

TP AGENTS CONVERSATIONNELS : TP rasa (version 2.8)

Tutoriel disponible en ligne

Accéder à la page suivante : <https://rasa.com/docs/rasa/2.x/> (<https://rasa.com/docs/rasa/2.x/>) A travers cette page, procéder aux différentes étapes de développement d'un chatbot. Ce notebook sert de compte rendu et devra être déposé à l'issue des séances. Travail individuel ou en binôme.

NOM ETUDIANT : Manon Augé Couffin

ENVIRONNEMENT DE TRAVAIL MFJA ou PC PERSO

TP sur les machines de la MFJA

Si vous êtes sur les PC de la MFJA, connectez-vous sous Windows (seul choix possible) 1- Lancer la machine virtuelle (VMware) nommée Ubuntu 64-bit_SRI_PAROLE (dédiée aux TP de parole 2A et 3A SRI), sous C:\VWware\Ubuntu 64-bit_SRI_PAROLE_2023

- Cliquer sur la machine = icône 3 carrés imbriqués
 - Ne pas saisir de licence --> Continue
 - Sélectionner I copied It et Skip this version
- La machine est lancée avec le login sriparole
- Le mot de passe vous sera donné en séance

2- Aller dans le dossier dédié home/Documents/SRI3A_TP_RASA

3- Créer un Dossier à votre nom TP_RASA_VOTRE_NOM et vous y déplacer

4- Activer l'environnement de travail -> rasa est installé dans un environnement virtuel conda

5- Si conda est activé par défaut vous avez (base) dans le prompt sinon, depuis le terminal, lancer conda avec \$ source ~/.bashrcCONDA

6- une fois conda lancé vérifier la présence de l'environnement virtuel env_rasa2.8 dédié au TP et l'activer *conda info -e* conda activate env_rasa2.8

--> Vous devez avoir le prompt : (env_rasa2.8) sriparole@sriparole-virtual-machine:~Documents/SRI3A_TP_RASA

7- A la fin du TP quitter l'environnement avec : \$ conda deactivate

Pour retrouver votre travail rester sur la même machine en salle de TP, sinon, en fin de séance faire une archive de votre répertoire de TP et déposez-là sur moodle ou sur votre drive.

TP sur votre machine

Si vous avez installé rasa v2.8 sur votre PC (en suivant les directives sur moodle) et en ayant créé un environnement virtuel pour faire cette installation (avec conda de préférence), vous devez exécuter la commande depuis le terminal et sous le répertoire dédié au TP_RASA_VOTRE_NOM \$ rasa init' à vous de voir où vous l'avez mis MAIS on ne travaille pas sous Téléchargement !!!

Pour installer conda voir : <https://docs.conda.io/projects/conda/en/latest/commands/install.html> (<https://docs.conda.io/projects/conda/en/latest/commands/install.html>)

Positionnez-vous dans votre répertoire TP_RASA_VOTRE_NOM

Dans les deux cas, vous êtes dans le répertoire TP_RASA_VOTRE_NOM. Créer une copie de CHATBOT_V0 situé dans le répertoire père et nommer la copie CHATBOT_V1_VOTRE_NOM. Vous déplacer dans ce dossier.

- Ce sera votre version de travail
- Ceci afin de ne pas modifier la version CHATBOT_V0 initiale

Vous avez alors l'organisation suivante :

```
.
├── actions
│   ├── __init__.py
│   └── actions.py
├── config.yml
├── credentials.yml
├── data
│   ├── nlu.yml
│   └── stories.yml
├── domain.yml
├── endpoints.yml
├── models
│   └── <timestamp>.tar.gz
├── tests
│   └── test_stories.yml
```

Votre chatbot est dans votre répertoire TP_RASA_VOTRE_NOM/CHATBOT_V1_VOTRE_NOM. Il faudra faire une archive du chatbot développé par la suite.

PARTIE 1 : ETUDE DU CHATBOT DE BASE

1- Partie Compréhension

INTENTS et ENTITIES (Intentions et Entités) - rasa V2

Observer le contenu du fichier "data/nlu.yml"

```
version: "2.0"

nlu:
- intent: greet
  examples: |
    - hey
    - hello
    - hi
    - hello there
    - good morning
    - good evening
    - moin
    - hey there
    - let's go
    - hey dude
    - goodmorning
    - goodevening
    - good afternoon

- intent: goodbye
  examples: |
    - good afternoon
    - cu
    - good by
    - cee you later
    - good night
    - bye
    - goodbye
    - have a nice day
    - see you around
    - bye bye
    - see you later

- intent: affirm
  examples: |
    - yes
    - y
    - indeed
    - of course
    - that sounds good
    - correct

- intent: deny
  examples: |
    - no
    - n
    - never
    - I don't think so
    - don't like that
    - no way
    - not really

- intent: mood_great
  examples: |
    - perfect
    - great
    - amazing
    - feeling like a king
    - wonderful
    - I am feeling very good
```

```

- I am great
- I am amazing
- I am going to save the world
- super stoked
- extremely good
- so so perfect
- so good
- so perfect

- intent: mood_unhappy
  examples: |
    - my day was horrible
    - I am sad
    - I don't feel very well
    - I am disappointed
    - super sad
    - I'm so sad
    - sad
    - very sad
    - unhappy
    - not good
    - not very good
    - extremly sad
    - so saad
    - so sad

- intent: bot_challenge
  examples: |
    - are you a bot?
    - are you a human?
    - am I talking to a bot?
    - am I talking to a human?

```

1) Comment interprétez-vous le contenu de ce fichier (= commentez en quelques lignes) ?

Le contenu de ce fichier est des intents avec des exemples pour chacun d'eux. Cela permet au chatbot d'avoir une ligne de depart pour comprendre l'utilisateur.

2) Combien d'intents et lesquelles ?

Il y a sept intents : greet, goodbye, affirm, deny, mood_great, mood_unhappy et bot_challenge.

3) Combien d'entités et lesquelles ?

Il y en a aucune.

2- Configuration

PIPELINE et POLICIES

Observer le contenu du fichier "config.yml" - Celles-ci peuvent ne pas être commentées suivant les versions - commentaire # pipeine par défaut

```
# Configuration for Rasa NLU.
# https://rasa.com/docs/rasa/nlu/components/
language: en

pipeline:
# # No configuration for the NLU pipeline was provided. The
# following default pipeline was used to train your model.
# # If you'd like to customize it, uncomment and adjust the
# pipeline.
# # See https://rasa.com/docs/rasa/tuning-your-model for more
# information.
#   - name: WhitespaceTokenizer
#   - name: RegexFeaturizer
#   - name: LexicalSyntacticFeaturizer
#   - name: CountVectorsFeaturizer
#   - name: CountVectorsFeaturizer
#     analyzer: char_wb
#     min_ngram: 1
#     max_ngram: 4
#   - name: DIETClassifier
#     epochs: 100
#   - name: EntitySynonymMapper
#   - name: ResponseSelector
#     epochs: 100
#   - name: FallbackClassifier
#     threshold: 0.3
#     ambiguity_threshold: 0.1

# Configuration for Rasa Core.
# https://rasa.com/docs/rasa/core/policies/
policies:
# # No configuration for policies was provided. The following
# default policies were used to train your model.
# # If you'd like to customize them, uncomment and adjust the
# policies.
# # See https://rasa.com/docs/rasa/policies for more information.
#   - name: MemoizationPolicy
#   - name: TEDPolicy
#     max_history: 5
#     epochs: 100
#   - name: RulePolicy
```

1) Comment interprétez-vous le contenu de ce fichier ?

2) Chacune des 3 parties ?

3- Gestion de dialogue

3.1- STORIES

Observer le contenu du fichier "data/stories.yml"

```
version: "2.0"
```

```
stories:
```

```
- story: happy path
  steps:
  - intent: greet
  - action: utter_greet
  - intent: mood_great
  - action: utter_happy

- story: sad path 1
  steps:
  - intent: greet
  - action: utter_greet
  - intent: mood_unhappy
  - action: utter_cheer_up
  - action: utter_did_that_help
  - intent: affirm
  - action: utter_happy

- story: sad path 2
  steps:
  - intent: greet
  - action: utter_greet
  - intent: mood_unhappy
  - action: utter_cheer_up
  - action: utter_did_that_help
  - intent: deny
  - action: utter_goodbye
```

1) Comment interprétez-vous le contenu de ce fichier ?

Ce sont des chemins à suivre sur l'ordre d'utilisation des intents.

2) Combien de stories ?

Il existe trois stories : happy path, sad path 1 et sad path 2.

3) Quel rôle ?

Le premier a pour but d'avoir une discussion avec quelqu'un de joyeux/heureux. Les deux autres sont dans une mauvaise phase (mauvaise humeur. Leur différence est que le chatbot fini sur une note positive sur le premier. Cela n'est pas le cas pour le dernier.

3.2- RULES

Observer le contenu du fichier "data/rules.yml"

```
version: "2.0"
```

```
rules:
```

```
- rule: Say goodbye anytime the user says goodbye
  steps:
  - intent: goodbye
  - action: utter_goodbye

- rule: Say 'I am a bot' anytime the user challenges
```

```
steps:
- intent: bot_challenge
- action: utter_iamabot
```

1) Comment interprétez-vous le contenu de ce fichier ?

Ce sont des règles pour faire répondre le chatbot dans certaines situations bien particulières.

2) Combien de règles ?

Il y a deux règles : goodbye et challenge.

3) Quel rôle ?

Goodbye a pour but de dire au revoir en si l'utilisateur le dit. Challenge a pour but de dire que c'est un bot quand l'utilisateur le challenge.

4- Domaine

Consultez le fichier décrivant l'environnement du chatbot

--

Observer le contenu du fichier "domain.yml" peut être légèrement différent suivant les versions.

```
session_config:
  session_expiration_time: 60
  carry_over_slots_to_new_session: true

intents:
- greet
- goodbye
- affirm
- deny
- mood_great
- mood_unhappy
- bot_challenge

responses:
  utter_greet:
    - text: Hey! How are you?
  utter_cheer_up:
    - image: https://i.imgur.com/nGF1K8f.jpg
      text: 'Here is something to cheer you up:'
  utter_did_that_help:
    - text: Did that help you?
  utter_happy:
    - text: Great, carry on!
  utter_goodbye:
    - text: Bye
  utter_iamabot:
    - text: I am a bot, powered by Rasa.
```

--

Comment interprétez-vous le contenu de ce fichier ? Chacune des parties ?

C'est un fichier qui liste tous les intent et les actions.

PARTIE 2 : APPRENTISSAGE ET EVALUATION DU CHATBOT

RASA a été installé en local dans l'environnement virtuel env_rasa2.8 construit avec conda.

Il faut que c'est environnement ait été activé. -> voir détail dans la section ENVIRONNEMENT en début de sujet.

5- Apprentissage

ATTENTION A BIEN ETRE DANS VOTRE REPERTOIRE CHATBOT_V1_VOTRE_NOM

5.1- Consultez le répertoire ./models de votre Chatbot

Si celui-ci contient un fichier .tar.gz supprimez ou renommez ce fichier

5.2- Depuis le répertoire CHATBOT_V1_VOTRE_NOM exécutez la commande : rasa train

```
-----  
-----  
Training NLU model...  
2023-12-06 14:51:27 INFO  
rasa.shared.nlu.training_data.training_data - Training data stats:  
2023-12-06 14:51:27 INFO  
rasa.shared.nlu.training_data.training_data - Number of intent  
examples: 69 (7 distinct intents)  
  
2023-12-06 14:51:27 INFO  
rasa.shared.nlu.training_data.training_data - Found intents:  
'bot_challenge', 'deny', 'mood_unhappy', 'goodbye', 'affirm',  
'mood_great', 'greet'  
2023-12-06 14:51:27 INFO  
rasa.shared.nlu.training_data.training_data - Number of response  
examples: 0 (0 distinct responses)  
2023-12-06 14:51:27 INFO  
rasa.shared.nlu.training_data.training_data - Number of entity  
examples: 0 (0 distinct entities)  
2023-12-06 14:51:27 INFO      rasa.nlu.model - Starting to train  
component WhitespaceTokenizer  
2023-12-06 14:51:27 INFO      rasa.nlu.model - Finished training  
component.  
2023-12-06 14:51:27 INFO      rasa.nlu.model - Starting to train  
component RegexFeaturizer  
2023-12-06 14:51:27 INFO      rasa.nlu.model - Finished training  
component.  
2023-12-06 14:51:27 INFO      rasa.nlu.model - Starting to train  
component LexicalSyntacticFeaturizer  
2023-12-06 14:51:27 INFO      rasa.nlu.model - Finished training  
component.
```



```

2023-12-06 14:51:27 INFO      rasa.nlu.model - Starting to train
component CountVectorsFeaturizer
2023-12-06 14:51:27 INFO
rasa.nlu.featurizers.sparse_featurizer.count_vectors_featurizer -
80 vocabulary items were created for text attribute.
2023-12-06 14:51:27 INFO      rasa.nlu.model - Finished training
component.
2023-12-06 14:51:27 INFO      rasa.nlu.model - Starting to train
component CountVectorsFeaturizer
2023-12-06 14:51:27 INFO
rasa.nlu.featurizers.sparse_featurizer.count_vectors_featurizer -
697 vocabulary items were created for text attribute.
2023-12-06 14:51:27 INFO      rasa.nlu.model - Finished training
component.
2023-12-06 14:51:27 INFO      rasa.nlu.model - Starting to train
component DIETClassifier
Epochs: 100%|██████████| 100/100 [00:18<00:00, 5.41it/s,
t_loss=1.2, i_acc=0.986]
2023-12-06 14:51:46 INFO      rasa.nlu.model - Finished training
component.
2023-12-06 14:51:46 INFO      rasa.nlu.model - Starting to train
component EntitySynonymMapper
2023-12-06 14:51:46 INFO      rasa.nlu.model - Finished training
component.
2023-12-06 14:51:46 INFO      rasa.nlu.model - Starting to train
component ResponseSelector
2023-12-06 14:51:46 INFO      rasa.nlu.selectors.response_selector
- Retrieval intent parameter was left to its default value. This
response selector will be trained on training examples combining
all retrieval intents.
2023-12-06 14:51:46 INFO      rasa.nlu.model - Finished training
component.
2023-12-06 14:51:46 INFO      rasa.nlu.model - Starting to train
component FallbackClassifier
2023-12-06 14:51:46 INFO      rasa.nlu.model - Finished training
component.
2023-12-06 14:51:46 INFO      rasa.nlu.model - Successfully saved
model into '/tmp/tmpc41l6i5l/nlu'
NLU model training completed.
Training Core model...
2023-12-06 14:51:53 WARNING  rasa.shared.utils.common - The
Unexpected Intent Policy is currently experimental and might change
or be removed in the future 🐞 Please share your feedback on it in
the forum (https://forum.rasa.com) to help us make this feature
ready for production.
Processed story blocks: 100%|██████████| 3/3 [00:00<00:00,
3028.38it/s, # trackers=1]
Processed story blocks: 100%|██████████| 3/3 [00:00<00:00,
711.86it/s, # trackers=3]
Processed story blocks: 100%|██████████| 3/3 [00:00<00:00,
308.64it/s, # trackers=12]
Processed story blocks: 100%|██████████| 3/3 [00:00<00:00,
73.81it/s, # trackers=39]
Processed rules: 100%|██████████| 2/2 [00:00<00:00,
3018.57it/s, # trackers=1]
Processed trackers: 100%|██████████| 3/3 [00:00<00:00,
2072.63it/s, # action=12]
Processed actions: 12it [00:00, 11169.92it/s, # examples=12]
Processed trackers: 100%|██████████| 2/2 [00:00<00:00,
3723.31it/s, # action=5]

```

```

Processed actions: 5it [00:00, 18460.85it/s, # examples=4]
Processed trackers: 100%|████████████████████| 3/3 [00:00<00:00,
2258.65it/s, # action=12]
Processed trackers: 100%|████████████████████| 2/2
[00:00<00:00, 2269.65it/s]
Processed trackers: 100%|████████████████████| 5/5
[00:00<00:00, 2013.97it/s]
Processed trackers: 100%|████████████████████| 120/120 [00:00<00:00,
5169.32it/s, # intent=12]
Epochs: 100%|████████| 100/100 [00:07<00:00, 13.69it/s, t_loss=0.128,
loss=0.00744, acc=1]
Processed trackers: 100%|████████████████████| 120/120 [00:00<00:00,
3161.08it/s, # action=30]
Epochs: 100%|████████| 100/100 [00:08<00:00, 12.01it/s,
t_loss=0.595, loss=0.412, acc=1]
2023-12-06 14:52:11 INFO      rasa.core.agent - Persisted model to
'/tmp/tmpc41l6i5l/core'
Core model training completed.
Your Rasa model is trained and saved at '/home/sriparole/Documents
/TP_RASA_ManonAugeCouffin/CHATBOT_V1/models
/20231206-145213.tar.gz'.
-----

```

1) Comment interprétez-vous cette trace ?

Dans un premier temps il check les infos de la pipeline. Puis il vient tester les stories, les rules et les domaines.

2) Faites le lien avec le fichier config.yml et commentez ce que vous pensez avoir compris.

Dans le fichier config.yml, il y a les pipelines. L'entrainement entraine chacune d'entre elle.

6- EVALUATION

6.1- Données de test

Observer le contenu du fichier ./tests/test_stories.yml

```

##### This file contains tests to evaluate that your bot behaves as
expected.
##### If you want to learn more, please see the docs:
https://rasa.com/docs/rasa/testing-your-assistant

stories:
- story: happy path 1
  steps:
  - user: |
    hello there!
    intent: greet
  - action: utter_greet
  - user: |
    amazing
    intent: mood_great
  - action: utter_happy

```

```
- story: happy path 2
steps:
- user: |
    hello there!
  intent: greet
- action: utter_greet
- user: |
    amazing
  intent: mood_great
- action: utter_happy
- user: |
    bye-bye!
  intent: goodbye
- action: utter_goodbye

- story: sad path 1
steps:
- user: |
    hello
  intent: greet
- action: utter_greet
- user: |
    not good
  intent: mood_unhappy
- action: utter_cheer_up
- action: utter_did_that_help
- user: |
    yes
  intent: affirm
- action: utter_happy

- story: sad path 2
steps:
- user: |
    hello
  intent: greet
- action: utter_greet
- user: |
    not good
  intent: mood_unhappy
- action: utter_cheer_up
- action: utter_did_that_help
- user: |
    not really
  intent: deny
- action: utter_goodbye

- story: sad path 3
steps:
- user: |
    hi
  intent: greet
- action: utter_greet
- user: |
    very terrible
  intent: mood_unhappy
- action: utter_cheer_up
- action: utter_did_that_help
- user: |
    no
```

```

        intent: deny
    - action: utter_goodbye

- story: say goodbye
  steps:
  - user: |
      bye-bye!
      intent: goodbye
  - action: utter_goodbye

- story: bot challenge
  steps:
  - user: |
      are you a bot?
      intent: bot_challenge
  - action: utter_iamabot

```

1) Comment interprétez-vous le contenu de ce fichier ?

C'est une serie de tests sur les sories.

2) Combien y-at-il de story de test ?

Il y a sept stories : happy path 1, happy path 2, sad path 1, sad path 2, sad path 3, say goodbye et bot challenge.

3) Combien y-a-t'il de tour de parole utilisateur ? combien y-a-t'il d'actions ?

Pour happy path 1 : il y a 2 tours de parole utilisateur et 2 actions.

Pour happy path 2 : il y a 3 tours de parole utilisateur et 3 actions.

Pour sad path 1 : il y a 3 tours de parole utilisateur et 4 actions.

Pour sad path 2 : il y a 2 tours de parole utilisateur et 4 actions.

Pour sad path 3 : il y a 3 tours de parole utilisateur et 4 actions.

Pour say goodbye : il y a 1 tour de parole utilisateur et 1 action.

Pour bot challenge : il y a 1 tour de parole utilisateur et 1 action.

6.2- Résultats de l'évaluation

Depuis le répertoire CHATBOT_V1_VOTRE_NOM exécutez la commande : rasa test

```

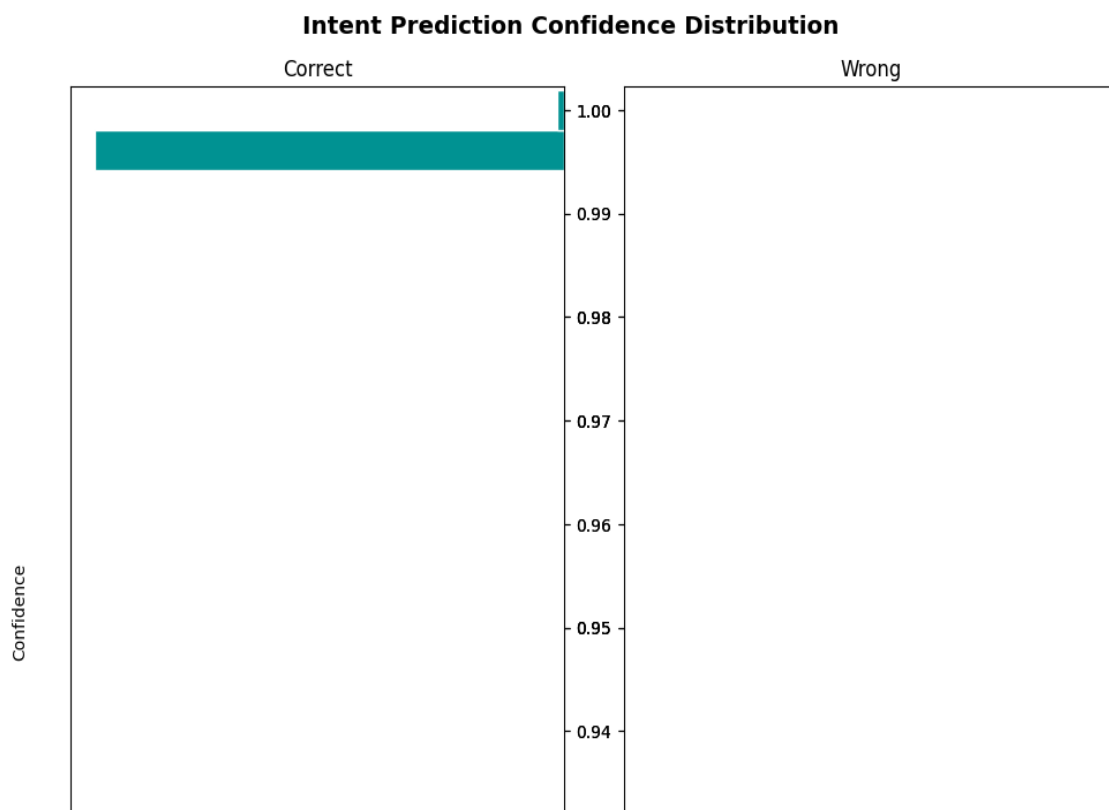
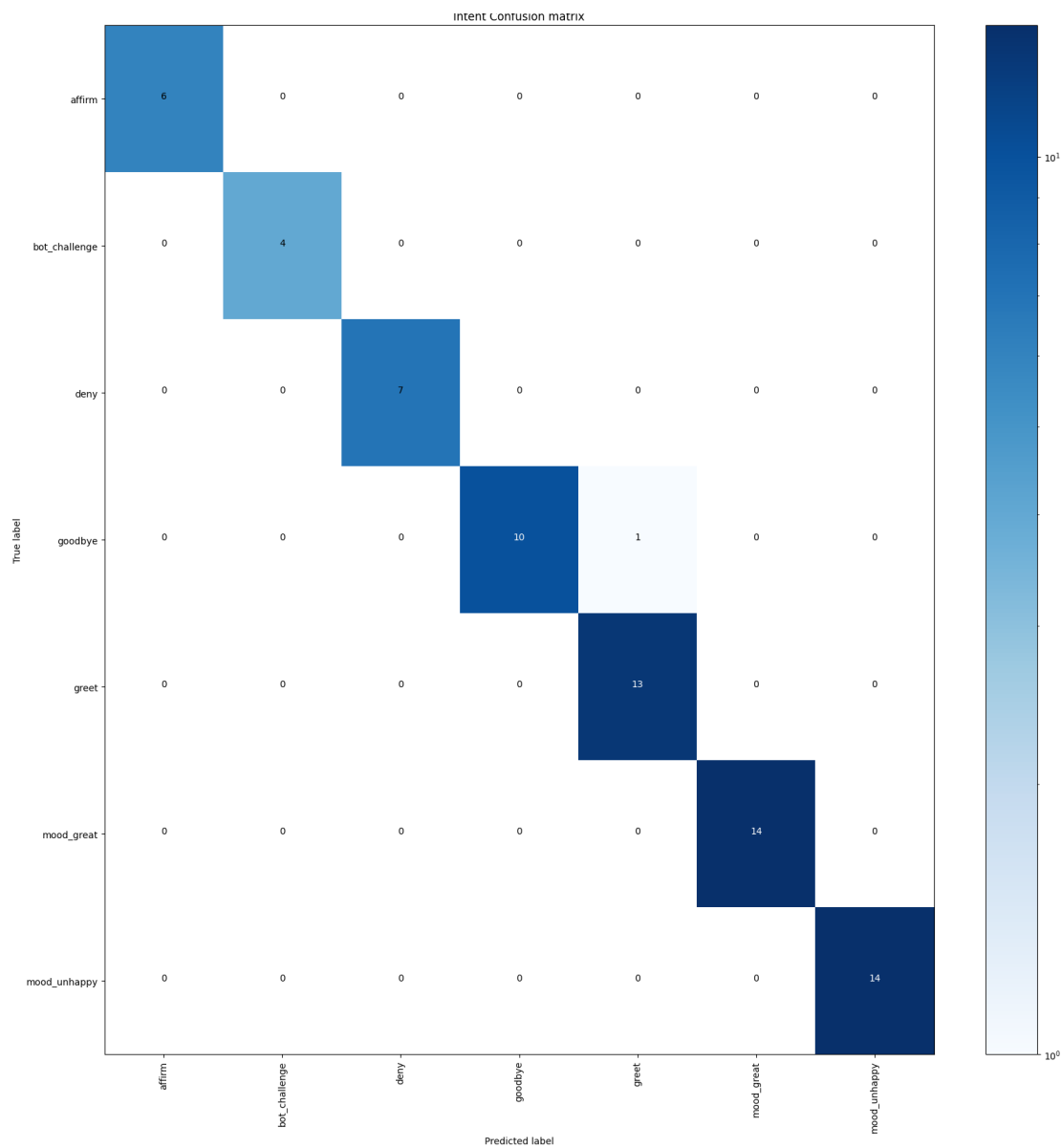
-----
-----
2023-12-06 15:16:45 INFO      rasa.model - Loading model
models/20231206-145213.tar.gz...
2023-12-06 15:16:58 WARNING  rasa.shared.utils.common - The
Unexpected Intent Policy is currently experimental and might change
or be removed in the future 🐛 Please share your feedback on it in
the forum (https://forum.rasa.com) to help us make this feature
ready for production.

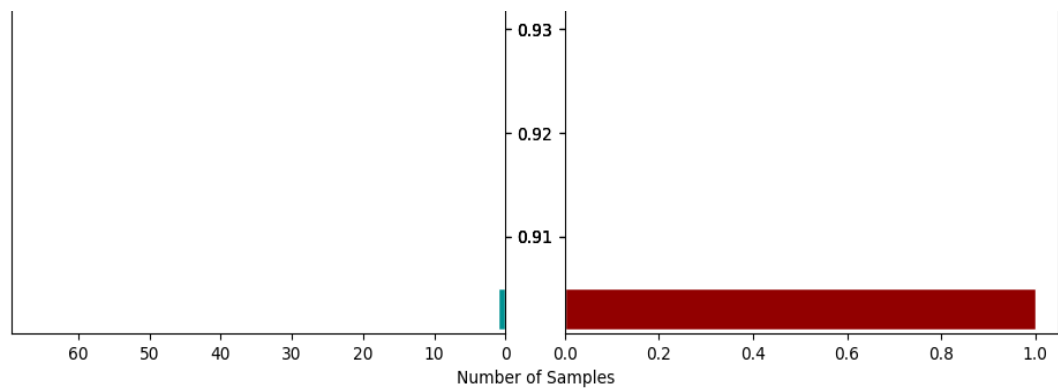
```

```

/home/sriparole/miniconda3/envs/env_rasa2.8/lib/python3.7/site-
packages/rasa/shared/utils/io.py:97: UserWarning: the value of
'evaluate_every_number_of_epochs' is greater than the value of
'epochs'. No evaluation will occur.
2023-12-06 15:17:04 INFO      rasa.shared.utils.validation - The
'version' key is missing in the training data file /home/sriparole
/Documents/TP_RASA_ManonAugeCouffin/CHATBOT_V1/tests
/test_stories.yml. Rasa Open Source will read the file as a version
'2.0' file. See https://rasa.com/docs/rasa/training-data-format.
Processed story blocks: 100%|██████████| 7/7 [00:00<00:00,
2356.54it/s, # trackers=1]
2023-12-06 15:17:04 INFO      rasa.core.test - Evaluating 7 stories
Progress:
100%|████████████████████████████████████████████████████████████████████████████████| 7/7
[00:00<00:00, 9.00it/s]
2023-12-06 15:17:05 INFO      rasa.core.test - Finished collecting
predictions.
2023-12-06 15:17:05 INFO      rasa.core.test - Evaluation Results
on END-T0-END level:
2023-12-06 15:17:05 INFO      rasa.core.test - Correct:          7
/ 7
2023-12-06 15:17:05 INFO      rasa.core.test - Accuracy:
1.000
2023-12-06 15:17:05 INFO      rasa.core.test - Stories report saved
to results/story_report.json.
2023-12-06 15:17:05 INFO      rasa.nlu.test - Evaluation for entity
extractor: TEDPolicy
2023-12-06 15:17:05 WARNING    rasa.model_testing - No labels to
evaluate. Skip evaluation.
2023-12-06 15:17:05 INFO      rasa.nlu.test - Classification report
saved to results/TEDPolicy_report.json.
2023-12-06 15:17:05 INFO      rasa.nlu.test - Every entity was
predicted correctly by the model.
/home/sriparole/miniconda3/envs/env_rasa2.8/lib/python3.7/site-
packages/rasa/utils/plotting.py:109: UserWarning: Attempting to set
identical left == right == -0.5 results in singular
transformations; automatically expanding.
    norm=LogNorm(vmin=zmin, vmax=zmax),
/home/sriparole/miniconda3/envs/env_rasa2.8/lib/python3.7/site-
packages/rasa/utils/plotting.py:109: UserWarning: Attempting to set
identical bottom == top == -0.5 results in singular
transformations; automatically expanding.
    norm=LogNorm(vmin=zmin, vmax=zmax),
2023-12-06 15:17:06 INFO      rasa.utils.plotting - Confusion
matrix, without normalization:
[]
2023-12-06 15:17:08 INFO      rasa.core.test - Evaluation Results
on ACTION level:
2023-12-06 15:17:08 INFO      rasa.core.test - Correct:
35 / 35
2023-12-06 15:17:08 INFO      rasa.core.test - F1-Score:
1.000
2023-12-06 15:17:08 INFO      rasa.core.test - Precision:
1.000
2023-12-06 15:17:08 INFO      rasa.core.test - Accuracy:
1.000
2023-12-06 15:17:08 INFO      rasa.core.test - In-data fraction: 1
2023-12-06 15:17:08 INFO      rasa.utils.plotting - Confusion
matrix, without normalization:
[[16  0  0  0  0  0  0]

```



A quoi correspond ces images ?

La première image est une représentation de la matrice de confusion des intentions. La seconde est son histogramme. Comme on peut le voir, il y a seulement une erreur.

6.3.2- Evaluation des Intentions - B

Étudier le contenu des fichiers `intent_errors.json` et `intent_report.json`

`intent_errors.json`

```
[
  {
    "text": "good afternoon",
    "intent": "goodbye",
    "intent_prediction": {
      "name": "greet",
      "confidence": 0.9030025601387024
    }
  }
]
```

`intent_report.json`

```
{
  "mood_great": {
    "precision": 1.0,
    "recall": 1.0,
    "f1-score": 1.0,
    "support": 14,
    "confused_with": {}
  },
  "goodbye": {
    "precision": 1.0,
    "recall": 0.9090909090909091,
    "f1-score": 0.9523809523809523,
    "support": 11,
    "confused_with": {
      "greet": 1
    }
  },
}
```



```

"bot_challenge": {
  "precision": 1.0,
  "recall": 1.0,
  "f1-score": 1.0,
  "support": 4,
  "confused_with": {}
},
"greet": {
  "precision": 0.9285714285714286,
  "recall": 1.0,
  "f1-score": 0.962962962962963,
  "support": 13,
  "confused_with": {}
},
"affirm": {
  "precision": 1.0,
  "recall": 1.0,
  "f1-score": 1.0,
  "support": 6,
  "confused_with": {}
},
"deny": {
  "precision": 1.0,
  "recall": 1.0,
  "f1-score": 1.0,
  "support": 7,
  "confused_with": {}
},
"mood_unhappy": {
  "precision": 1.0,
  "recall": 1.0,
  "f1-score": 1.0,
  "support": 14,
  "confused_with": {}
},
"accuracy": 0.9855072463768116,
"macro avg": {
  "precision": 0.9897959183673469,
  "recall": 0.987012987012987,
  "f1-score": 0.9879062736205594,
  "support": 69
},
"weighted avg": {
  "precision": 0.9865424430641822,
  "recall": 0.9855072463768116,
  "f1-score": 0.9854305651407099,
  "support": 69
}
}

```

Vous pouvez consulter internet pour comprendre le rôle des métriques utilisées en complément du cours 4 EVALUATION, par exemple : <https://towardsdatascience.com/multi-class-metrics-made-simple-part-ii-the-f1-score-eb8b2c2ca1>
(<https://towardsdatascience.com/multi-class-metrics-made-simple-part-ii-the-f1-score-eb8b2c2ca1>)

A quoi correspondent-il ? Quelle informations peut-on en déduire ?

L'erreur qu'il y a eu est que le chatbot a confondu la signification d'un "good afternoon".
Pour lui c'était un goodbye alors que normalment c'était un greet.

```
-----  
# None of the test stories failed - all good!  
-----
```

Que concluez-vous ?

Tous les tests ont été concluants.

6.3.4- Evaluation des stories -B

Etudier le contenu du fichier story_report.json

```
-----  
{  
  "utter_greet": {  
    "precision": 1.0,  
    "recall": 1.0,  
    "f1-score": 1.0,  
    "support": 5  
  },  
  "action_listen": {  
    "precision": 1.0,  
    "recall": 1.0,  
    "f1-score": 1.0,  
    "support": 16  
  },  
  "utter_iamabot": {  
    "precision": 1.0,  
    "recall": 1.0,  
    "f1-score": 1.0,  
    "support": 1  
  },  
  "mood_great": {  
    "precision": 1.0,  
    "recall": 1.0,  
    "f1-score": 1.0,  
    "support": 2  
  },  
  "goodbye": {  
    "precision": 1.0,  
    "recall": 1.0,  
    "f1-score": 1.0,  
    "support": 2  
  },  
  "bot_challenge": {  
    "precision": 1.0,  
    "recall": 1.0,  
    "f1-score": 1.0,  
    "support": 1  
  },  
  "utter_goodbye": {  
    "precision": 1.0,
```

```
    "recall": 1.0,
    "f1-score": 1.0,
    "support": 4
},
"greet": {
    "precision": 1.0,
    "recall": 1.0,
    "f1-score": 1.0,
    "support": 5
},
"utter_cheer_up": {
    "precision": 1.0,
    "recall": 1.0,
    "f1-score": 1.0,
    "support": 3
},
"affirm": {
    "precision": 1.0,
    "recall": 1.0,
    "f1-score": 1.0,
    "support": 1
},
"deny": {
    "precision": 1.0,
    "recall": 1.0,
    "f1-score": 1.0,
    "support": 2
},
"mood_unhappy": {
    "precision": 1.0,
    "recall": 1.0,
    "f1-score": 1.0,
    "support": 3
},
"utter_did_that_help": {
    "precision": 1.0,
    "recall": 1.0,
    "f1-score": 1.0,
    "support": 3
},
"utter_happy": {
    "precision": 1.0,
    "recall": 1.0,
    "f1-score": 1.0,
    "support": 3
},
"accuracy": 1.0,
"macro avg": {
    "precision": 1.0,
    "recall": 1.0,
    "f1-score": 1.0,
    "support": 51
},
"weighted avg": {
    "precision": 1.0,
    "recall": 1.0,
    "f1-score": 1.0,
    "support": 51
},
"conversation_accuracy": {
```

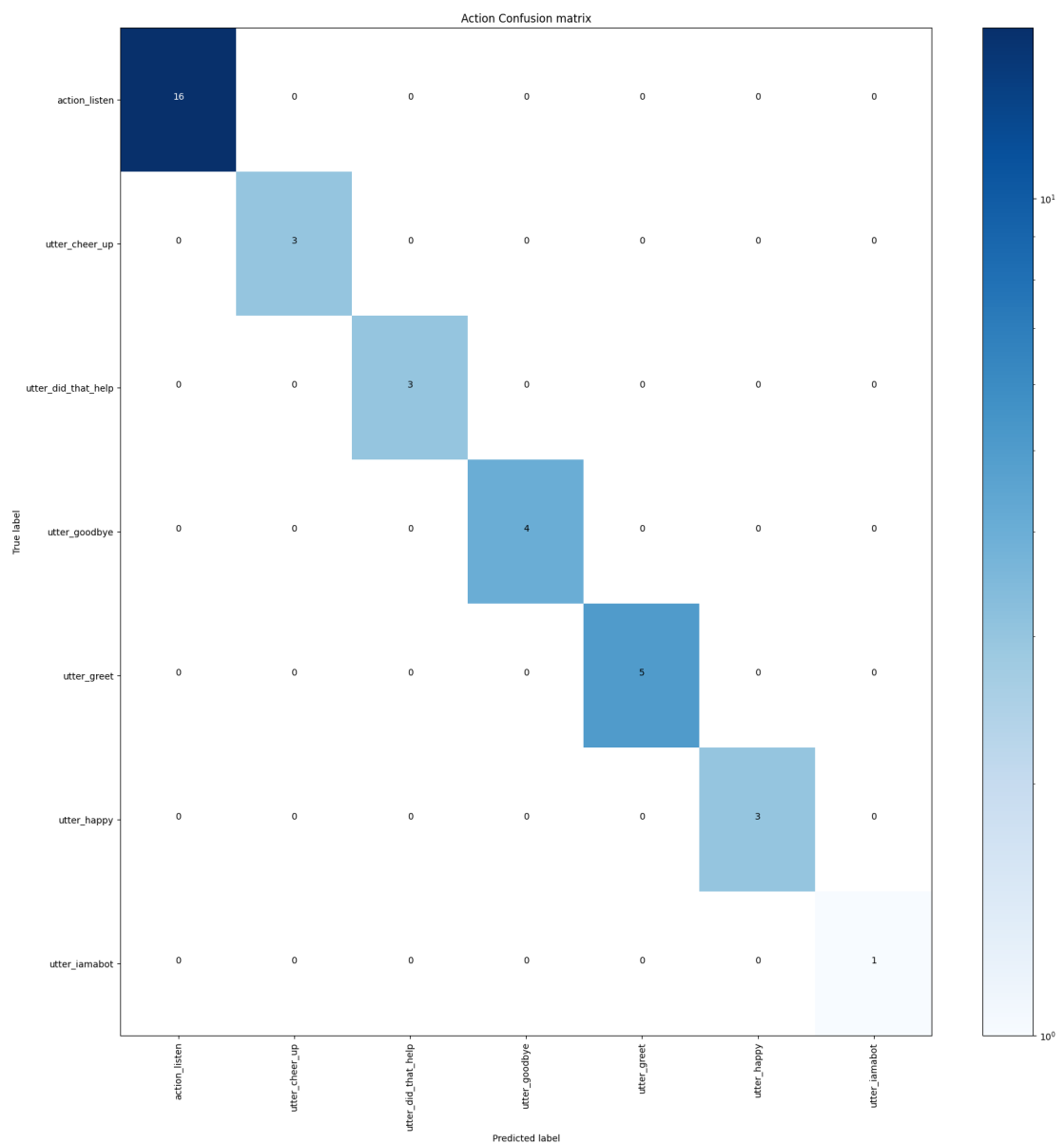
```
"accuracy": 1.0,  
"correct": 7,  
"with_warnings": 0,  
"total": 7  
}  
}
```

A quoi correspond-il ce fichier ?

Il verifit la bonne execution de chacune des actions lors du test.

6.3.5- Evaluation des Intents - C

Ouvrir le fichier story_confusion_matrix.png



A quoi correspond cette image ?

Cette image est une representation de la matrice de confusin des actions. Comme on peut le voir il n'y a pas d'erreur.

PARTIE 3 : DEVELOPPEMENT DU CHATBOT

7- Lancement du chatbot en mode commande

7.1- Depuis le répertoire où se trouve le code du chatbot, lancer la commande : `rasa shell`

Vous avez le prompt suivant :

Bot loaded. Type a message and press enter (use '/stop' to exit) :

Your input --> ...

7.2- Essayer différents cas, copier-coller plusieurs exemples (pertinents) de dialogue et commentez leur déroulement.

Ajoutez autant de cellule que d'exemples. Appuyez-vous sur ce que vous avez retenu des questions précédentes. L'objectif est de tester les capacités de ce mini-chatbot et de comprendre ce qui se passe.

```
Your input -> i gonna kill myself
Here is something to cheer you up:
Image: https://i.imgur.com/nGF1K8f.jpg
Did that help you?
Your input -> not really
Bye
```

Dans cette conversation on peut voir que même si il ne connaît pas l'expression il comprend que l'on va mal.

```
Your input -> good afternoon
Hey! How are you?
Your input -> and you ?
Bye
```

Ce test nous permet de comprendre que si il ne comprend pas il jogue pour voir si on va bien ou mal. Dans ce cas le bot ne voit pas de synonyme de sad donc il finit le dialogue comme un happy mood.

8- Lancement du chatbot en mode interactif

8.1- Depuis le répertoire où se trouve le code du chatbot, lancer la commande : `rasa x`

Ouverture du navigateur et de l'application interactive permettant d'enrichir le chatbot. Sélectionner l'onglet en haut à gauche : Talk to your bot (Interactive Learning), assurez-vous qu'un modèle a été sélectionné et tapez votre texte dans la zone du bas : "Start typing a message"

8.2- Essayer différents cas de dialogue (similaires à ceux de la question précédente)

Copier-coller la partie story associée dans les zones de texte, une zone par story. Ajouter des cellules au besoin.

Story 1 :

```
version: "2.0"
stories:
- story: Story from Conversation ID
81f2d478c5644be3b5611d37c271103e
  steps:
  - intent: greet
  - action: utter_greet
  - intent: deny
  - action: utter_goodbye
```

C'est une conversation qui se deroule avec quelqu'un qui va bien.

Story 2 :

```
version: "2.0"
stories:
- story: Story from Conversation ID
efc512705e584383a5a9aa5ea0b66f3a
  steps:
  - intent: greet
  - action: utter_greet
  - intent: mood_unhappy
  - action: utter_cheer_up
  - action: utter_did_that_help
  - intent: mood_great
  - action: utter_happy
```

Cette conversation se deroule avec quelqu'un qui va mal et qui va mieux après avoir reçu l'image du bot.

PARTIE 4 : EVOLUTION DU CHATBOT

9- Ajout de nouvelles INTENTS

9.1- Essayer à nouveau un dialogue :

- en répondant 'Good', 'Fine thanks', 'not well'
- en ajoutant, à la fin, des énoncés utilisateurs du type 'Thanks' / 'Thank you' / 'Many thanks'

Que se passe-t-il du point de vue "intent" ?

Le système ne reconnaît pas le remerciement.

Copier-Coller ci-après les "story" correspondantes et commenter. Que suggérez-vous de faire pour améliorer ?

Il faudrait modifier les stories et les intents pour qu'il puisse comprendre le remerciement.

Story 1 :

```
version: "2.0"
stories:
- story: Story from Conversation ID
6ed461528952400eb10709dc0246b665
  steps:
  - intent: greet
  - action: utter_greet
  - intent: deny
  - action: utter_goodbye
```

Dans ce dialogue je suis un un great mood. Je lui dis aller bien et le remercie. Comme on peut le voir ma deuxième interaction avec lui n'est pas comprise. On voit cela a la valeur de l'intent.

Story 2 :

```
version: "2.0"
stories:
- story: Story from Conversation ID
3cfc140c307d465d9d3171b1f960bb8c
  steps:
  - intent: greet
  - action: utter_greet
  - intent: mood_unhappy
  - action: utter_cheer_up
  - action: utter_did_that_help
  - intent: affirm
  - action: utter_happy
  - intent: deny
  - action: utter_goodbye
```

Ici, je suis dans un mauvais mood avec un fin heureuse. Je le remercie à la fin et à nouveau il ne comprend pas.

9.2- Explorer les possibilités de rasa-x pour ajouter :

- une nouvelle intent 'thank_you',
- une réponse 'utter_welcome' et au moins deux réponses possibles : 'you are welcome', 'you are very welcome'...
- une règle prévoyant ce type d'échange à la fin de la conversation : User : 'thank you' / 'thanks' / 'many thanks' Chatbot : 'you are welcome' / 'you are very welcome'

Copier-coller ici une story obtenue après nouveau dialogue intégrant les remerciements et expliquer quelles seront selon vous les étapes d'améliorations suivantes.

```
version: "2.0"
stories:
- story: Story from Conversation ID
2fecbac2f0b04d1e94e84ef3a430e535
  steps:
  - intent: greet
  - action: utter_greet
```

```
- intent: mood_unhappy
- action: utter_cheer_up
- action: utter_did_that_help
- intent: affirm
- action: utter_happy
- intent: deny
- action: utter_goodbye
- intent: goodbye
- action: utter_goodbye
```

Cela n'a rien changé. Il faut entrainer le modèle sinon il n'y a aucun changement.

9.3 - Est-ce que ces nouvelles informations ont bien été intégrées dans le déroulement des dialogues exécutés par la suite ?

Non elles n'ont pas été pris en compte.

Quelles sont les 'intents' associées à l'énoncé 'Thank you' ? Commentez ci-dessous. Selon vous, comment peut-on améliorer la situation ?

Dans le dialogue, il est toujours relié à 'deny'. Il faut entrainer le modèle pour qu'il prenne en compte le nouvel intent.

10- Apprentissage et génération d'un nouveau modèle

10.1- Explorer les fonctionnalités associées à la rubrique Training. Que constatez-vous ?

Dans l'onglet Training, on peut ajouter des intents à l'intérieur de domain.yml. On peut aussi accéder au data NLU, changer les règles, voir les réponses du système et avoir accès aux anciennes stories. On peut aussi lancer l'apprentissage du système.

10.2- Après avoir fait le nécessaire dans chaque rubrique (NLU Training, Responses, Stories, Configuration, Domain, ...) procéder à l'apprentissage via l'interface

Que se passe-t-il si vous allez voir dans la rubrique Models ?

J'ai obtenu un nouveau modèle datant d'aujourd'hui.

10.3- Sélectionner le bon modèle et recommencer les tests. Que constatez-vous ? Illustrer et commenter :

```
version: "2.0"
stories:
- story: Story from Conversation ID
5fe898e8b6f644dd93acb6631b8c4a22
  steps:
    - intent: greet
    - action: utter_greet
    - intent: mood_unhappy
    - action: utter_cheer_up
    - action: utter_did_that_help
    - intent: thank_you
    - action: utter_welcome
    - intent: goodbye
    - action: utter_goodbye
```


Il connaît à présent le remerciement.

11- Impact sur les données du chatbot

Examiner le fichier

- data/nlu.yml
- data/stories.yml
- data/rules.yml
- domain.yml

Que constatez vous dans chacun des cas ? Commentez.

Dans le fichier :

- nlu.yml : il y a le nouvel intent 'thank_you'
- stories.yml : il n'y a pas de changement
- rules.yml : il y a la nouvelle règle de remerciement
- domain.yml : la nouvelle action 'utter_welcome' est implémenté

PARTIE 5 : COMPREHENSION ou NLU -

COMMENT CA MARCHE ?

12- Utilisation de SpaCy et des modèles de langage

Les opérations suivantes ont été faites dans l'environnement virtuel env_rasa2.8

```
## installation de la librerie SpaCy pour les traitement NLU et
modèles de langage
# pip3 --use-feature=2020-resolver install rasa[spacy]

## téléchargement du modèle d'embedding anglais : https://spacy.io
/models/en
# python3 -m spacy download en_core_web_md

## téléchargement du modèle d'embedding anglais : https://spacy.io
/models/fr
# python3 -m spacy download fr_core_news_md
```

13- Consultez les site web restectif et expliquer ci-dessous quelles sont les principales caractéristiques de ces modèles

```
en_core_web_md
- Language : enEnglish
- Type : Vocabulary, syntax, entities, vectors
- Genre : written text (blogs, news, comments)
- Size : 40 MB
```

```
fr_core_news_md
- Language : frFrench
- Type : Vocabulary, syntax, entities, vectors
- Genre : written text (news, media)
- Size : 43 MB
```

14- Exécuter la séquence python suivante.

pprint permettra un affichage indenté du résultat de la partie compréhension (NLU)

```
In [1]: %matplotlib inline
import logging, io, json, warnings
logging.basicConfig(level="INFO")
warnings.filterwarnings('ignore')

def pprint(o):
    # small helper to make dict dumps a bit prettier
    print(json.dumps(o, indent=2))
```

```
In [2]: # Positionnez-vous dans le bon répertoire à l'aide du code python su
import os

PATH = '/home/sriparole/Documents/TP_RASA_ManonAugeCouffin/CHATBOT_V1'
os.chdir(PATH)
os.listdir()
```

```
Out[2]: ['rasa.db',
         'tests',
         'domain.yml',
         'data',
         'events.db-wal',
         'endpoints.yml',
         'models',
         'events.db-shm',
         'config.yml',
         'events.db',
         'credentials.yml',
         'actions',
         'results']
```

15- Exécuter la séquence python suivante.

BIEN VERIFIER QUE VOUS AVEZ LANCE JUPYTER NOTEBOOK DEPUIS
L'ENVIRONNEMENT VIRTUEL OU RASA EST INSTALLE (env_rasa2.8)

Les modèles chargés peuvent être utilisés pour traiter les énoncés des utilisateurs

```
In [3]: import rasa.nlu
import rasa.core
import spacy

print("rasa.nlu: {} rasa.core: {}".format(rasa.nlu.__version__, rasa.
print("Loading spaCy language model...")
#print(spacy.load('fr')("Bonjour chez vous! Le modèle français est cl
print(spacy.load('fr_core_news_md')("Bonjour chez vous! Le modèle fra
#print(spacy.load('en')("Hello world! The English model has been loa
print(spacy.load('en_core_web_md')("Hello world! The English model ha
```

```
2023-12-19 10:31:37.344287: W tensorflow/stream_executor/platform/d
efault/dso_loader.cc:59] Could not load dynamic library 'libcudart.
so.10.1'; dLError: libcudart.so.10.1: cannot open shared object fil
e: No such file or directory
```

```
2023-12-19 10:31:37.344319: I tensorflow/stream_executor/cuda/cudar
t_stub.cc:29] Ignore above cudart dLError if you do not have a GPU
set up on your machine.
```

```
2023-12-19 10:31:38.120200: W tensorflow/stream_executor/platform/d
efault/dso_loader.cc:59] Could not load dynamic library 'libcuda.s
o.1'; dLError: libcuda.so.1: cannot open shared object file: No suc
h file or directory
```

```
2023-12-19 10:31:38.120236: W tensorflow/stream_executor/cuda/cuda_
driver.cc:312] failed call to cuInit: UNKNOWN ERROR (303)
```

```
2023-12-19 10:31:38.120254: I tensorflow/stream_executor/cuda/cuda_
diagnostics.cc:156] kernel driver does not appear to be running on
this host (sriparole-virtual-machine): /proc/driver/nvidia/version
does not exist
```

```
rasa.nlu: 2.8.0 rasa.core: 2.8.0
```

```
Loading spaCy language model...
```

```
Bonjour chez vous! Le modèle français est chargé
```

```
Hello world! The English model has ben loaded
```

16- Exécuter la séquence python suivante.

BIEN VERIFIER QUE VOUS AVEZ LANCE JUPYTER NOTEBOOK DEPUIS

L'ENVIRONNEMENT VIRTUEL OU RASA EST INSTALLE (env_rasa2.8)

La partie nlu peut être utilisée dans du code python pour développer ses propres applications. Exécuter le code suivant. Il faut construire un interpréteur qui correspond à la partie pipeline

```

In [4]: import os
from rasa.shared.nlu.training_data.loading import load_data
from rasa.nlu.config import RasaNLUModelConfig
from rasa.nlu.model import Trainer
from rasa.nlu import config
from os import system

# Adapter le chemin à votre contexte
PATH = '/home/sriparole/Documents/TP_RASA_ManonAugeCouffin/CHATBOT_V1'
os.chdir(PATH)
os.listdir()
# Compléter avec le chemin vers votre environnement virtuel créé lors
print(os.getcwd())
print(os.listdir('data'))
warnings.filterwarnings('ignore')

# loading the nlu training samples
training_data = load_data("./data/nlu.yml")

# trainer to educate our pipeline
# décommenter le pipeline si besoin
trainer = Trainer(config.load("./config.yml"))

# train the model!
interpreter = trainer.train(training_data)

# store it for future use
model_directory = trainer.persist("./models/nlu", fixed_model_name="c

/home/sriparole/Documents/TP_RASA_ManonAugeCouffin/CHATBOT_V1
['nlu.yml', 'stories.yml', 'rules.yml']

INFO:rasa.nlu.model:Starting to train component WhitespaceTokenizer
INFO:rasa.nlu.model:Finished training component.
INFO:rasa.nlu.model:Starting to train component RegexFeaturizer
INFO:rasa.nlu.model:Finished training component.
INFO:rasa.nlu.model:Starting to train component LexicalSyntacticFeaturizer
INFO:rasa.nlu.model:Finished training component.
INFO:rasa.nlu.model:Starting to train component CountVectorsFeaturizer
INFO:rasa.nlu.featurizers.sparse_featurizer.count_vectors_featurizer:83 vocabulary items were created for text attribute.
INFO:rasa.nlu.model:Finished training component.
INFO:rasa.nlu.model:Starting to train component CountVectorsFeaturizer
INFO:rasa.nlu.featurizers.sparse_featurizer.count_vectors_featurizer:714 vocabulary items were created for text attribute.
INFO:rasa.nlu.model:Finished training component.
INFO:rasa.nlu.model:Starting to train component DIETClassifier
2023-12-19 10:31:44.103744: I tensorflow/core/platform/cpu_feature_guard.cc:142] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instruction sets in performance-critical operations: AVX2 FMA
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
2023-12-19 10:31:44.107600: I tensorflow/core/platform/profile_utils/cpu_utils.cc:104] CPU Frequency: 3696000000 Hz
2023-12-19 10:31:44.107860: I tensorflow/compiler/xla/service/service.cc:168] XLA service 0xc27ea0 initialized for platform Host (this

```

```

does not guarantee that XLA will be used). Devices:
2023-12-19 10:31:44.107874: I tensorflow/compiler/xla/service/service.cc:176] StreamExecutor device (0): Host, Default Version
Epochs: 100%|██████████| 100/100 [00:14<00:00, 7.08it/s, t_loss=1.29, i_acc=0.986]
INFO:rasa.nlu.model:Finished training component.
INFO:rasa.nlu.model:Starting to train component EntitySynonymMapper
INFO:rasa.nlu.model:Finished training component.
INFO:rasa.nlu.model:Starting to train component ResponseSelector
INFO:rasa.nlu.selectors.response_selector:Retrieval intent parameter was left to its default value. This response selector will be trained on training examples combining all retrieval intents.
INFO:rasa.nlu.model:Finished training component.
INFO:rasa.nlu.model:Starting to train component FallbackClassifier
INFO:rasa.nlu.model:Finished training component.
INFO:rasa.nlu.model:Successfully saved model into '/home/sriparole/Documents/TP_RASA_ManonAugeCouffin/CHATBOT_V1/models/nlu/current'

```

17- Commenter la trace obtenue et aller voir ce qu'il y a sous `models/nlu/current` en exécutant le code suivant

Le modèle a été entraîné à l'aide de la pipeline de configuration.

```

In [5]: #PATH = '/Users/ferrane/Documents/ENSEIGNEMENT/SRI-3A/DOHM/BE_2020/TP1'
#os.chdir(PATH)
#os.listdir()
#Compléter avec le chemin vers votre environnement virtuel créé lors
print(os.getcwd())
print(os.listdir('models/nlu/current'))
warnings.filterwarnings('ignore')

```

```

/home/sriparole/Documents/TP_RASA_ManonAugeCouffin/CHATBOT_V1
['metadata.json', 'component_5_DIETClassifier.tf_model.index', 'component_2_LexicalSyntacticFeaturizer.feature_to_idx_dict.pkl', 'checkpoint', 'component_5_DIETClassifier.entity_tag_specs.json', 'component_1_RegexFeaturizer.patterns.pkl', 'component_5_DIETClassifier.index_label_id_mapping.json', 'component_5_DIETClassifier.label_data.pkl', 'component_5_DIETClassifier.sparse_feature_sizes.pkl', 'component_5_DIETClassifier.tf_model.data-00000-of-00001', 'component_3_CountVectorsFeaturizer.pkl', 'component_5_DIETClassifier.data_example.pkl', 'component_4_CountVectorsFeaturizer.pkl']

```

18- Exécuter le code suivant en remplaçant la chaîne de caractères vide "" par la phrase à analyser représentant l'énoncé de l'utilisateur :

- I am fine thanks
- Thank you
- Thank you very much
- I would like to thank you
- I would rather prefer a puppy
- I would rather have preferred a puppy
- I would like to thank you very much for this

Vous pouvez également proposer des phrases à traiter.

ATTENTION : DUPLIQUER LA CELLULE A CHAQUE NOUVEL ESSAI POUR GARDER
TRACE DES RESULTATS.

In [6]: `pprint(interpreter.parse(""))`

```
{
  "text": "",
  "intent": {
    "name": null,
    "confidence": 0.0
  },
  "entities": []
}
```

```
In [7]: pprint(interpreter.parse("I am fine thanks"))
```

```
{
  "text": "I am fine thanks",
  "intent": {
    "id": -2060081205132760995,
    "name": "thank_you",
    "confidence": 0.9645185470581055
  },
  "entities": [],
  "intent_ranking": [
    {
      "id": -2060081205132760995,
      "name": "thank_you",
      "confidence": 0.9645185470581055
    },
    {
      "id": 8801768551522549680,
      "name": "mood_great",
      "confidence": 0.027468634769320488
    },
    {
      "id": -3637056081654315957,
      "name": "bot_challenge",
      "confidence": 0.005325043108314276
    },
    {
      "id": -3767365156534728919,
      "name": "deny",
      "confidence": 0.0020486831199377775
    },
    {
      "id": 8266924324776116934,
      "name": "goodbye",
      "confidence": 0.0002914549841079861
    },
    {
      "id": -8492416158431785154,
      "name": "affirm",
      "confidence": 0.00020847430278081447
    },
    {
      "id": -6651140628253025356,
      "name": "greet",
      "confidence": 0.00011868860019603744
    },
    {
      "id": -7863660608552312342,
      "name": "mood_unhappy",
      "confidence": 2.035895704466384e-05
    }
  ],
  "response_selector": {
    "all_retrieval_intents": [],
    "default": {
      "response": {
        "id": null,
        "responses": null,
        "response_templates": null,

```

```
        "confidence": 0.0,  
        "intent_response_key": null,  
        "utter_action": "utter_None",  
        "template_name": "utter_None"  
    },  
    "ranking": []  
}  
}
```



```
In [8]: pprint(interpreter.parse("Thank you"))
```

```
{
  "text": "Thank you",
  "intent": {
    "id": -2060081205132760995,
    "name": "thank_you",
    "confidence": 0.9968551397323608
  },
  "entities": [],
  "intent_ranking": [
    {
      "id": -2060081205132760995,
      "name": "thank_you",
      "confidence": 0.9968551397323608
    },
    {
      "id": -3637056081654315957,
      "name": "bot_challenge",
      "confidence": 0.0025232124608010054
    },
    {
      "id": 8801768551522549680,
      "name": "mood_great",
      "confidence": 0.00035780054167844355
    },
    {
      "id": -3767365156534728919,
      "name": "deny",
      "confidence": 0.00016752966621425003
    },
    {
      "id": -8492416158431785154,
      "name": "affirm",
      "confidence": 4.8679765313863754e-05
    },
    {
      "id": 8266924324776116934,
      "name": "goodbye",
      "confidence": 3.417143670958467e-05
    },
    {
      "id": -7863660608552312342,
      "name": "mood_unhappy",
      "confidence": 6.806338660680922e-06
    },
    {
      "id": -6651140628253025356,
      "name": "greet",
      "confidence": 6.68545180815272e-06
    }
  ],
  "response_selector": {
    "all_retrieval_intents": [],
    "default": {
      "response": {
        "id": null,
        "responses": null,
        "response_templates": null,

```

```
        "confidence": 0.0,  
        "intent_response_key": null,  
        "utter_action": "utter_None",  
        "template_name": "utter_None"  
    },  
    "ranking": []  
}  
}
```

```
In [9]: pprint(interpreter.parse("I would prefer a puppy"))
```

```
{
  "text": "I would prefer a puppy",
  "intent": {
    "id": 8801768551522549680,
    "name": "mood_great",
    "confidence": 0.8846246600151062
  },
  "entities": [],
  "intent_ranking": [
    {
      "id": 8801768551522549680,
      "name": "mood_great",
      "confidence": 0.8846246600151062
    },
    {
      "id": -7863660608552312342,
      "name": "mood_unhappy",
      "confidence": 0.10686986148357391
    },
    {
      "id": -3637056081654315957,
      "name": "bot_challenge",
      "confidence": 0.003656716551631689
    },
    {
      "id": 8266924324776116934,
      "name": "goodbye",
      "confidence": 0.0014053646009415388
    },
    {
      "id": -6651140628253025356,
      "name": "greet",
      "confidence": 0.0013280218699947
    },
    {
      "id": -8492416158431785154,
      "name": "affirm",
      "confidence": 0.0008994671516120434
    },
    {
      "id": -2060081205132760995,
      "name": "thank_you",
      "confidence": 0.0008555885287933052
    },
    {
      "id": -3767365156534728919,
      "name": "deny",
      "confidence": 0.0003603448567446321
    }
  ],
  "response_selector": {
    "all_retrieval_intents": [],
    "default": {
      "response": {
        "id": null,
        "responses": null,
        "response_templates": null,

```

```
        "confidence": 0.0,  
        "intent_response_key": null,  
        "utter_action": "utter_None",  
        "template_name": "utter_None"  
    },  
    "ranking": []  
}  
}
```

```
In [10]: pprint(interpreter.parse("I would have preferred a puppy"))
```

```
{
  "text": "I would have preferred a puppy",
  "intent": {
    "id": 8801768551522549680,
    "name": "mood_great",
    "confidence": 0.9408574104309082
  },
  "entities": [],
  "intent_ranking": [
    {
      "id": 8801768551522549680,
      "name": "mood_great",
      "confidence": 0.9408574104309082
    },
    {
      "id": -7863660608552312342,
      "name": "mood_unhappy",
      "confidence": 0.03008890710771084
    },
    {
      "id": 8266924324776116934,
      "name": "goodbye",
      "confidence": 0.022284917533397675
    },
    {
      "id": -2060081205132760995,
      "name": "thank_you",
      "confidence": 0.002500376431271434
    },
    {
      "id": -6651140628253025356,
      "name": "greet",
      "confidence": 0.0016584027325734496
    },
    {
      "id": -8492416158431785154,
      "name": "affirm",
      "confidence": 0.001473387354053557
    },
    {
      "id": -3637056081654315957,
      "name": "bot_challenge",
      "confidence": 0.0008898072992451489
    },
    {
      "id": -3767365156534728919,
      "name": "deny",
      "confidence": 0.00024685030803084373
    }
  ],
  "response_selector": {
    "all_retrieval_intents": [],
    "default": {
      "response": {
        "id": null,
        "responses": null,
        "response_templates": null,

```

```
        "confidence": 0.0,  
        "intent_response_key": null,  
        "utter_action": "utter_None",  
        "template_name": "utter_None"  
    },  
    "ranking": []  
}  
}
```

```
In [11]: pprint(interpreter.parse("I would like to thank you very much for th
```

```
{
  "text": "I would like to thank you very much for this",
  "intent": {
    "id": -2060081205132760995,
    "name": "thank_you",
    "confidence": 0.6712017059326172
  },
  "entities": [],
  "intent_ranking": [
    {
      "id": -2060081205132760995,
      "name": "thank_you",
      "confidence": 0.6712017059326172
    },
    {
      "id": -3767365156534728919,
      "name": "deny",
      "confidence": 0.13264404237270355
    },
    {
      "id": -3637056081654315957,
      "name": "bot_challenge",
      "confidence": 0.0966126024723053
    },
    {
      "id": 8801768551522549680,
      "name": "mood_great",
      "confidence": 0.08993498980998993
    },
    {
      "id": 8266924324776116934,
      "name": "goodbye",
      "confidence": 0.0033780792728066444
    },
    {
      "id": -7863660608552312342,
      "name": "mood_unhappy",
      "confidence": 0.003044979413971305
    },
    {
      "id": -6651140628253025356,
      "name": "greet",
      "confidence": 0.002292061923071742
    },
    {
      "id": -8492416158431785154,
      "name": "affirm",
      "confidence": 0.0008916007936932147
    }
  ],
  "response_selector": {
    "all_retrieval_intents": [],
    "default": {
      "response": {
        "id": null,
        "responses": null,
        "response_templates": null,

```

```

        "confidence": 0.0,
        "intent_response_key": null,
        "utter_action": "utter_None",
        "template_name": "utter_None"
    },
    "ranking": []
}

```

Commentaires :

Pour chacun des tests, "response_selector" est null. Cela est dû au fait que nous interprétons le texte, aucune réponse n'est attendu de la part du système.

La mauvaise interprétation de "I am fine thank you" est due au fait que le système ne doit pas connaître fine en tant que good mood. De plus, le système ne comprend pas bien les phrase longue. En effet, "I would like to thank you very much for this" a un niveau de confiance plus bas que "Thank you" mais il le reconnaît tout de même.

De par ailleurs, les phrases non connues telles que : "I would prefer a puppy" et "I would have preferred a puppy", sont classées en tant que "mood_great" à un haut niveau de confiance.

PARTIE 6 : Evolution du chatbot et évaluation

19- Intégration de nouvelles intents et d'entités

19.1- Que faut-il faire pour prendre en compte de nouvelles intentions (intent) qui permettraient de traiter des phrases comme :

- I would like to see another animal
- Do you have other animals to show
- I want to see another picture

Avec une réponse du type : I don't have any

Avec une réponse du type : Here is another one (avec affichage d'une autre image).

Réponse :

Pour faire cela on doit créer un nouvel intent ainsi qu'une nouvelle action et une nouvelle règle.

Faites le nécessaire au niveau du chatbot pour prendre en compte ces modifications et pour faire fonctionner le chatbot comme prévu.

19.2- Que faut-il faire pour prendre en compte de nouvelles intentions (intent) qui permettraient de traiter des phrases comme :

- I would like to see a puppy
- Do you have any pictures of butterflies
- I want to see a panda

Avec une réponse du type : Let me see ... I found this

19.3) Créer une nouvelle version V2 (pour éviter d'écraser ce qui marche) et intégrer les modifications évoquées dans la questions précédente.

Fichier nlu.yml :

- intent: change_image examples: |
 - I would like to see a [puppy>{"entity": "animal", "value": "dog"}
 - I would like to see a [kitty>{"entity": "animal", "value": "cat"}
 - Do you have any pictures of [butterfly>{"entity": "animal", "value": "butterflies"}
 - I want to see a [cat \(animal\)](#)
 - Can you show me a [dog \(animal\)](#)
 - I would like to see a [butterflies \(animal\)](#)

Ajout d'entities. Cela nous permet d'avoir des synonymes.

Fichier action.py : class ActionShowImage(Action):

```
def name(self) -> Text:
    return "action_show_image"

def run(self, dispatcher: CollectingDispatcher,
        tracker: Tracker,
        domain: Dict[Text, Any]) -> List[Dict[Text, Any]]:

    animal_type = next(tracker.get_latest_entity_values("animal"), None)

    if not animal_type:
        msg = f"I don't have a photo of a {animal_type}...
        Try another animal like a dog, a cat,..."
        dispatcher.utter_message(text=msg)
        return []

    animal_image = animals_db.get(animal_type, None)

    dispatcher.utter_message(text = "Let me see ... I found
    this ... Does it help ? ", image=animal_image)
```

Fichier rules.yml :

- rule: Shows the picture of the requested animal steps:
 - intent: change_image
 - action: action_show_image

Type *Markdown* and LaTeX: α^2

19.4) Illustrer en donnant des exemples de dialogue incluant ce type d'échanges

- User : I would like to see a puppy
- Chatbot : Let me see ... I found this ... Does it help ?

Exemple 1

Exemple 2

19.5) Faire évoluer votre chatbot sur les mêmes principes. Commenter et illustrer ces ajouts (Demander quel est votre animal préféré, ajouter une table lookup pour gérer les synonymes ... ou ajouter d'autres règles

J'ai ajouté un intent animal favori.

20- A partir de rasa x et l'onglet stories, observer les modèles de dialogue associés aux différentes stories.

Faire une capture écran de l'un des graphes les plus représentatifs et insérer le dans la cellule suivante

21- Visualisation Quitter rasa x en fermant la fenêtre du navigateur ET en tapant ^C dans le terminal de lancement.

Une fois rasa x attrêté, tapez la commande : rasa visualize

- Visualisez le fichier Graph.html
- Insérer dans la cellule suivante une copie image de ce graphe (copie écran ou autre)

Que représente-t-il ?

22- Evaluation de la nouvelle version

Constituer un fichier de test adapté à ce nouveau chatbot et reprendre les questions de la partie 2 - Q7 pour visualiser et commenter les résultats obtenus.

23- Bilan

Je me suis arrêté à la question 19.5. Mon code est disponible sur mon github :

https://github.com/ManonCouffin/TP_RASA (https://github.com/ManonCouffin/TP_RASA)