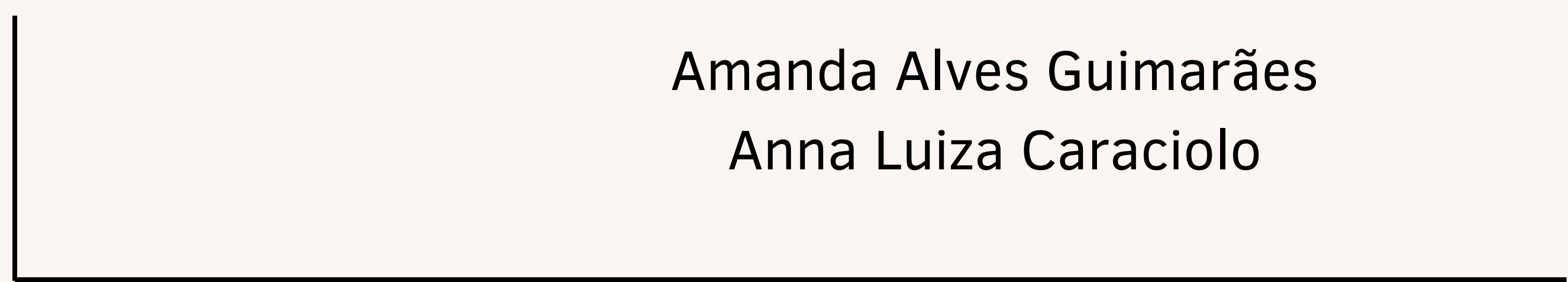




# PROJETO – CHATBOT

## Processamento de Linguagem Natural

Amanda Alves Guimarães  
Anna Luiza Caraciolo



# TÓPICOS DE ABORDAGEM

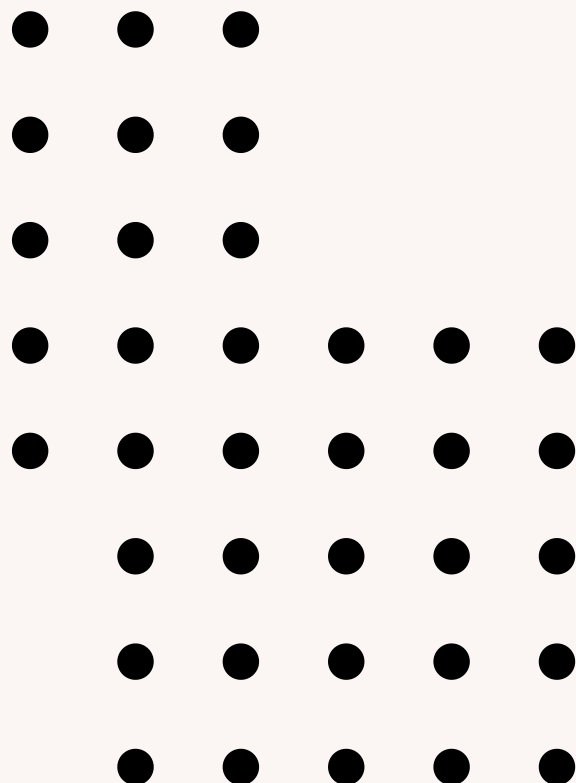
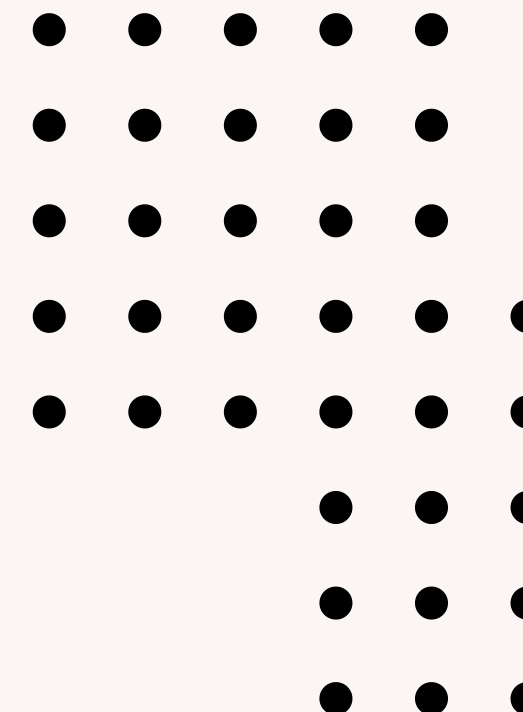
- Contextualização
- Conjunto de dados
- Classificadores de intenção
- Extrator de entidades
- Chatbot utilizando Rasa



---

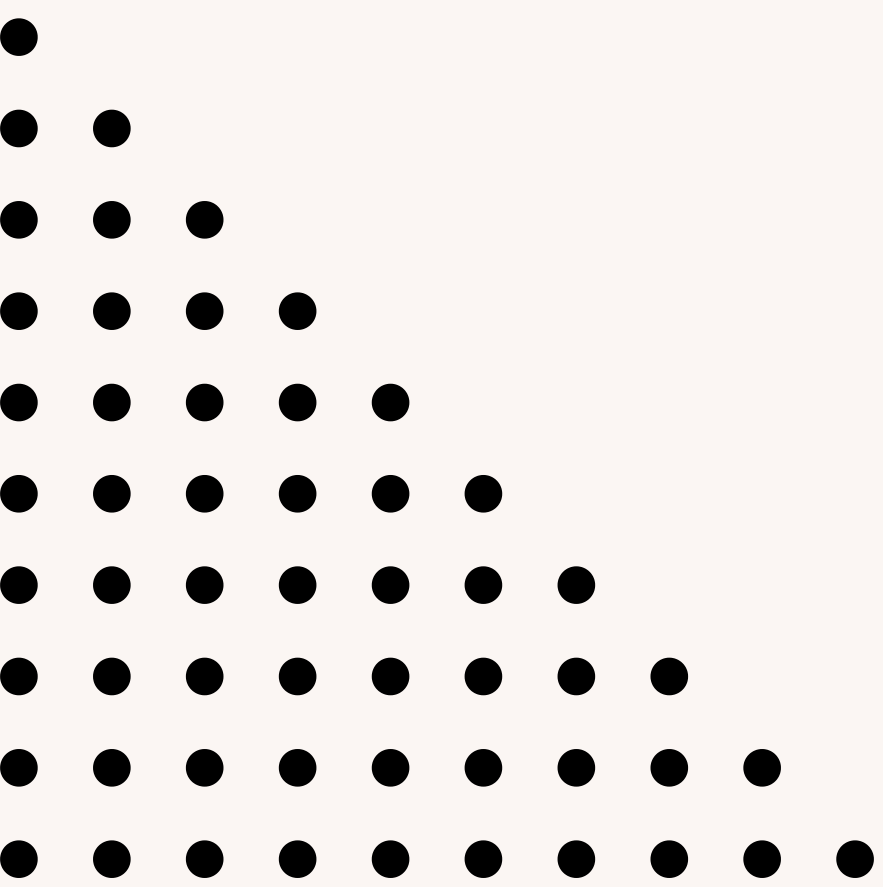
# CONTEXTUALIZAÇÃO

O domínio escolhido para o chatbot foi busca e marcação de voos.

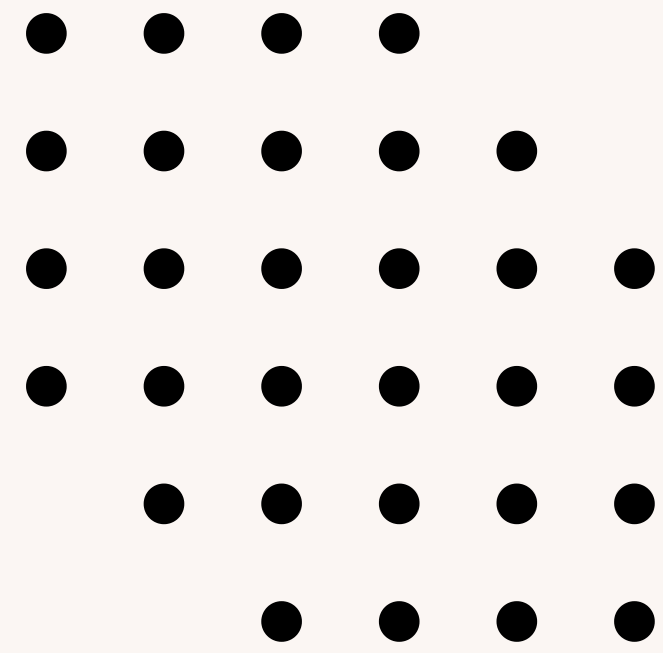




# CONJUNTO DE DADOS

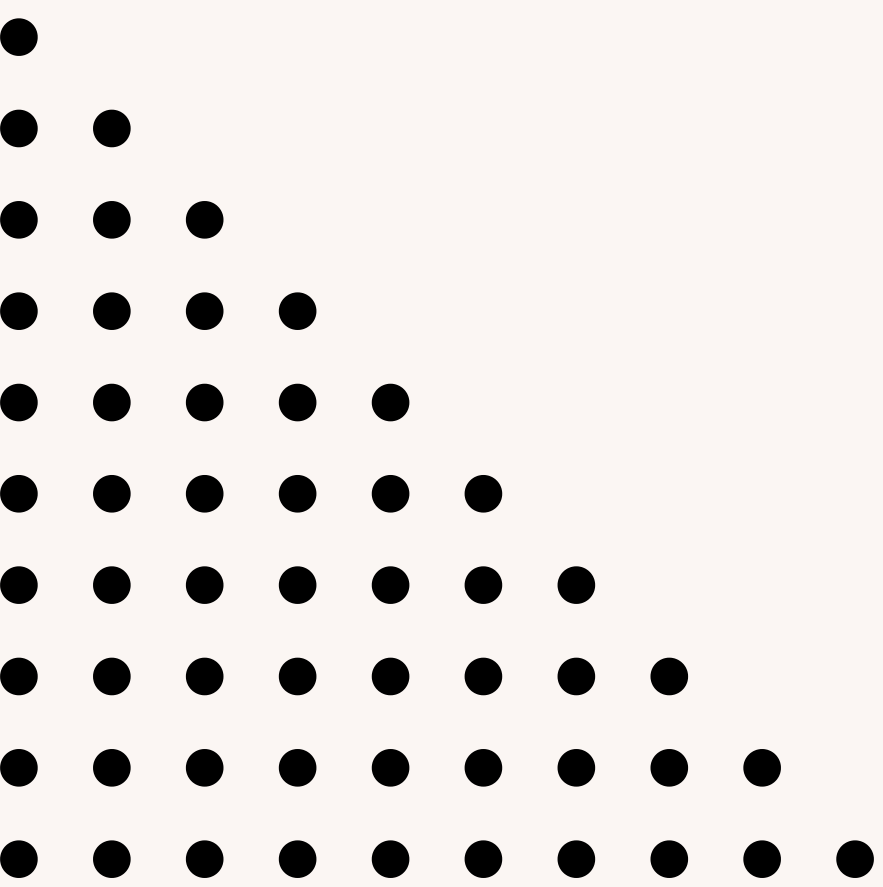
- Subconjunto do Schema-Guided Dialogue (SGD);
  - Pré-processamento:
    - Criação de diretórios para cada classe esperada no modelo;
    - Divisão entre treino e validação, com dois tensores cada;
    - Tokenizer();
    - Padded sequences
- 

# CLASSIFICADORES DE INTENÇÃO



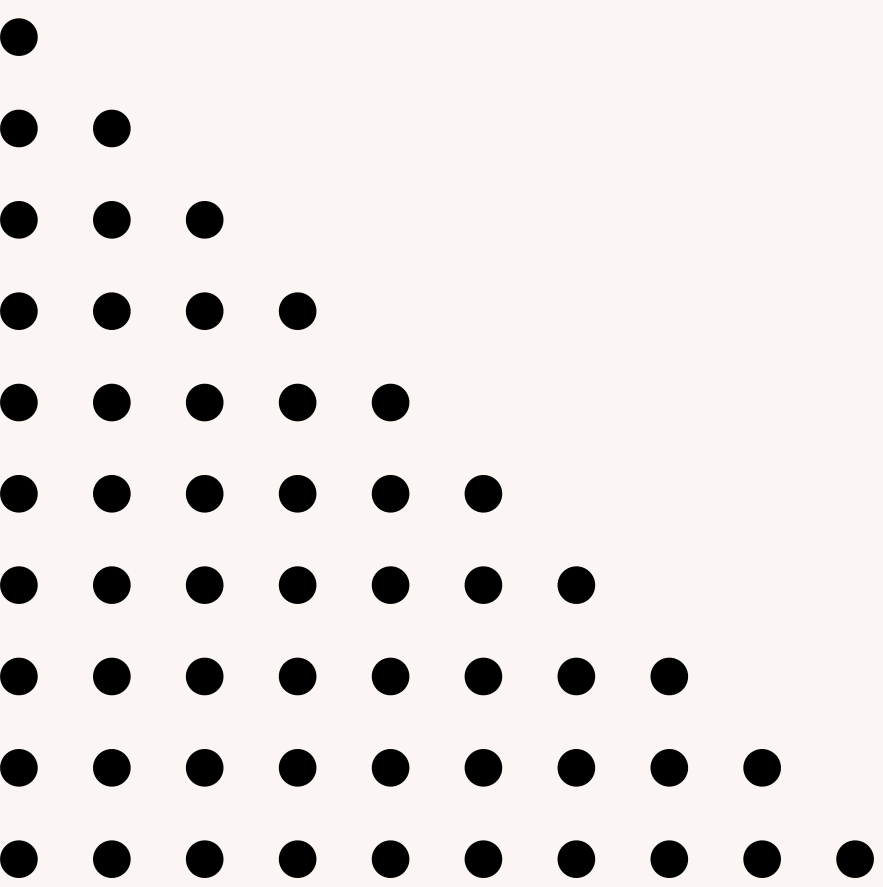


# LSTM

- Biblioteca Keras;
  - Dimensões:
    - Sequential
    - Embedding(input\_dim = 842, output\_dim=100)
    - LSTM(units = 64)
  - Camada de dimensão Dense(5, activation = softmax)
  - Treinamento:
    - 0.89 de acurácia e 0.32 de loss
  - Validação:
    - 0.93 de acurácia e 0.18 de loss
- 



# SVM

- Sklearn;
    - Kernel = linear;
    - Random State = 42;
  - Acurácia do modelo = 0.81;
- 

# CNN

- Biblioteca Keras;
- Dimensões:
  - Optimizer = Adam
  - Loss = categorical\_crossentropy
- Epochs = 10
- Batch size = 5
- Validação:
  - Acurácia de 0.51
- Treinamento:
  - Acurácia de 0.85

```
model.add(layers.Conv1D(128, 5, activation='relu'))  
model.add(layers.GlobalMaxPooling1D())  
model.add(layers.Dense(64, activation='relu'))  
model.add(layers.Dense(5, activation='softmax'))
```

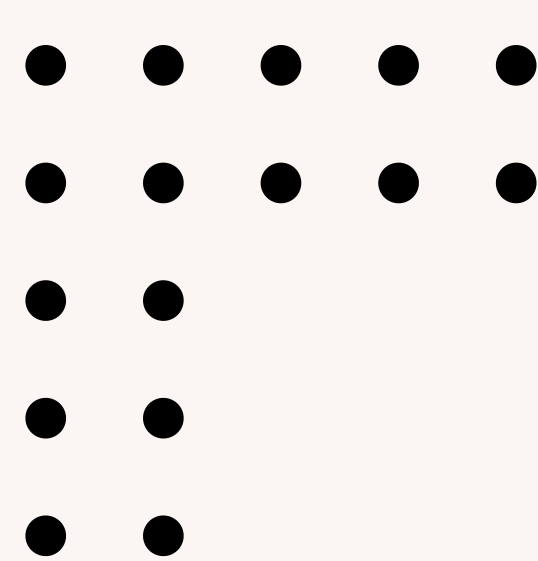


# TRANSFORMER

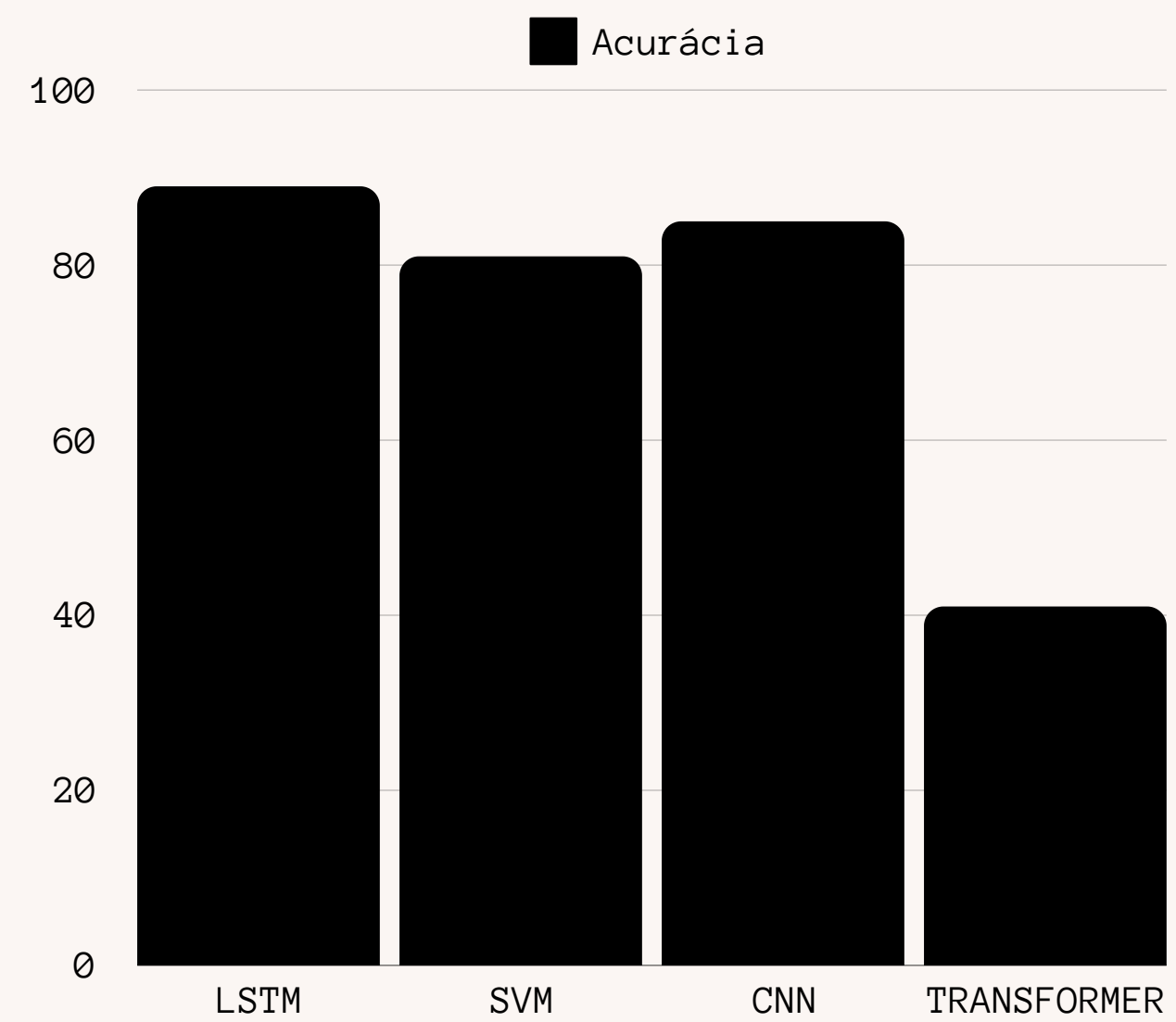
- BERT pré-treinado disponibilizado pela Hugging Face 🙌:

## **TFBertForSequenceClassification**

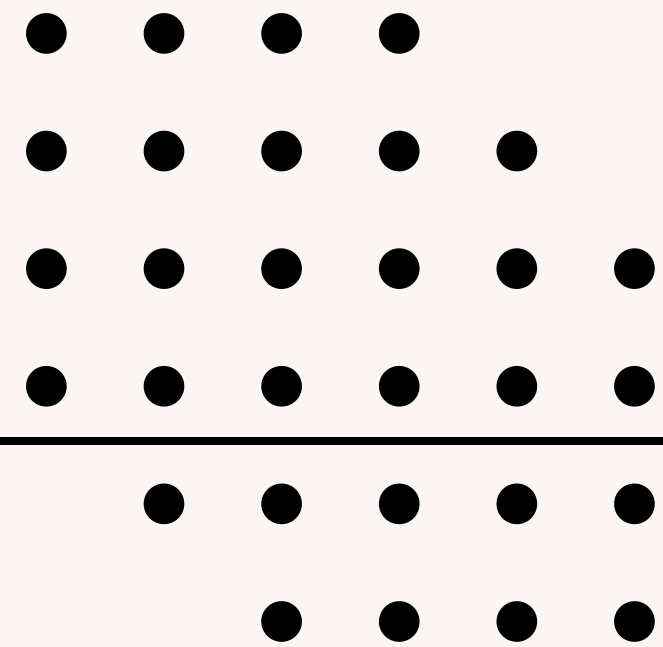
- Redefinição da última camada:
  - otimizador = Adam
  - learning rate =  $2e-5$
  - loss function = sparse categorical cross entropy
- Treinamento:
  - 3 epochs, 0.41 de acurácia e 1.45 de loss



# CLASSIFICADORES DE INTENÇÃO — COMPARAÇÕES




# EXTRATOR DE ENTIDADES



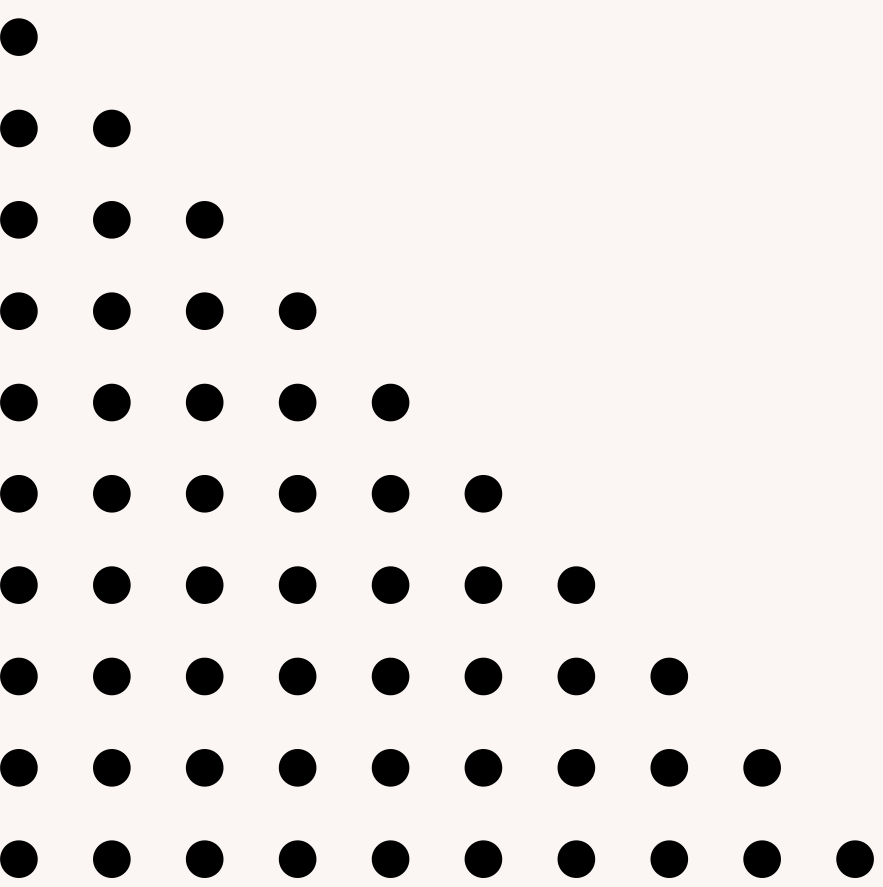


# LSTM EXTRACTOR

- `extractor_lstm = Sequential()`
  - `extractor_lstm.add(Embedding(input_dim=vocab_size+1, output_dim=20))`
  - `extractor_lstm.add(LSTM(units=100))`
  - 
  - Número de unidades na Dense Layer = 296;
  - Função de ativação = softmax;
  - Otimizador = Adam;
  - Loss function = sparse\_categorical\_crossentropy;
  - Acurácia muito baixa;
- 



# TRANSFORMER EXTRACTOR

- Número de unidades na Dense Layer = 512;
  - Função de ativação = softmax;
  - Otimizador = Adam;
  - Loss function = sparse\_categorical\_crossentropy;
  - Batch size = 32;
  - Epochs = 10;
  - Acurácia = 0.41;
- 

**CHATBOT COM O**

**RASA**



# CONFIG.YML

language: en

pipeline:

- name: WhitespaceTokenizer
- name: RegexFeaturizer
- name: LexicalSyntacticFeaturizer
- name: CountVectorsFeaturizer
- name: DIETClassifier
- name: EntitySynonymMapper
- name: ResponseSelector
- name: FallbackClassifier

policies:

- name: MemoizationPolicy
- name: RulePolicy

## intents:

...

- ReserveRoundtripFlights
- ask\_departure\_place
- ask\_arrival\_place
- SearchOnewayFlight

..

## responses:

...

- utter\_SearchRoundtripFlights/ask\_arrival\_place:
- text: "Where would you like to go?"

utter\_SearchOnewayFlight/ask\_departure\_place:

- text: "Where are you leaving from?"

...

# DOMAIN.YML

## rules:

...

- rule: Ask for arrival place when the user wants to search round-trip flights

steps:

- intent: inform

entities:

- location: "New York"
- action: utter\_ask\_arrival\_place

..



# NLU.YML

```
nlu:
```

- intent: SearchOnewayFlight...
- intent: ReserveOnewayFlight...
- intent: SearchRoundtripFlights

```
  examples: |
```

- I need to find round trip flights for [3](amount\_people) people travelling to [washington](location).
- I'm leaving for Philly next Thursday, returning on the [11th](time).
- Can you look for round trip flights?
- I am departing from Seattle, WA and would like to return on the [12th](time) of this month.
- I am traveling to [San Diego](location) tomorrow.
- Can you look up some round trip flights for me?
- Monday next week. I'd be leaving from [Las Vegas](location) and going to SD.
- The [11th](time) of March.

# NLU.YML

```
- intent: faq/longest_flight
  examples: |
    - What is the longest commercial flight today?

- intent: faq/airlines
  examples: |
    - How many airlines are there?

- lookup: entity...

- lookup: location
  examples: |
    - New York
    - Paris
    - London
    - Las Vegas
    - LA
    - NY
    - chicago
```

# STORIES.YML

```
stories:
- story: Search and Reserve Flights ...
- story: Search Round-trip Flights with Details
  steps:
  - intent: SearchRoundtripFlights
  - action: utter_ask_departure_place
  - intent: ask_arrival_place
  - action: utter_ask_arrival_place
  - intent: AskDates
  - action: utter_ask_dates
  - intent: tell_number_of_passengers
  - action: utter_tell_number_of_passengers
```