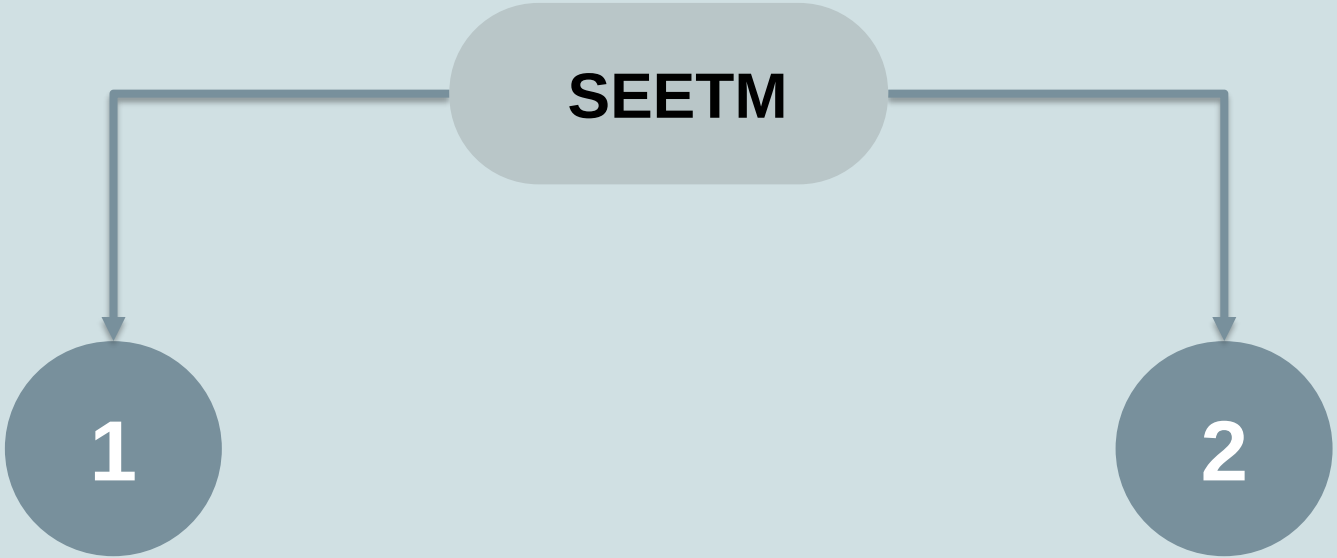
[2]. “Introduction to rasa X,” *Open source conversational AI*, 10-Dec-2021. [Online]. Available: https://rasa.com/docs/rasa-x/. [Accessed: 22-Jan-2022].

97.5%

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


SEETM
1
2

Problem: 1

How can I type Sinhala and English words together?

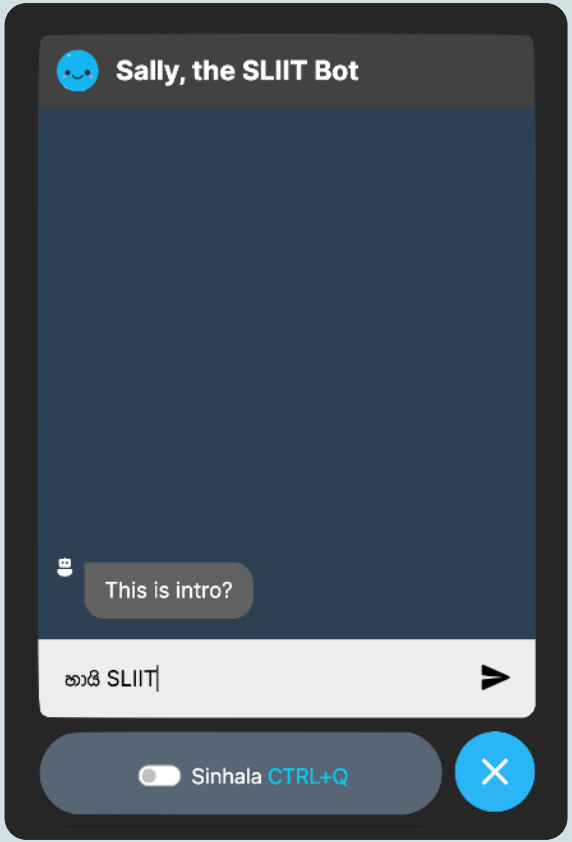
No Sinhala English Code-switching Typing Support in Chatbots!

1 Solution Found !

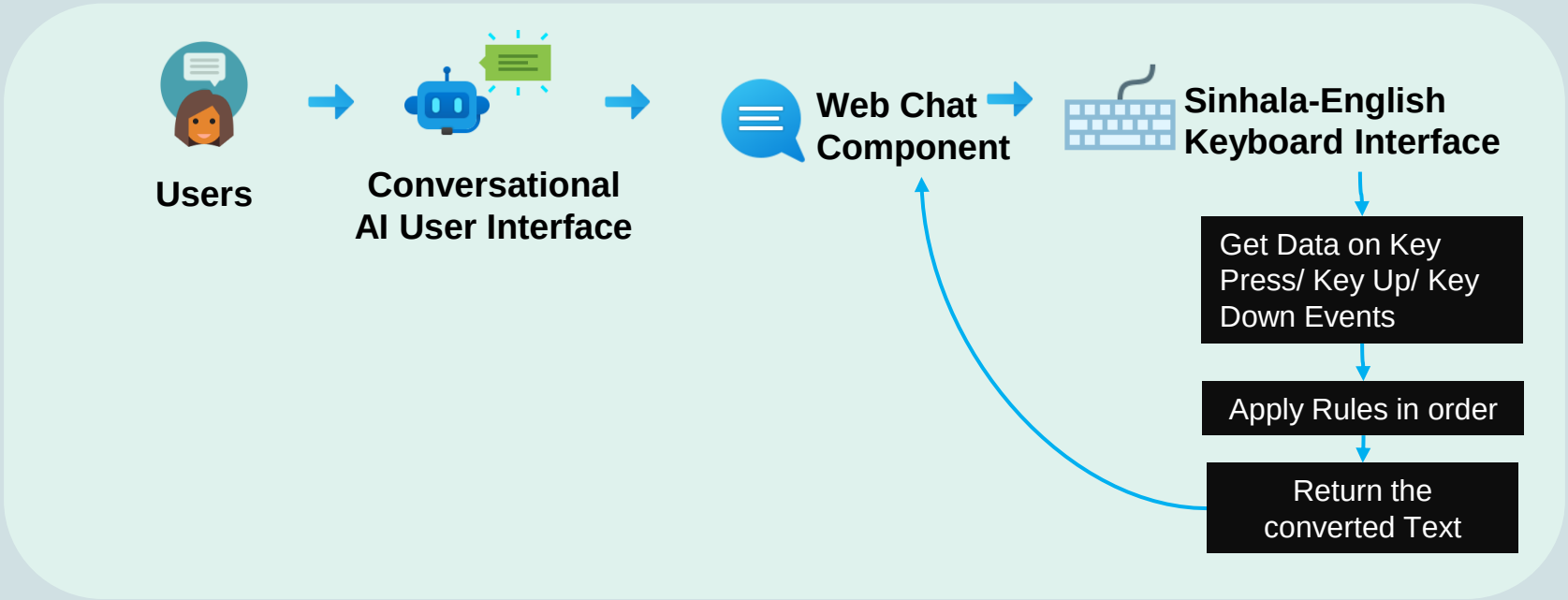
Sinhala-English code-switchable Keyboard Interface

Code-Switching

Library Membership
එකට apply
කරන්නේ කොහොමද?



Research
Flow



Keyboards
Comparison

Keyboard Name	Default (Misleading character Map)	Changes
Helakuru Keyboard (හෙළකුරු)	• Backspace + a → ා • c → ක් • u → ෞ (when re-enter) • ou → ඕ • Q/q → ද්	• Backspace + a → අ • c → ච් • u → උ • au → ඕ • Q/q → (Shortcut) (dh → ද්)

Keyboards
Comparison

Keyboard Name	Default (Misleading character Map)	Changes
UCSC Unicode Real-Time Font Converter	• c → c • q → ඣ • Q → Q • U → U • V → V • E → E • F → F • H → H • M → M • O → O • S → S • W → W • x, X → x • z→ z	• c → ච • Q/q → (Shortcut) • U → උ • V → ව • E → ඓ • F → ෆ් • H → ඃ • M → ම • O → ඔ෭ • S → ස් • X → ඩ • x → ං • W → ව • z → ්

Keyboard Interface Ruleset

Rule Activation

Rule		English Representation: aakramaNaya	
		Character Mapping	**Output**
1.	Non-joining Character Mappings	skipped as there are no non-joining characters	aakramaNaya
2.	Special Character Mappings	skipped as there are no special characters	aakramaNaya
3.	“Rakaranshaya” Mapping	kra → ක් + ර් + අ → ක්‍ර	aaක්‍රmaNaya
4.	Consonant + Vowel Mappings	ma → ම් + අ → ම Na → ණ් + අ → ණ ya → ය් + අ → ය	aaක්‍රමNaya aaක්‍රමණya aaක්‍රමණය
5.	Pure Consonant Mappings	skipped as there are no pure consonants left	aaක්‍රමණය
6.	Pure Vowel Mappings	aa → ආ	ආක්‍රමණය

```
[105]  convert("aakramaNaya")
```

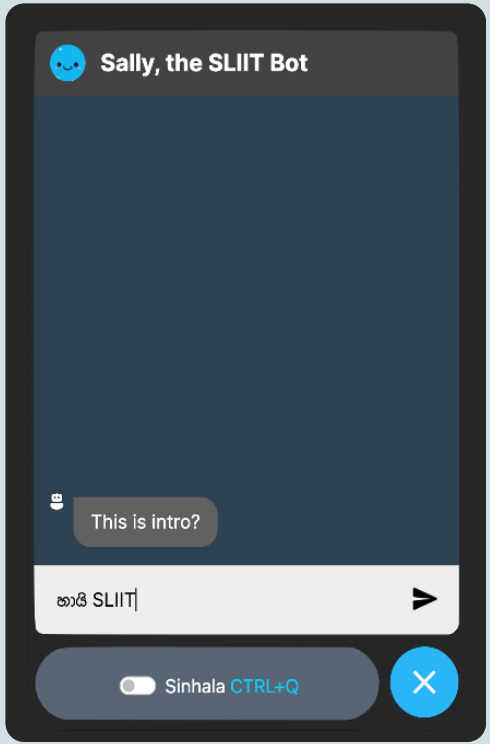
```
Rule 3.1 [Detected]: aakramaNaya
Rule 3.1 [Replaced]: aaක්‍රmaNaya
Rule 4 [Detected]: aaක්‍රmaNaya
Rule 4 [Replaced]: aaක්‍රmaණya
Rule 4 [Detected]: aaක්‍රmaණya
Rule 4 [Replaced]: aaක්‍රමණya
Rule 4 [Detected]: aaක්‍රමණya
Rule 4 [Replaced]: aaක්‍රමණය
Rule 6 [Detected]: aaක්‍රමණය
Rule 6 [Replaced]: ආක්‍රමණය
'ආක්\u200dරමණය'
```

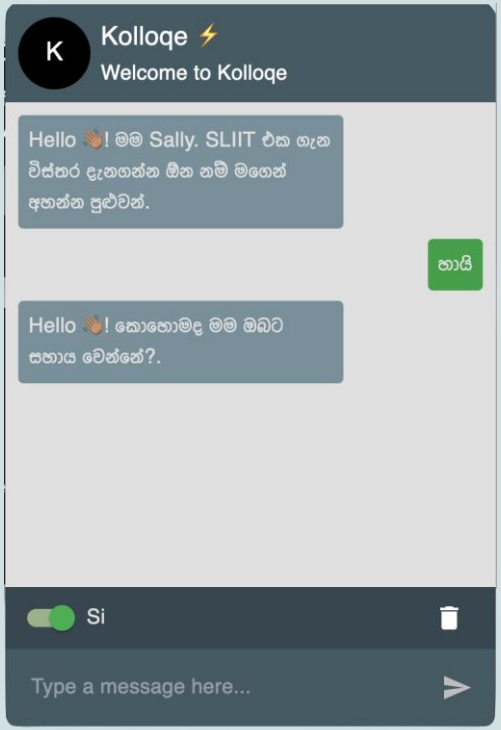
ආක්‍රමණය

React based
Keyboard Interface

Keyboard works on chatbot

```jsx
import { Input, TextField } from "@mui/material";
import React, { Component } from "react";
import PropTypes, { number, string } from "prop-types";

export default class KeyboardInput extends Component {
  constructor(props) {
    super(props);
    this.state = {
      language: this.props?.defaultLanguage,
      text: this.props?.value || "",
      previousText: "",
      newText: "",
      outOfScopeText: this.props?.value || "",
      shortcutKeyStore: {},
    };

    // refs
    const utilizeFocus = () => {
      const ref = React.createRef();
      const setFocus = () => {
        ref.current && ref.current.focus();
      };
      return { setFocus, ref };
    };
    this.inputRef = utilizeFocus();

    // handlers
    this.handleInputChange = this.handleInputChange.bind(this);
    this.handleInputKeyDown = this.handleInputKeyDown.bind(this);
    this.handleInputKeyUp = this.handleInputKeyUp.bind(this);
    this.handleInputKeyPress = this.handleInputKeyPress.bind(this);
    this.handleInputReset = this.handleInputReset.bind(this);
    this.handleInputFocus = this.handleInputFocus.bind(this);
    this.handleConvert = this.handleConvert.bind(this);
    this.handleLanguage = this.handleLanguage.bind(this);
    this.setText = this.setText.bind(this);
  }

  componentDidMount() {
```





Problem: 2

Equivalent words in Code-Switching datasets

IT	Information technology
අයිටී	තොරතුරු තාක්ෂණය

isn’t this similar?
ML models see equivalent words as different features!

2 Solution Found !

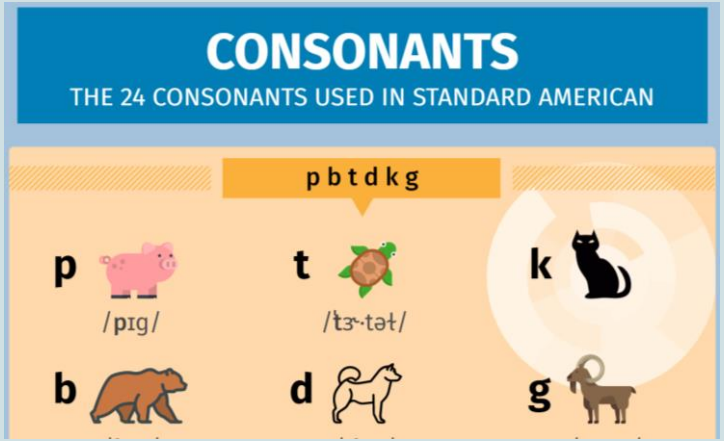
Translating approach ❌

Pronunciation based approach ❌

1 2 IPA Mapping

eɪ ඒ	aɪ අයි	aʊ අව්	ɔ අව්	ɔɪ ඕ	oʊ ඕ	æ ඇ		
l ඊ	u ඌ	ɪ ඉ	ʊ උ	ə අ	ɛ එ	ɝ අර්	æ ඇ	ɑ ආ
tʃ ච්	dʒ ජ්	p ප්	b බ්	t ට්	d ඩ්	k ක්	g ග්	θ ත්
ð ද්	f ෆ්	v ව්	s ස්	z ස්	ʃ ශ්	ʒ ෂ්	w ව්	m ම්
n න්	ŋ න්ග්	ɹ ර්	r ර්	j ය්	h හ්	ɬ ල්	l ල්	

https://americanipachart.com/



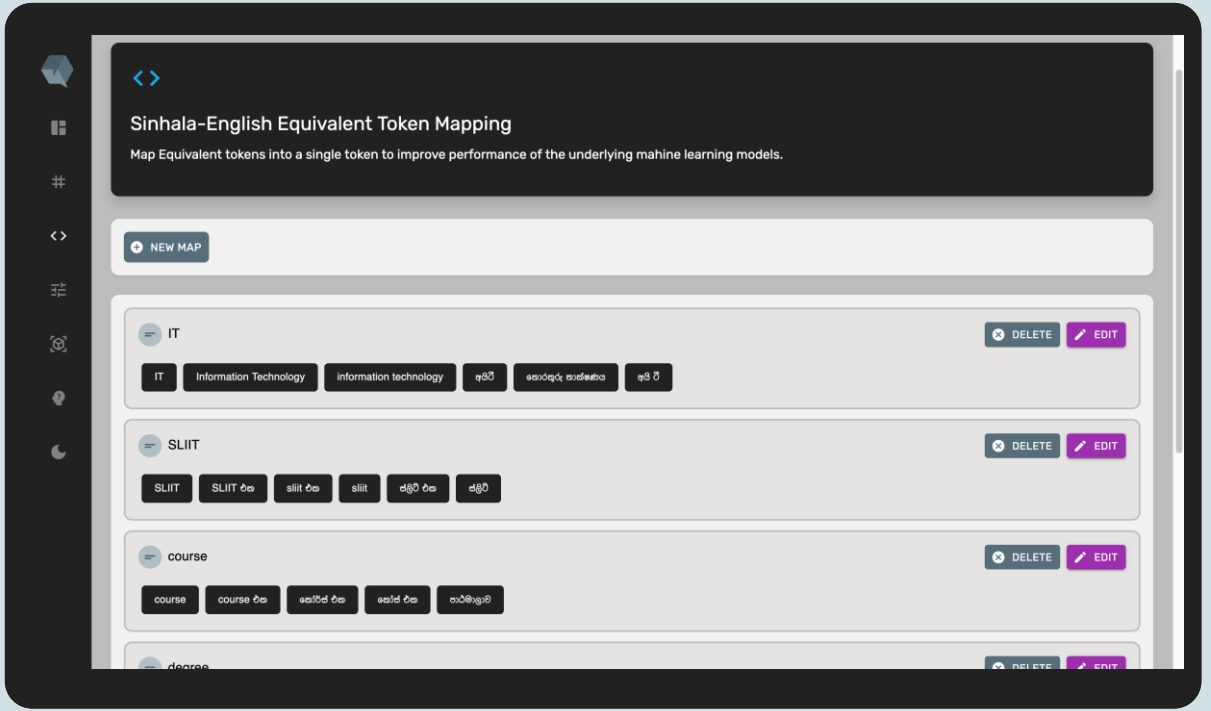
Translating approach 

Pronunciation based approach 

User defined rule-based map 

Equivalent Token Mapping

Rule	Equivalent Tokens
උපාධි	උපාධි, degrees, ඩිග්‍රි, ඩිග්‍රිස්, ඩිග්‍රී, ඩිග්‍රීස්
sliit	sliit, SLIIT, ශ්‍රී ලංකා තොරතුරු තාකෂණ ආයතනය, ස්ලිට්, ස්ලිට් එක



Token Mapping In action

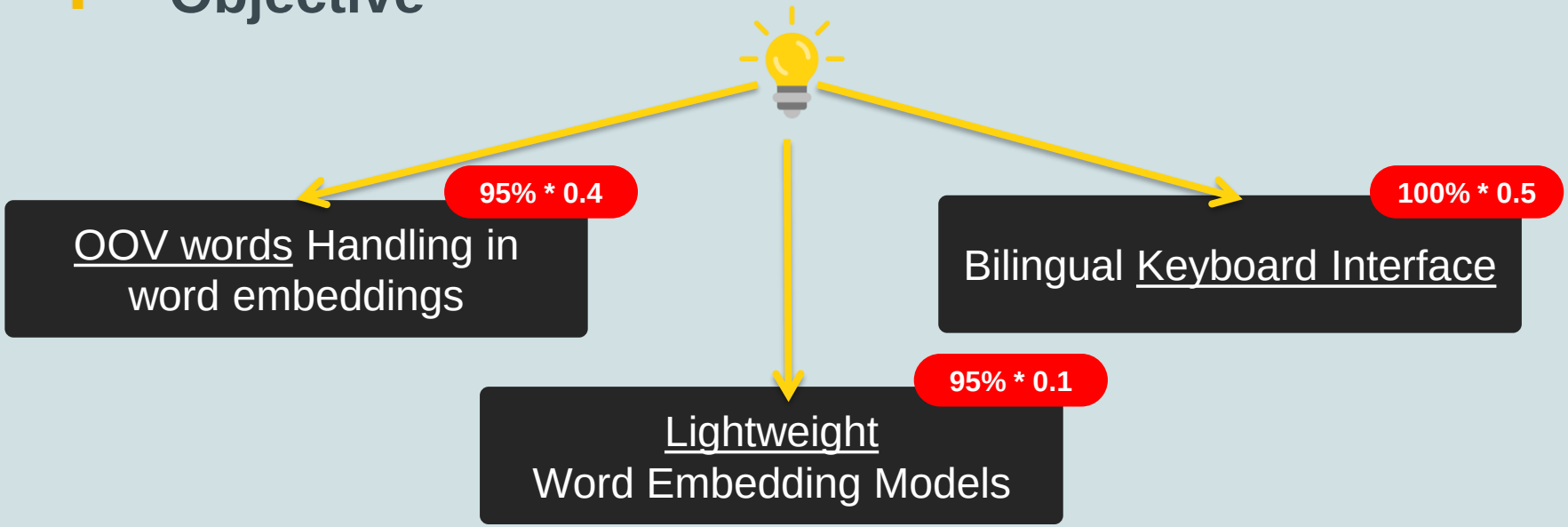
```
2022-10-11 13:06:12 INFO     rasa.shared.nlu.training_data.training_data  - Number of response exa
2022-10-11 13:06:12 INFO     rasa.shared.nlu.training_data.training_data  - Number of entity examp
2022-10-11 13:06:12 INFO     rasa.nlu.model  - Starting to train component SEETMTokenizer
2022-10-11 13:06:12 INFO     kolloqe_components.tokenizers.seetm_tokenizer  -
Persisted:
Source: ඔන්ලයින් වයිවා තියෙනවද
Mapped:ඔන්ලයින් viva තියෙනවද
2022-10-11 13:06:12 INFO     kolloqe_components.tokenizers.seetm_tokenizer  -
Persisted:
Source: ඔයාලගෙ තියෙන IT ඩී ග්‍රී වර්ග මොනවද ?
Mapped:ඔයාලගෙ තියෙන IT degree වර්ග මොනවද ?
2022-10-11 13:06:12 INFO     kolloqe_components.tokenizers.seetm_tokenizer  -
Persisted:
Source: ඕගොල්ලන්ගෙ තියෙන business ඩී ග්‍රී මොනවද ?
Mapped:ඕගොල්ලන්ගෙ තියෙන business degree මොනවද ?
2022-10-11 13:06:12 INFO     kolloqe_components.tokenizers.seetm_tokenizer  -
Persisted:
Source: තියෙන ඩී ග්‍රී වර්ග මොනවද ?
Mapped:තියෙන degree වර්ග මොනවද ?
2022-10-11 13:06:12 INFO     kolloqe_components.tokenizers.seetm_tokenizer  -
Persisted:
Source: ස්ලි ට් එකේ ඩී ග්‍රී වල තියෙන ස්පෙෂලයිසේෂන්ස් මොනවද
Mapped: ස්ලි ට් එකේ degree වල තියෙන ස්පෙෂලයිසේෂන්ස් මොනවද
2022-10-11 13:06:12 INFO     kolloqe_components.tokenizers.seetm_tokenizer  -
Persisted:
Source: හැමෝම Foundation course එක කරන්න ඕනෙද
Mapped:හැමෝම Foundation course කරන්න ඕනෙද
2022-10-11 13:06:12 INFO     kolloqe_components.tokenizers.seetm_tokenizer  -
Persisted:
Source: Foundation course එක කරන ගමන් ම degree එක කරන්න පුලුවන්ද
Mapped:Foundation course කරන ගමන් ම degree කරන්න පුලුවන්ද
2022-10-11 13:06:12 INFO     kolloqe_components.tokenizers.seetm_tokenizer  -
```

- OVV Handling
- Same Vector (Features)

Specific
Objective

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

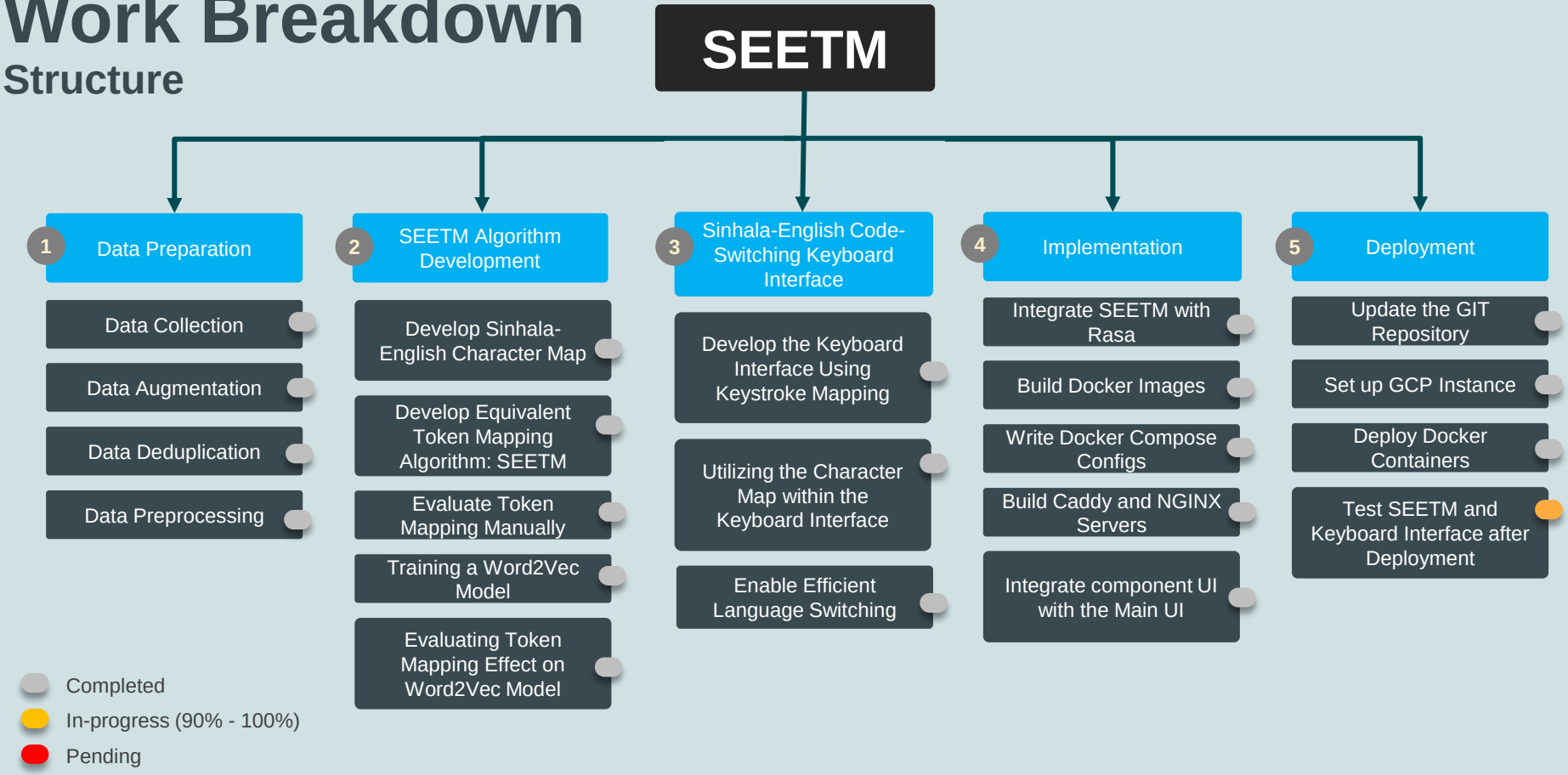
Sub Objective



Overall completion = (objective completion) x (objective weight) = 97.5%

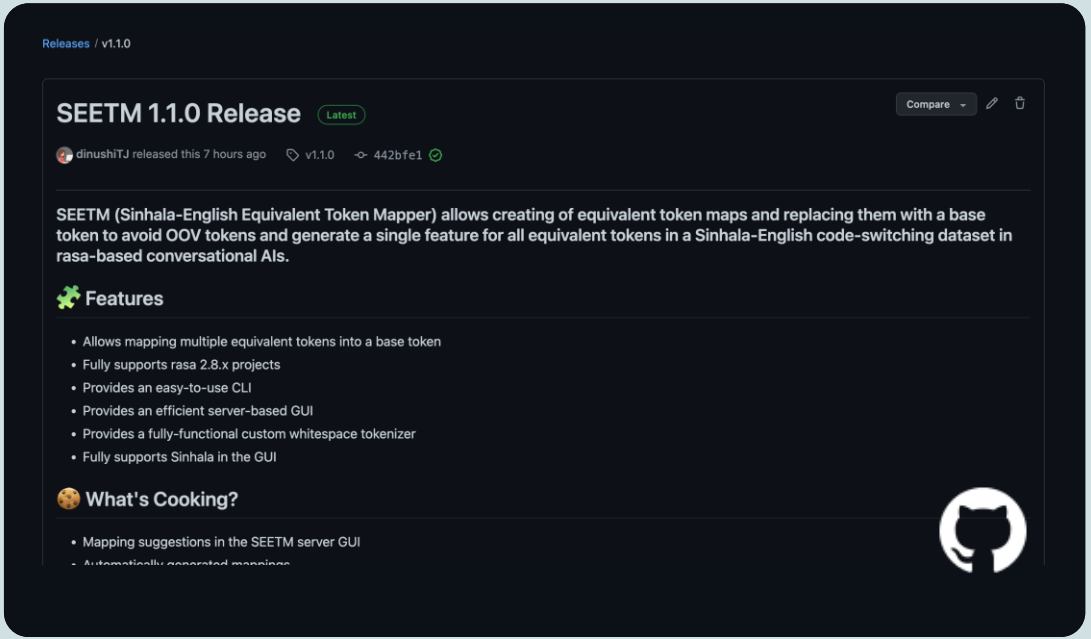
Work Breakdown

Structure

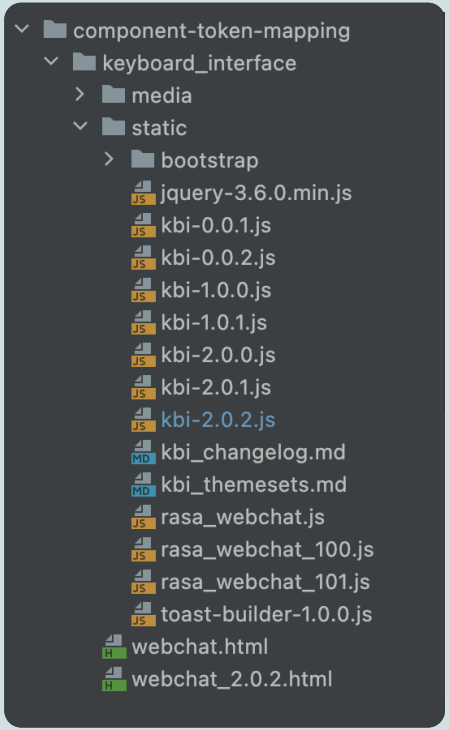


Best Practices

1 Semantic Versioning Releases



2 Implemented Versions

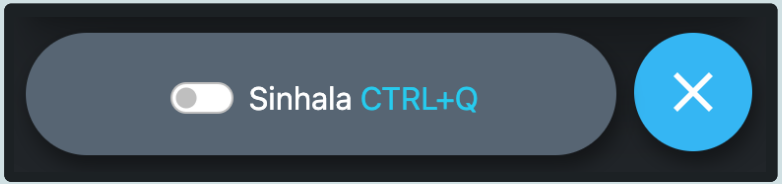


Functional Requirements

- 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
- Usability
- Reliability
- Scalability
- Modularity



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.

[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.

[3]. https://americanipachart.com/

99%

DIME: Dual Interpretable Model-Agnostic Explanations

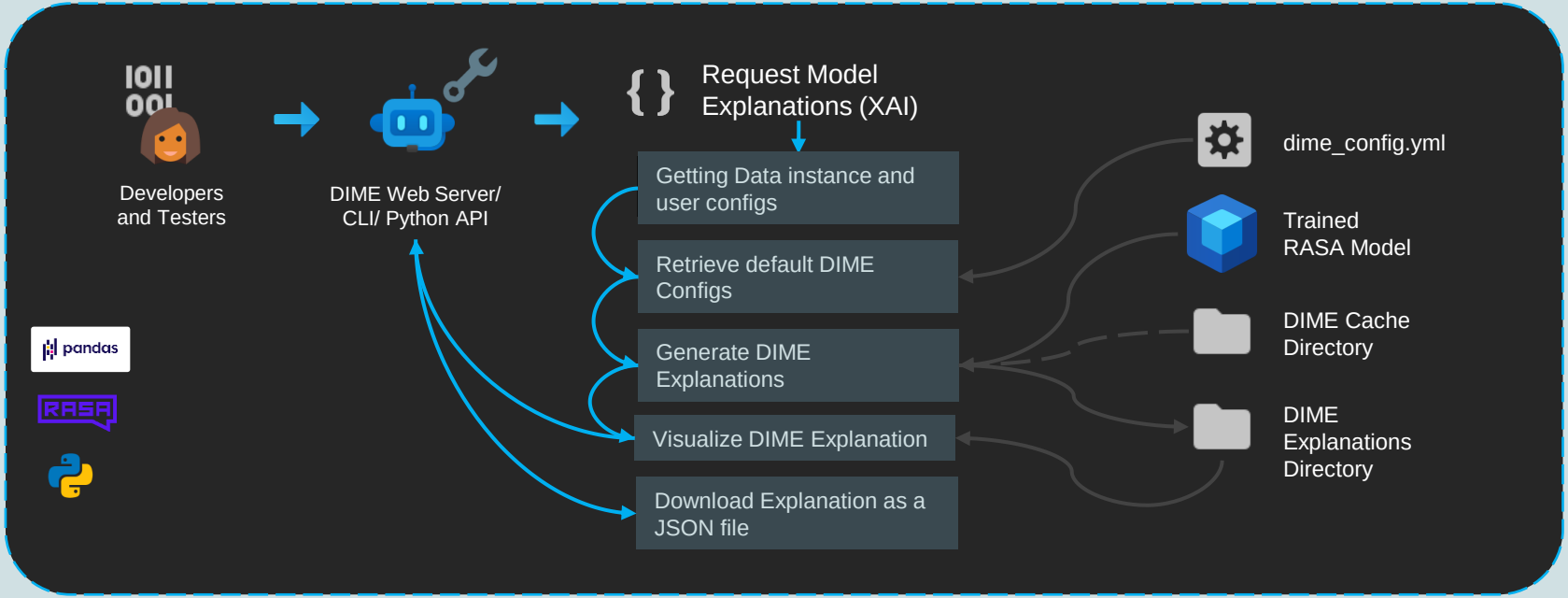
Using global explanations to generate local interpretations in intent classification models using explainable AI

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

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

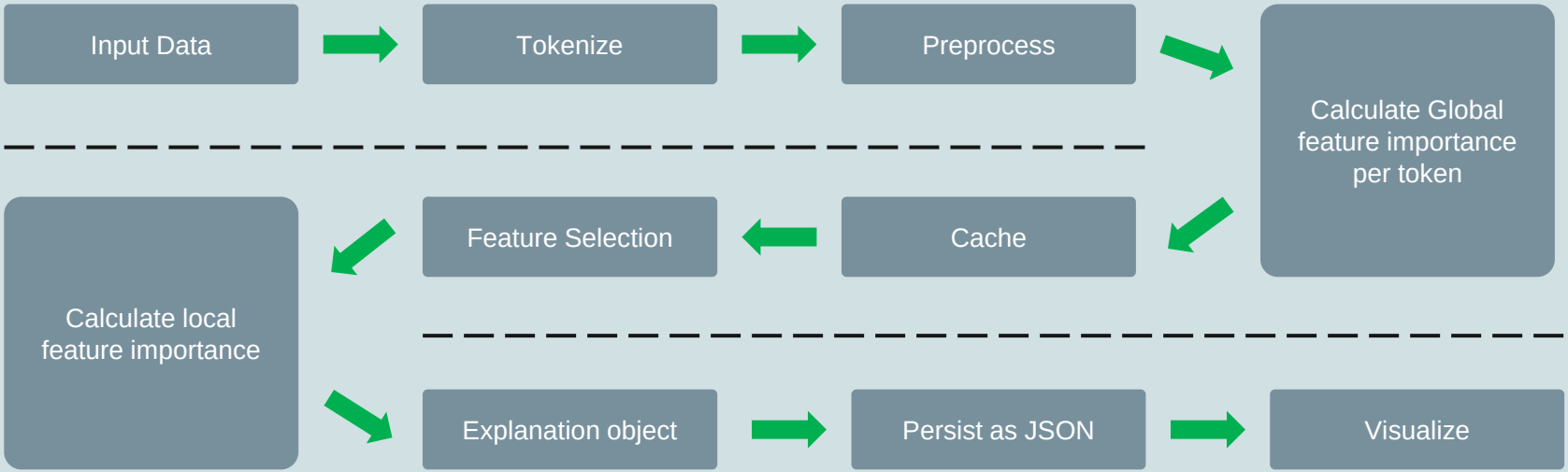
Generating DIME Explanations



Non-functional Requirements

- Efficient Calculations
- Reliable Explanations
- Simple & Interpretable Visualizations
- Modular package

DIME
Algorithm

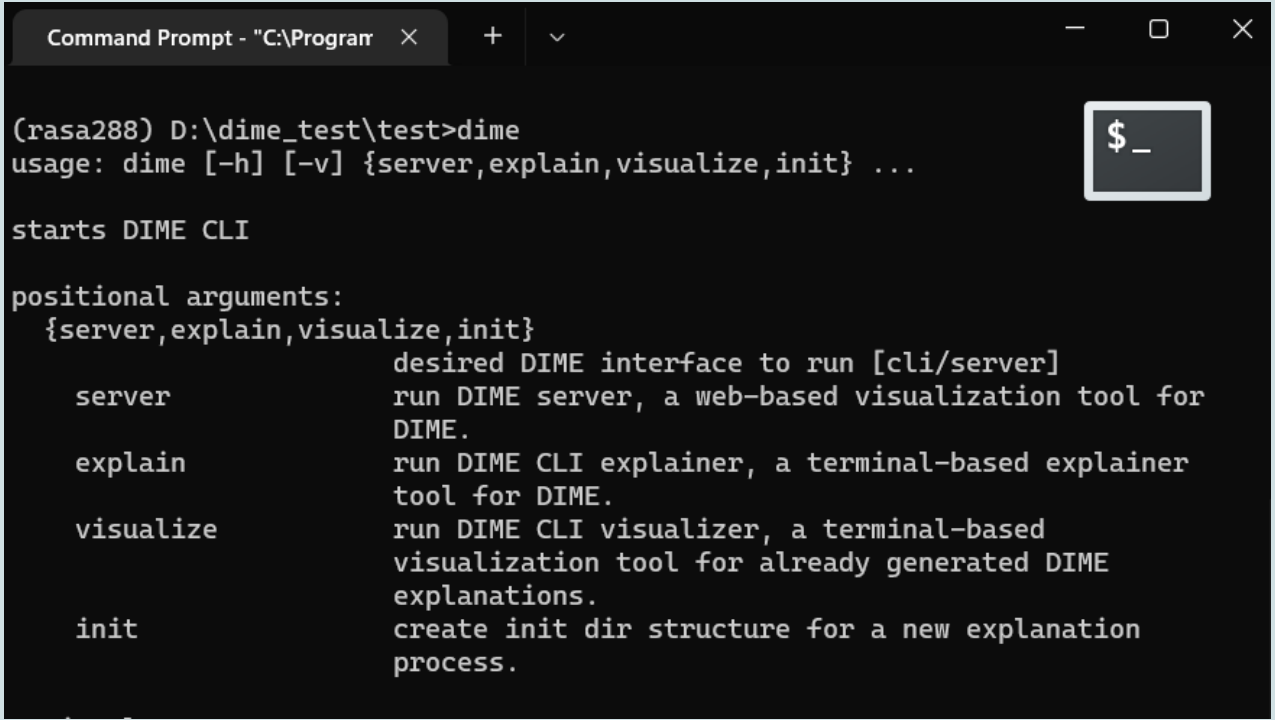


DIME Algorithm cont.

```
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 එක් නියෝග උපාධි මොහොදද?
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):         මොහොදද=4.373067244887352, නියෝග=1.5122178196907043, එක්=0.4100315570831299, උපාධි=0.152559399604797 36, SLIIT=0.141571402549743 65
Selected token list based on global score:      මොහොදද, නියෝග, එක්, උපාධි, SLIIT
Global feature importance scores (Normalized):  මොහොදද=1.0, නියෝග=0.345802553450023 25, එක්=0.09376292065083305, උපාධි=0.034886131646650956, SLIIT=0.03237347944174832
Global feature importance probability scores:   මොහොදද=0.6636470349671681, නියෝග=0.2294908392811836, එක්=0.06222548427978721, උපාධි=0.0231520778287742, SLIIT=0.021484563643086846


මොහොදද: ████████████████████████████████████████ 0.6636470349671681
නියෝග: ██████████████ 0.2294908392811836
එක් : ████ 0.06222548427978721
උපාධි : ▌ 0.0231520778287742
SLIIT: ▌ 0.021484563643086846



Dual feature importance scores (Raw):           SLIIT=0.006376385688781738, උපාධි=0.004443705081939697, මොහොදද=0.004433035850524902, එක්=0.0008720159530639648, නියෝග=-0.002359926700592041
Dual feature importance scores (Normalized):    SLIIT=1.0, උපාධි=0.69690029725738, මොහොදද=0.6952270560302118, එක්=0.13675709024285368, නියෝග=0.0
Dual feature importance probability scores:     SLIIT=0.3954312750660728, උපාධි=0.27557617313841093, මොහොදද=0.2749145212264587, එක්=0.054078030569057606, නියෝග=0.0


SLIIT: ████████████████████████████████████████ 0.3954312750660728
උපාධි : ███████████████████████████ 0.27557617313841093
මොහොදද: ███████████████████████████ 0.2749145212264587
එක් : ████ 0.054078030569057606
නියෝග: 0.0
```

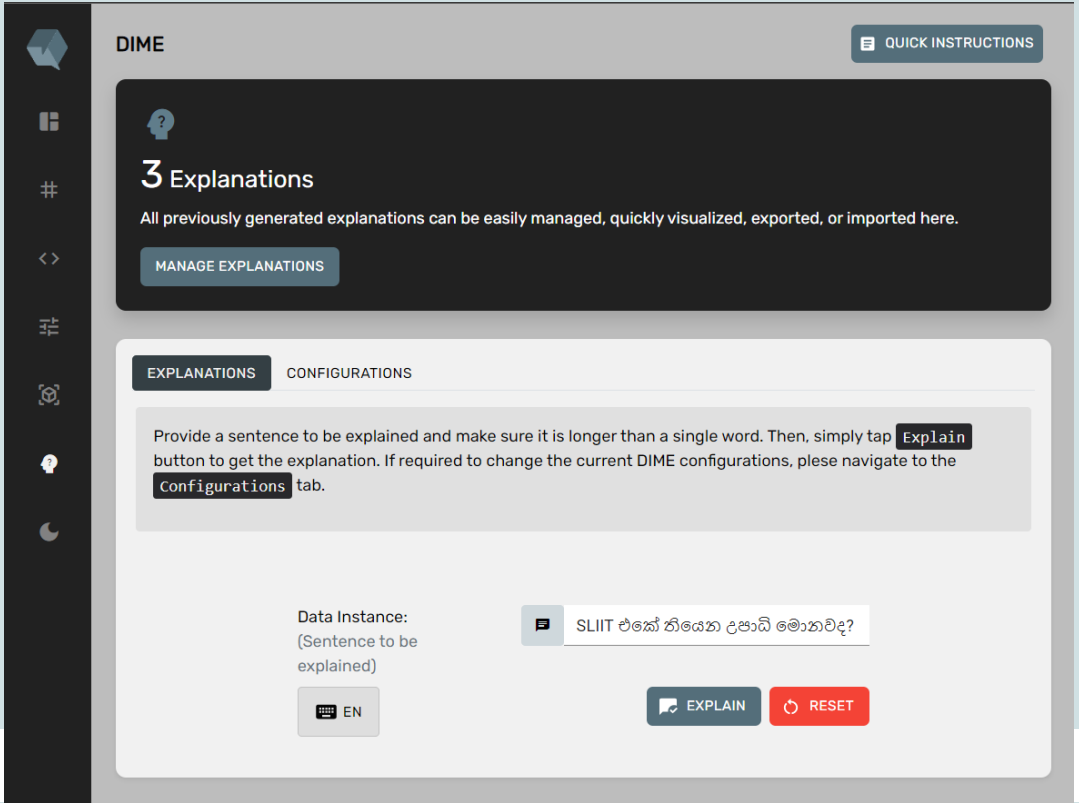
DIME Interfaces

CLI Interface
(terminal)



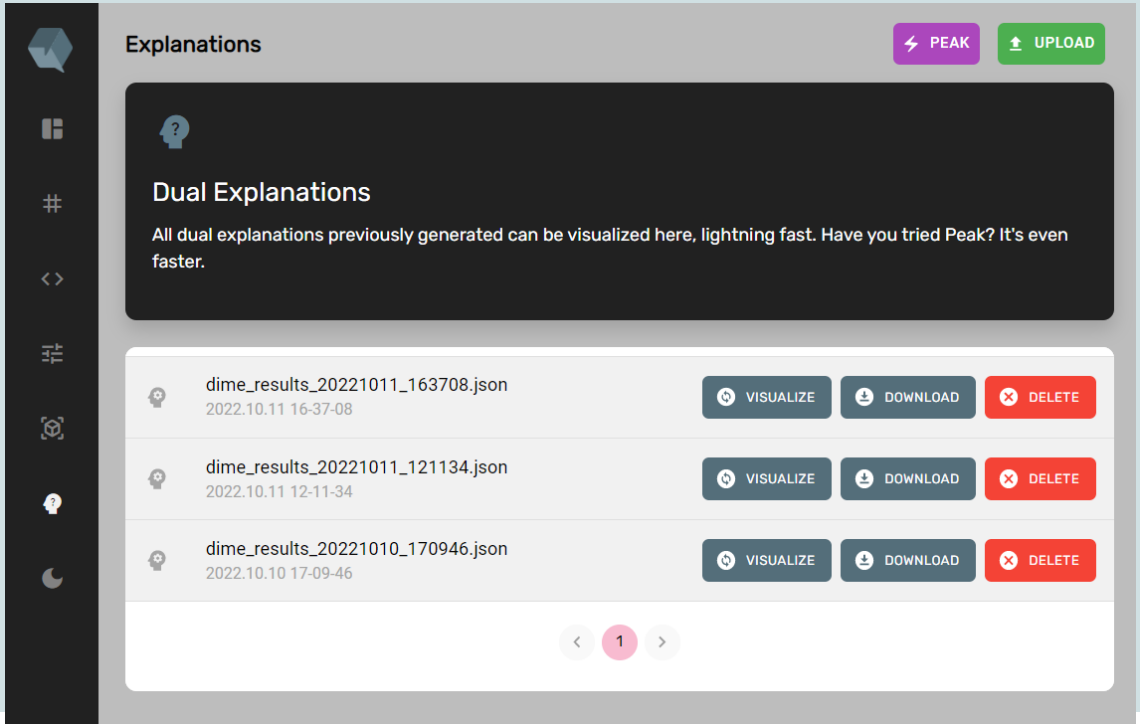
DIME Interfaces

GUI Interface (web)



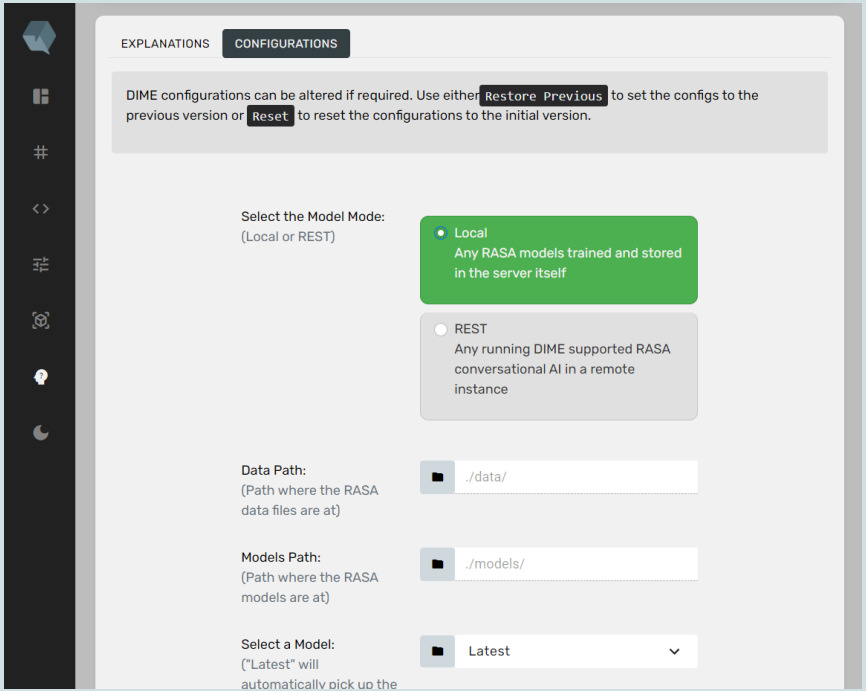
DIME Interfaces

GUI Interface (web)

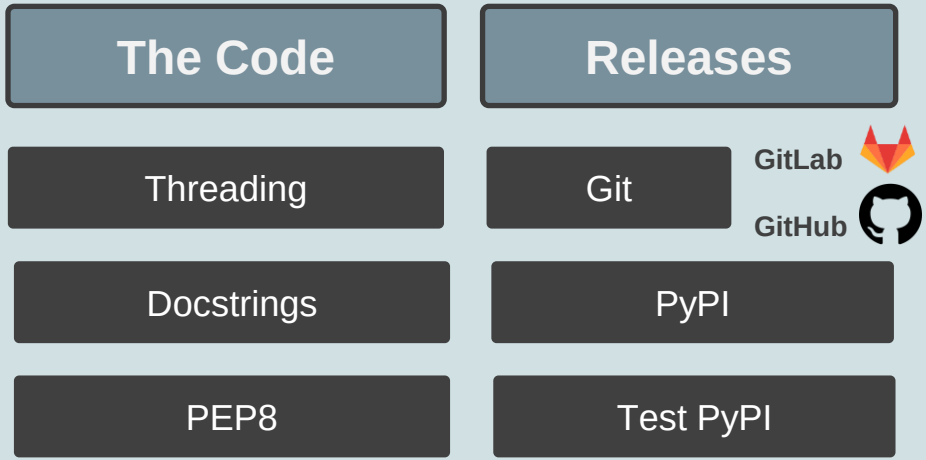


DIME Interfaces

GUI Interface (web)



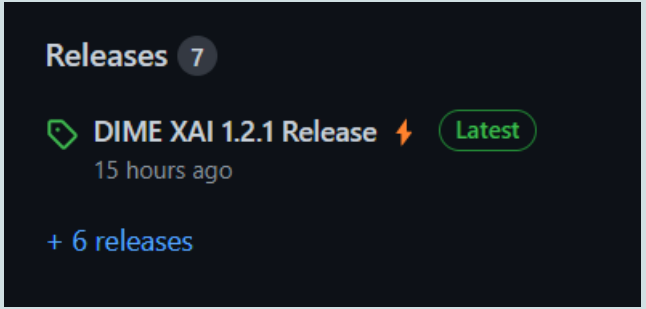
Best Practices



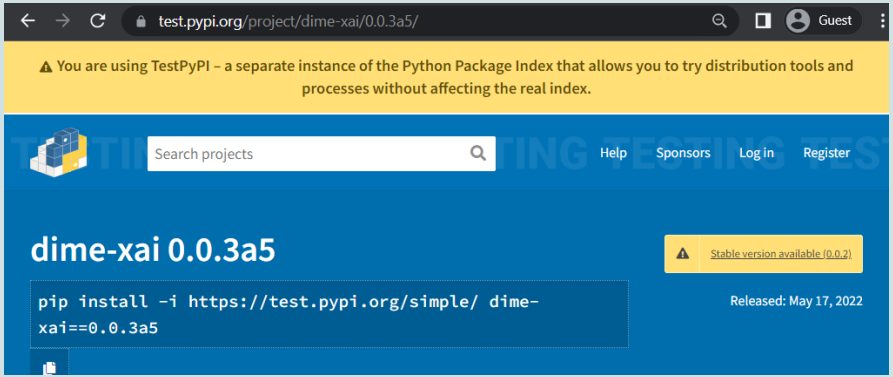
Docstrings & PEP8

```python
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 stri
        merge: if True, mergers a list of strings passed as

    Returns:
        for a single string, returns the list of tokens.
            for a list of unmerged strings, returns the lis
            of tokens per each string instance as a diction
    """
    instances_copy = instances
    if not instances_copy:
        return None

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

Versioning



Test PyPI alpha version release



Vulnerability and Security Testing with **Snyk**

Open **[Snyk] Fix for 7 vulnerabilities** #19

thisisishara wants to merge 1 commit into `main` from `snyk-fix-c86b1a6f8c20a061…`

Vulnerabilities that will be fixed

By pinning:

Severity	Priority Score (*)	Issue	Upgrade	Breaking Change	Exploit Maturity
H	**686/1000** **Why?** Proof of Concept exploit, Has a fix available, CVSS 7.3	HTTP Header Injection SNYK-PYTHON-AIOHTTP-1584144	`aiohttp: 3.7.4 -> 3.8.0`	No	Proof of Concept
H	**758/1000** **Why?** Proof of Concept exploit, Recently disclosed, Has a fix available, CVSS 7.3	Arbitrary Code Execution SNYK-PYTHON-JOBLIB-3027033	`joblib: 1.0.1 -> 1.2.0`	No	Proof of Concept
H	**654/1000** **Why?** Has a fix available, CVSS 8.8	Deserialization of Untrusted Data SNYK-PYTHON-NETWORKX-1062709	`networkx: 2.5.1 -> 2.6`	No	No Known Exploit
L	**506/1000** **Why?** Proof of Concept exploit, Has a fix available, CVSS 3.7	NULL Pointer Dereference SNYK-PYTHON-NUMPY-2321964	`numpy: 1.18.5 -> 1.22.2`	No	Proof of Concept

None yet

Milestone

No milestone

Development

Successfully merging th… issues.

None yet

Notifications

You're receiving notifica… thread.

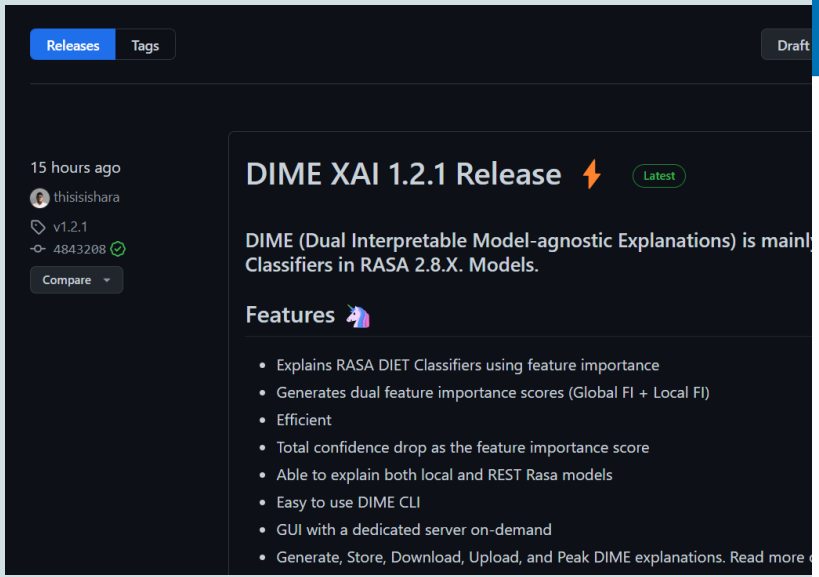
2 participants

Lock conversation

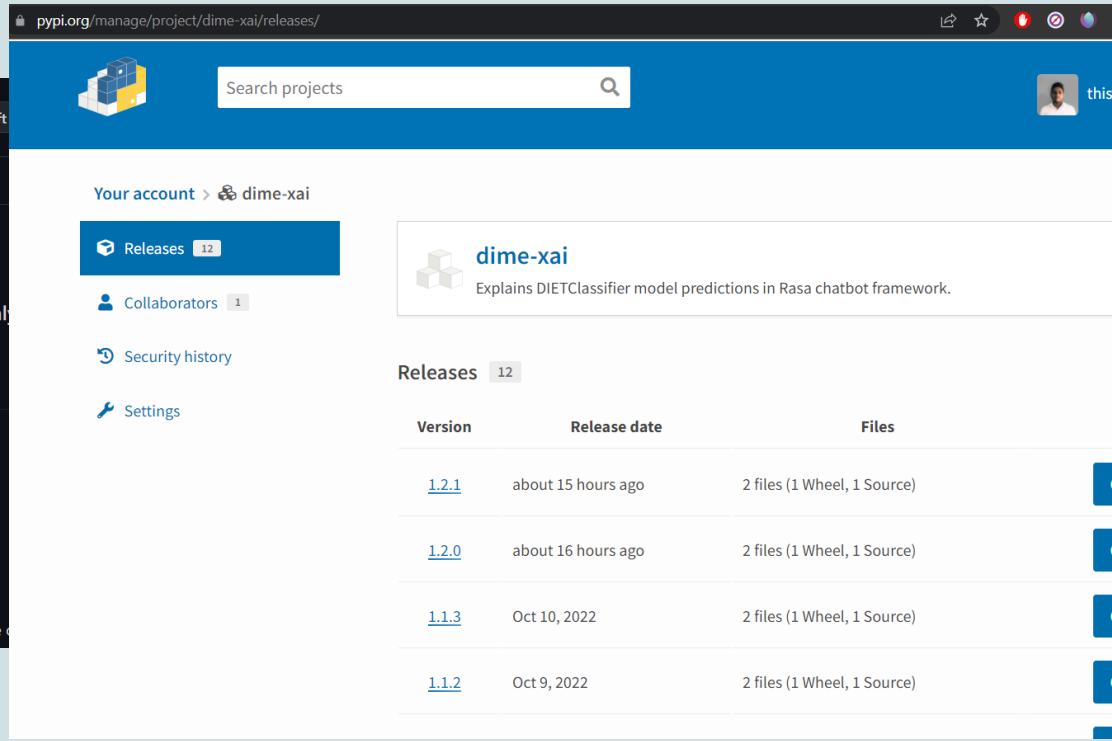
https://pypi.org/manage/project/dime-xai/releases/

DIME Python Package

- Open-source
- Rasa Compatible



https://github.com/DIME-XAI/dime-xai



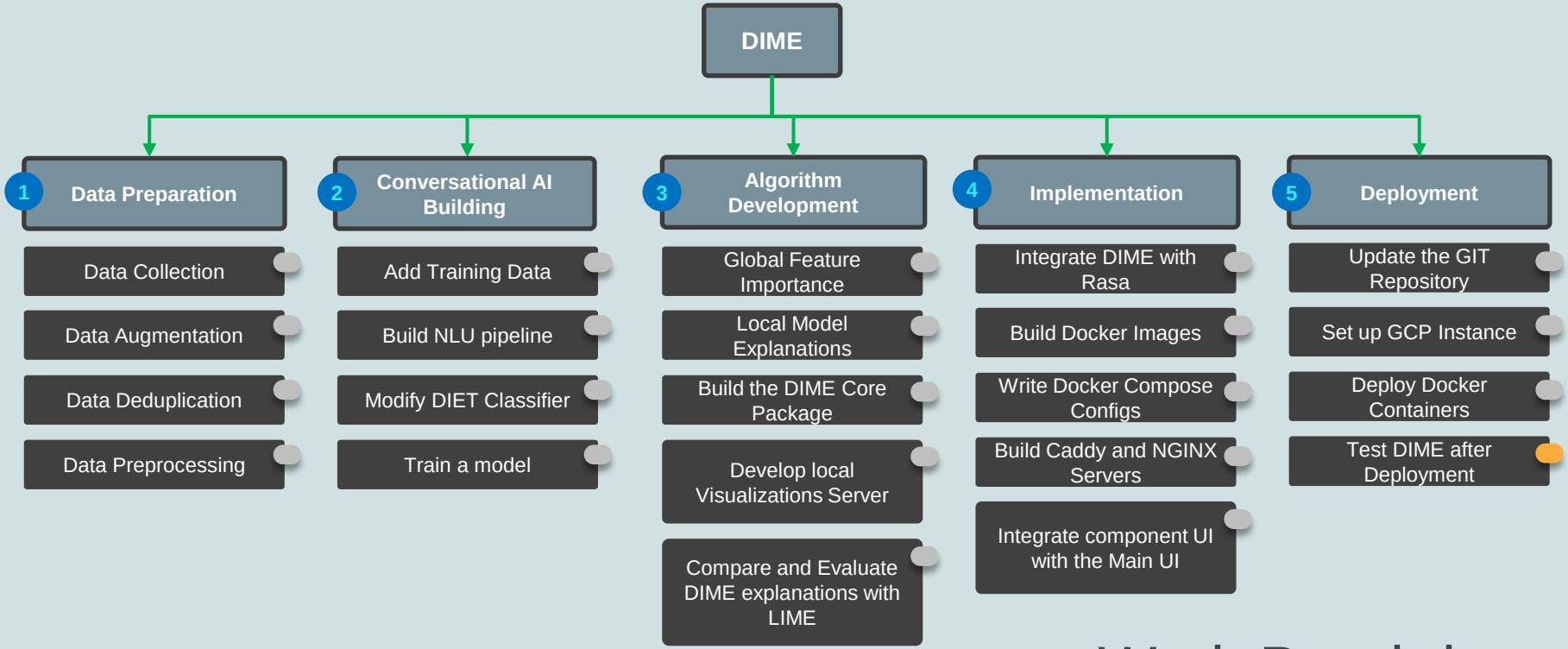
Specific Objective

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

Sub Objectives

- Find Global and Local Explanations logically — 100% * 0.4
- Develop a python package for DIME — 95% * 0.15
- Modify DIET intent classifier to get all confidence scores — 100% * 0.05
- Visualize explanations in an interpretable manner — 100% * 0.3
- Integrate DIME with Rasa seamlessly — 100% * 0.1

Overall completion = (objective completion) x (objective weight) = 99.25%



Completed

In-progress (90% - 100%)

Pending

Work Breakdown Structure

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/1705.078744

[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

Business Model

DEMO PACKAGE

FREE for one month

10 Question Categories
Entity Annotation
Model Evaluations
Model Explainability

ON PREM PACKAGE

$ 299.99 one time

1 Free Maintenance
($50.00 per additional call)
∞ Question Categories
All ML Tools

CaaS PACKAGE

STARTER

$ 9.99/Month

20 Question Categories
Entity Annotation
Limited Model Evaluations
No Model Explainability

PRO

$ 34.99/Month

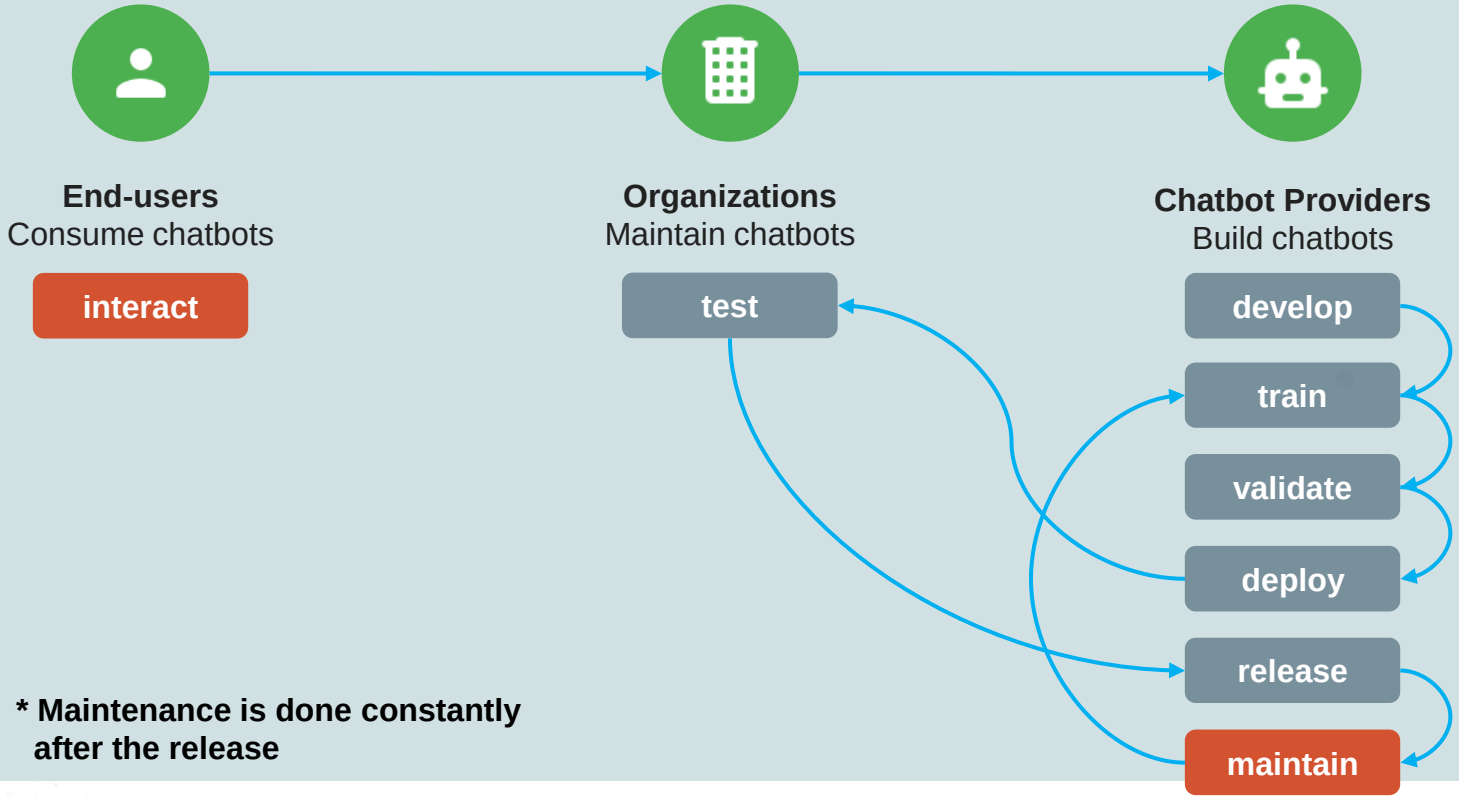
50 Question Categories
Entity Annotation
Model Evaluations
Limited Model Explainability

GENIUS

$ 49.99/Month

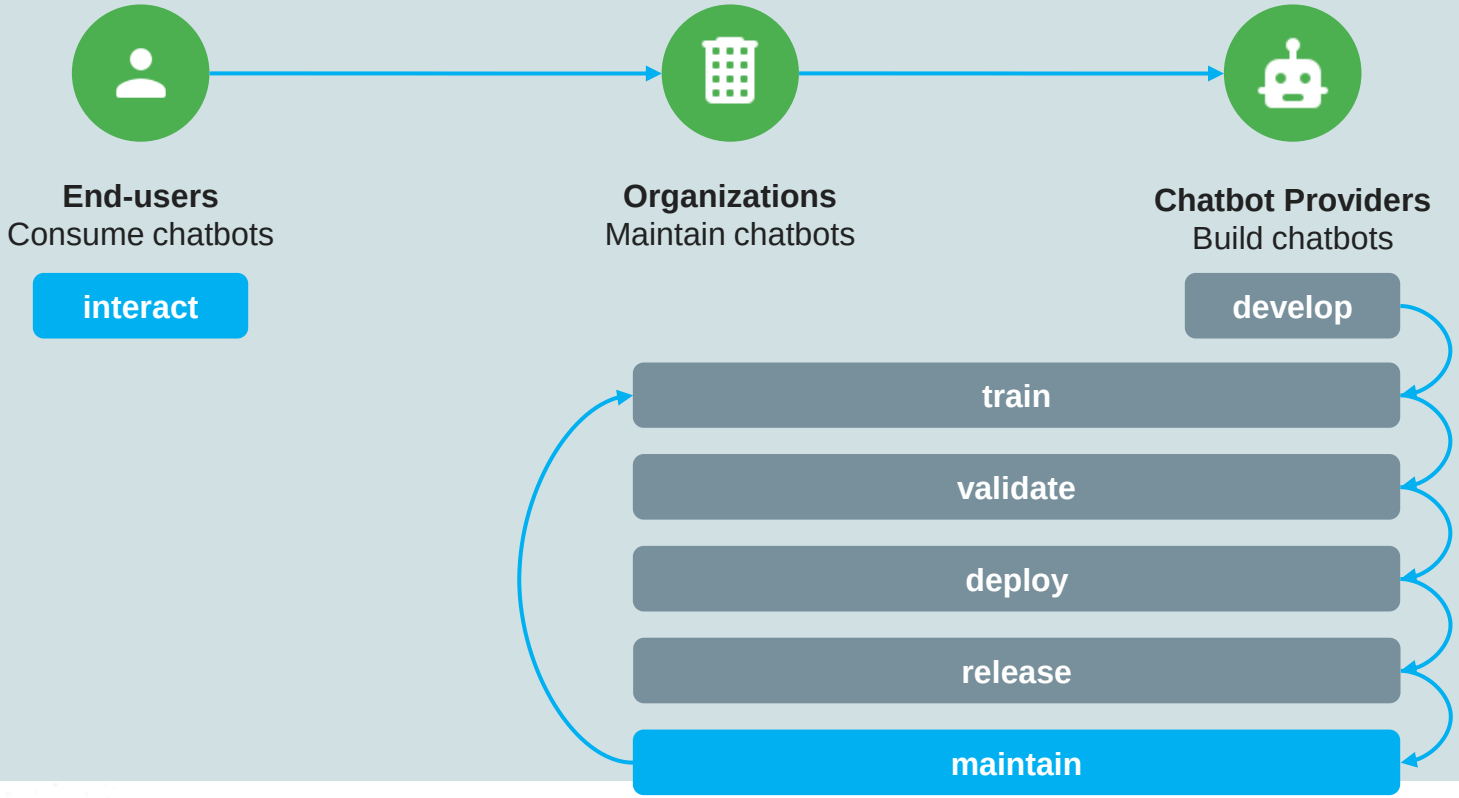
100 Question Categories *
Entity Annotation
Model Evaluations
Model Explainability

Chatbot development lifecycle



*** Maintenance is done constantly after the release**

Chatbot development lifecycle



Thank You

{ }

NLP Tools for Sinhalese

</>

Simple No-Code Development Console

Model Evaluations & Explainability

Any Questions

Demonstration