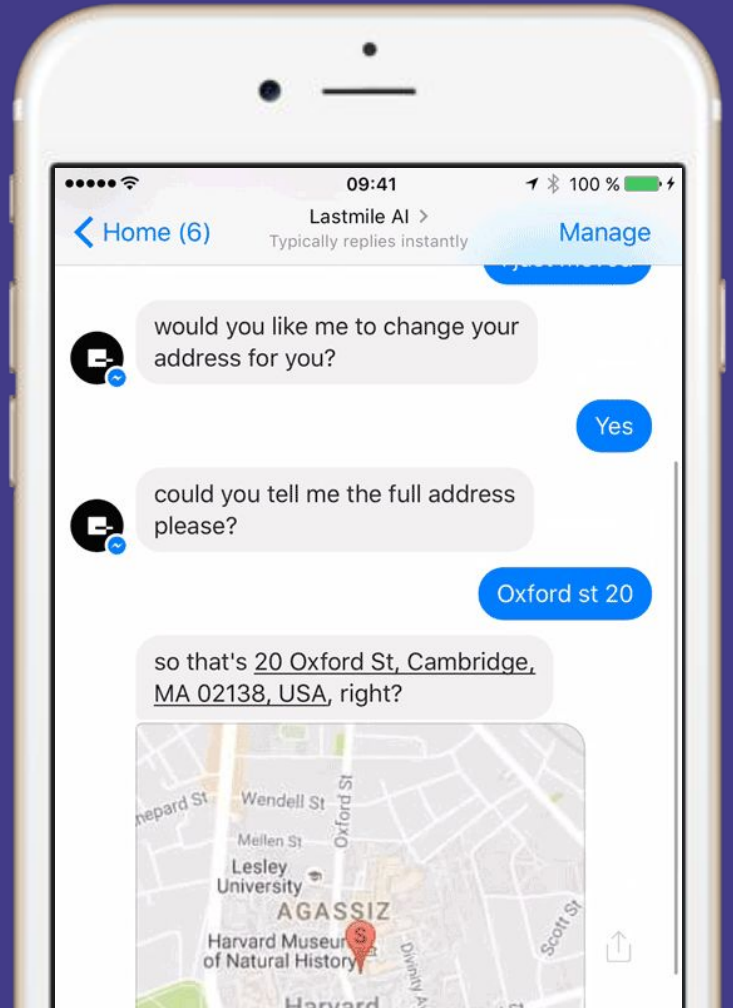
Conversational AI: Building clever chatbots

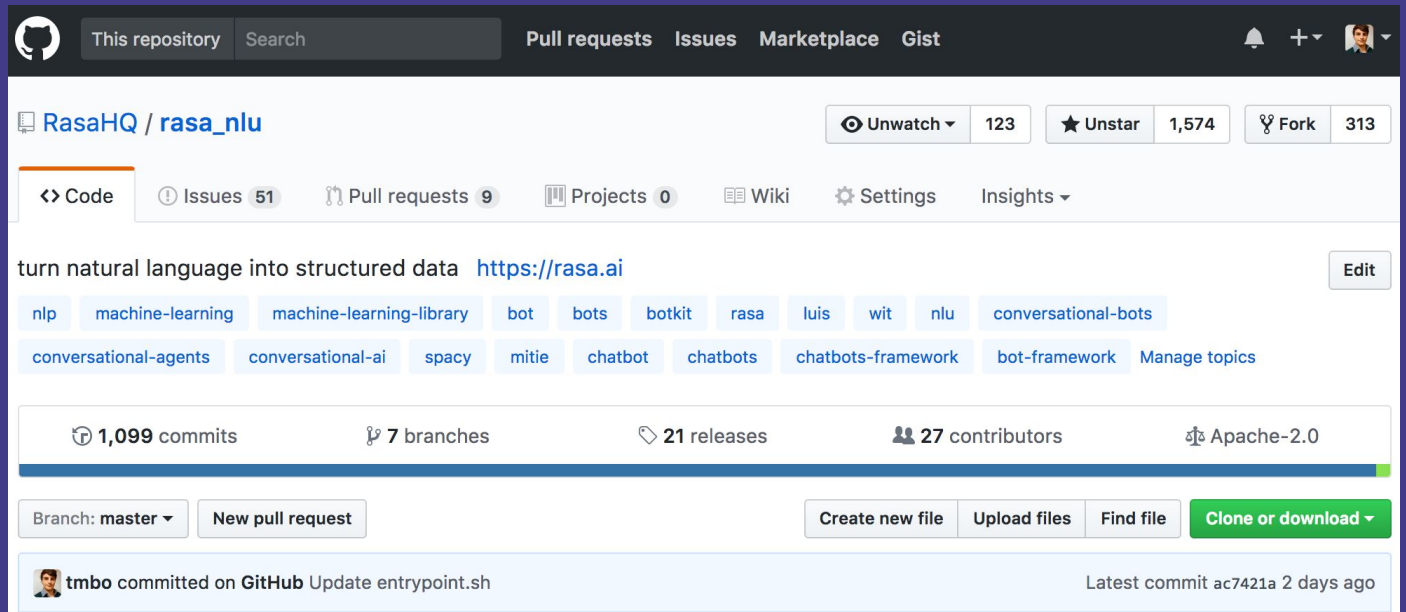
Tom Bocklisch, Lead Engineer

Conversational AI will dramatically change how users interact with you.

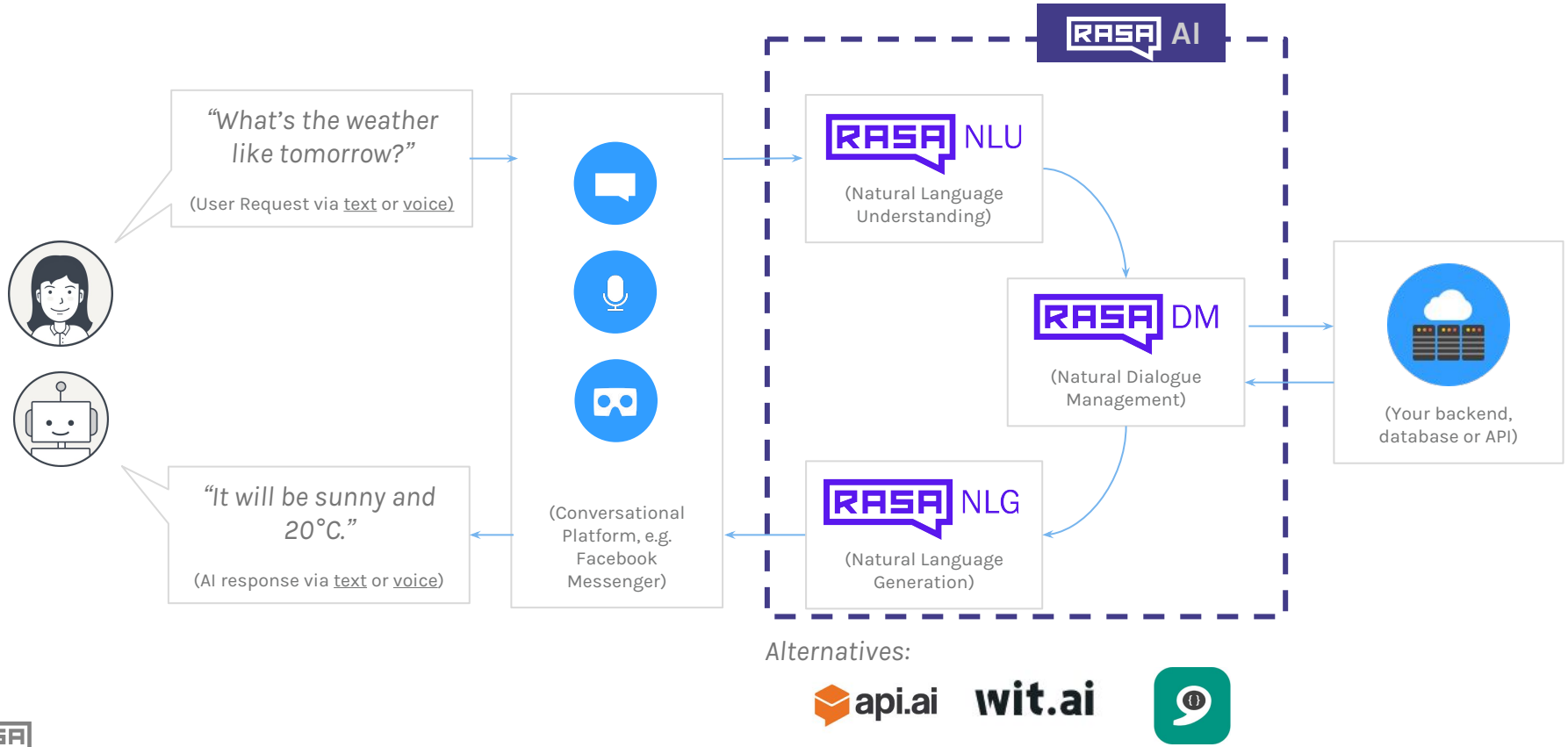
Example of a live Skill:
A customer can change her address via Facebook Messenger



An open source, highly scalable ML framework to build conversational software



Architectural Overview

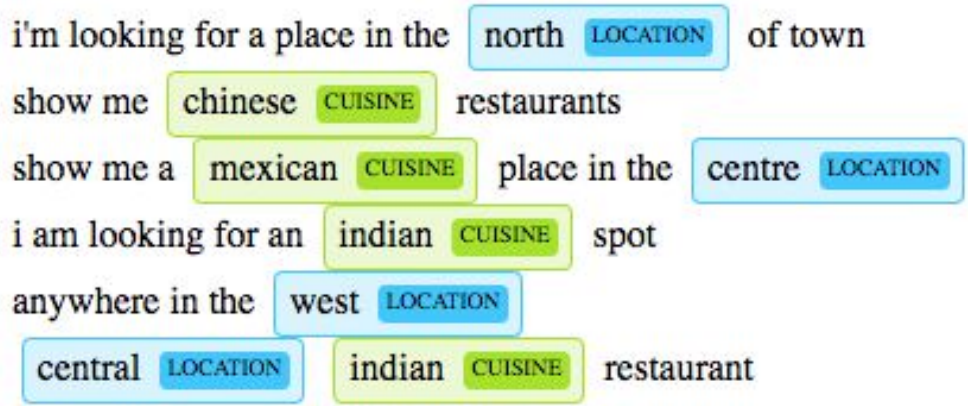


Under The Hood

Natural Language Understanding

Natural Language Understanding

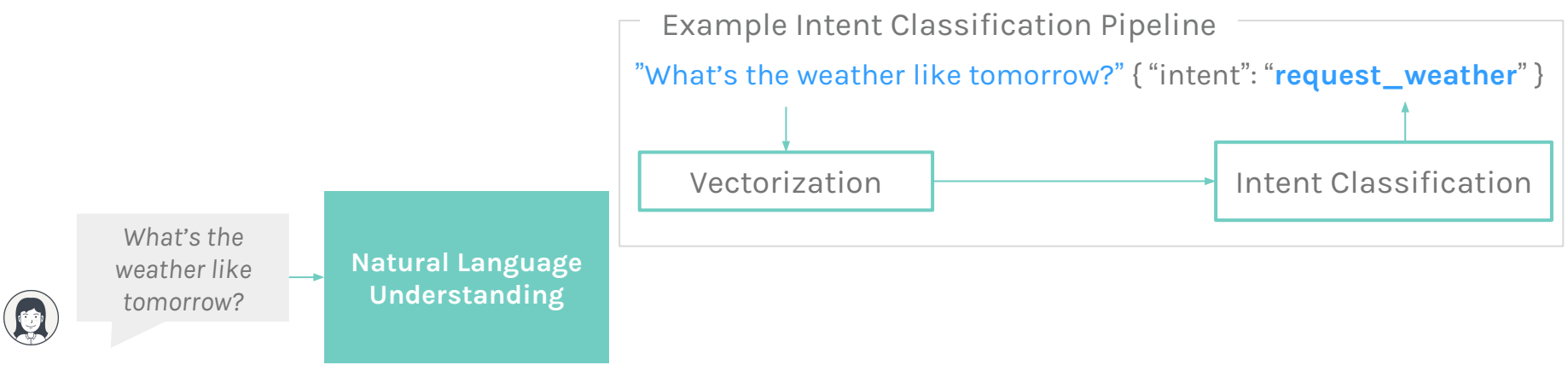
Goal: Create structured data




```
{
  "text": "show me chinese restaurants",
  "intent": "restaurant_search",
  "entities": [
    {
      "start": 8,
      "end": 15,
      "value": "chinese",
      "entity": "cuisine"
    }
  ]
}
```

Under The Hood

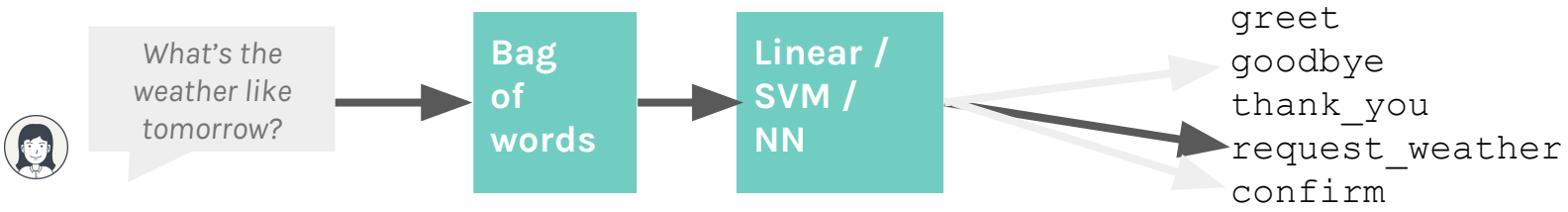
Natural Language Understanding



Natural Language Understanding

Bags are your friend

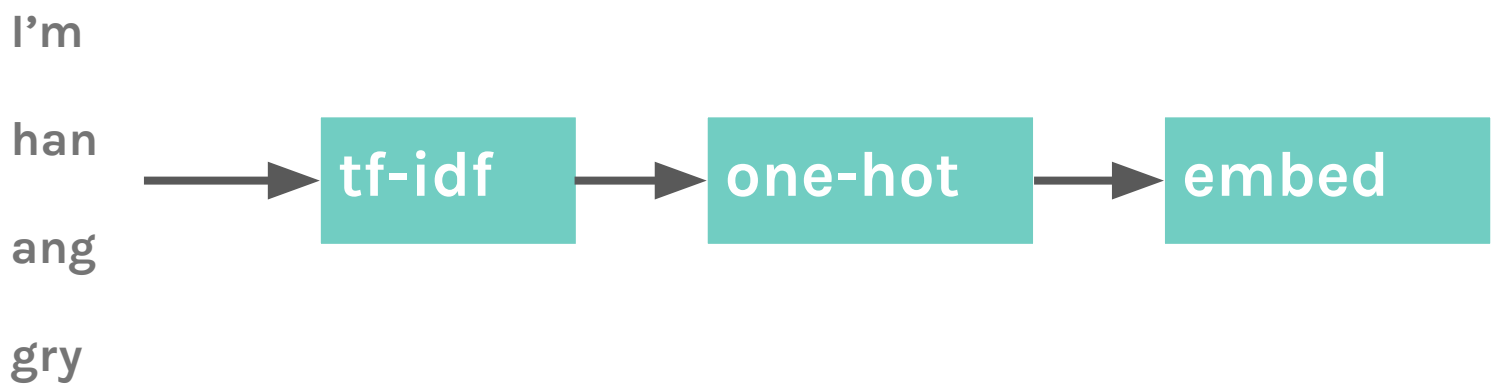
$$\{v_1, \ldots, v_s\} \rightarrow \frac{1}{s} \sum_i v_i$$



Natural Language Understanding

word_bag ⊕ char_feats

“I’m hangry”



Under The Hood

Natural Language Understanding

What's the weather like tomorrow?

Natural Language Understanding

Example Intent Classification Pipeline

"What's the weather like tomorrow?" { "intent": "**request_weather**" }

Vectorization → Intent Classification

Example Entity Extraction Pipeline

"What's the weather like **tomorrow**?" { "date": "**tomorrow**" }

Tokenizer → Part of Speech Tagger → Chunker → Named Entity Recognition → Entity Extraction

RASA

Demo

Demo

2. Configure the model

Configure the model

In [2]:

```
model_config = {
  "pipeline": ["nlp_spacy",
               "ner_crf",
               #"ner_spacy",
               "intent_featurizer_spacy",
               "intent_classifier_sklearn"],
  "language": "en"
}
```

RASA

Demo

3. Train

Training the model

In [3]:

```
# Train NLU model
config = RasaNLUConfig(cmdline_args=model_config)

trainer = Trainer(config)
training_data = load_data("example-data/demo-rasa.json")

# run the training
interpreter = trainer.train(training_data)
logging.info("done")
```

```
INFO:root:Trying to load spacy model with name 'en'
INFO:root:Added 'nlp_spacy' to component cache. Key 'nlp_spacy-en'.
INFO:root:Training data format at example-data/demo-rasa.json is rasa_nlu
INFO:root:Training data stats:
        - intent examples: 38 (4 distinct intents)
        - found intents: affirm, goodbye, greet, restaurant_search
        - entity examples: 7 (2 distinct entities)
        - found entities: cuisine, location

INFO:root:Starting to train component nlp_spacy
INFO:root:Finished training component.
INFO:root:Starting to train component ner_crf
INFO:root:Finished training component.
INFO:root:Starting to train component intent_featurizer_spacy
INFO:root:Finished training component.
INFO:root:Starting to train component intent_classifier_sklearn
[Parallel(n_jobs=1)]: Done  12 out of  12 | elapsed:    0.1s finished
INFO:root:Finished training component.
INFO:root:done
```

```
Fitting 2 folds for each of 6 candidates, totalling 12 fits
```

RASA

Demo

4. Use Model

Playing around with the trained model

In [27]:

```
# i am looking for an italian restaurant in Vienna
result = interpreter.parse(
  "i am looking for an italian restaurant in Vienna")
pprint(result)
```


```
{
  "entities": [
    {
      "start": 20,
      "extractor": "ner_crf",
      "end": 27,
      "value": "italian",
      "entity": "cuisine"
    },
    {
      "start": 42,
      "extractor": "ner_spacy",
      "end": 48,
      "value": "Vienna",
      "entity": "GPE"
    }
  ],
  "intent": {
    "confidence": 0.80703667042349947,
    "name": "restaurant_search"
  },
  "text": "i am looking for an italian restaurant in Vienna",
```


RASA

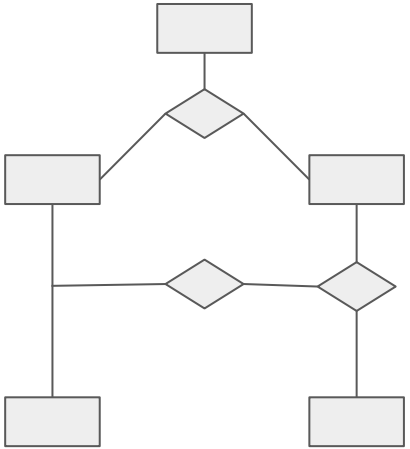
Under The Hood

Dialogue Handling

Dialogue Handling

hand-crafted

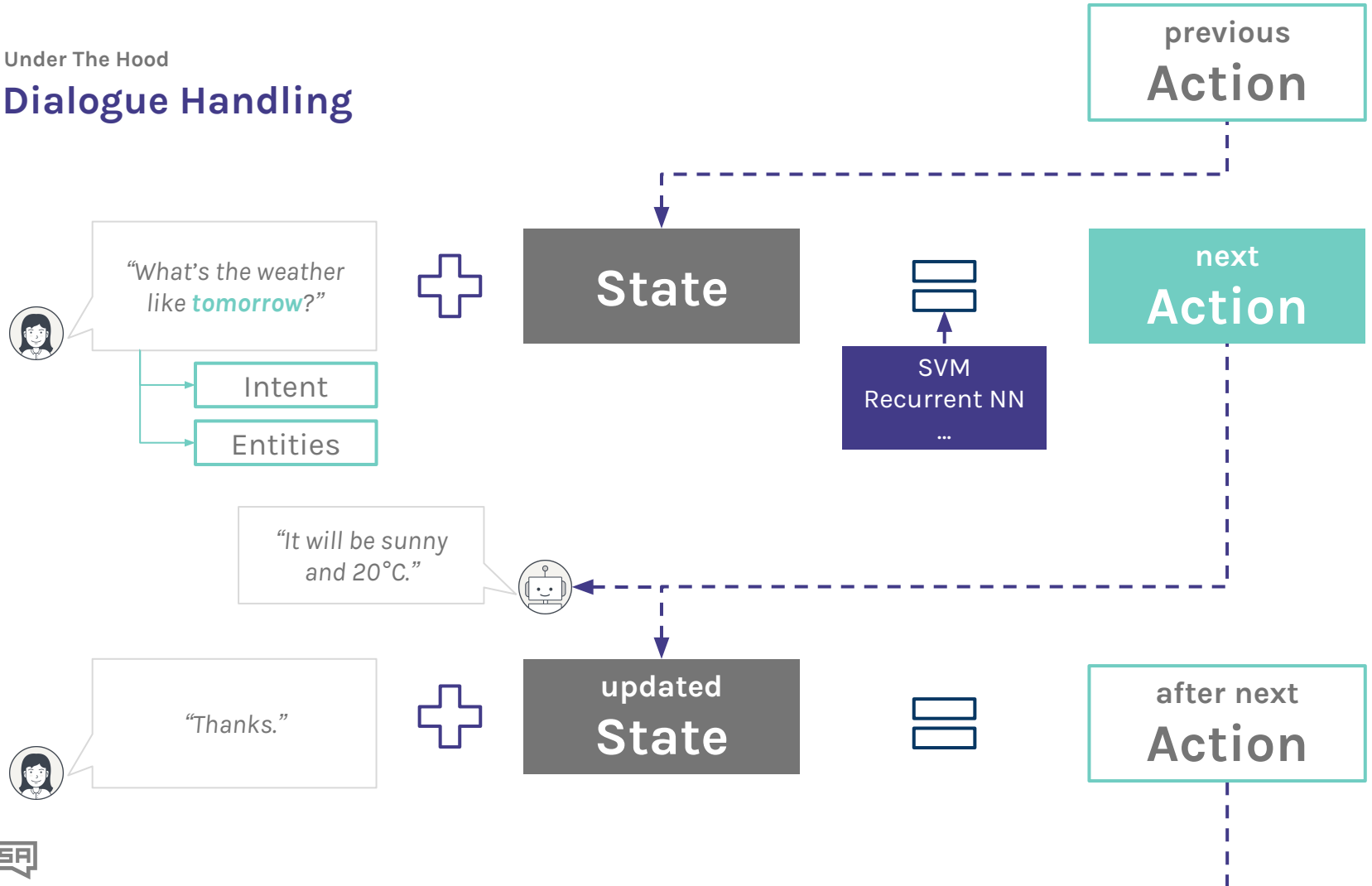
data-driven



Human: *What do you think about messi ?*
Machine: *he 's a great player .*
Human: *what do you think about cleopatra ?*
Machine: *oh , she 's very regal .*
Human: *what do you think about england during the reign of elizabeth ?*
Machine: *it was a great place .*

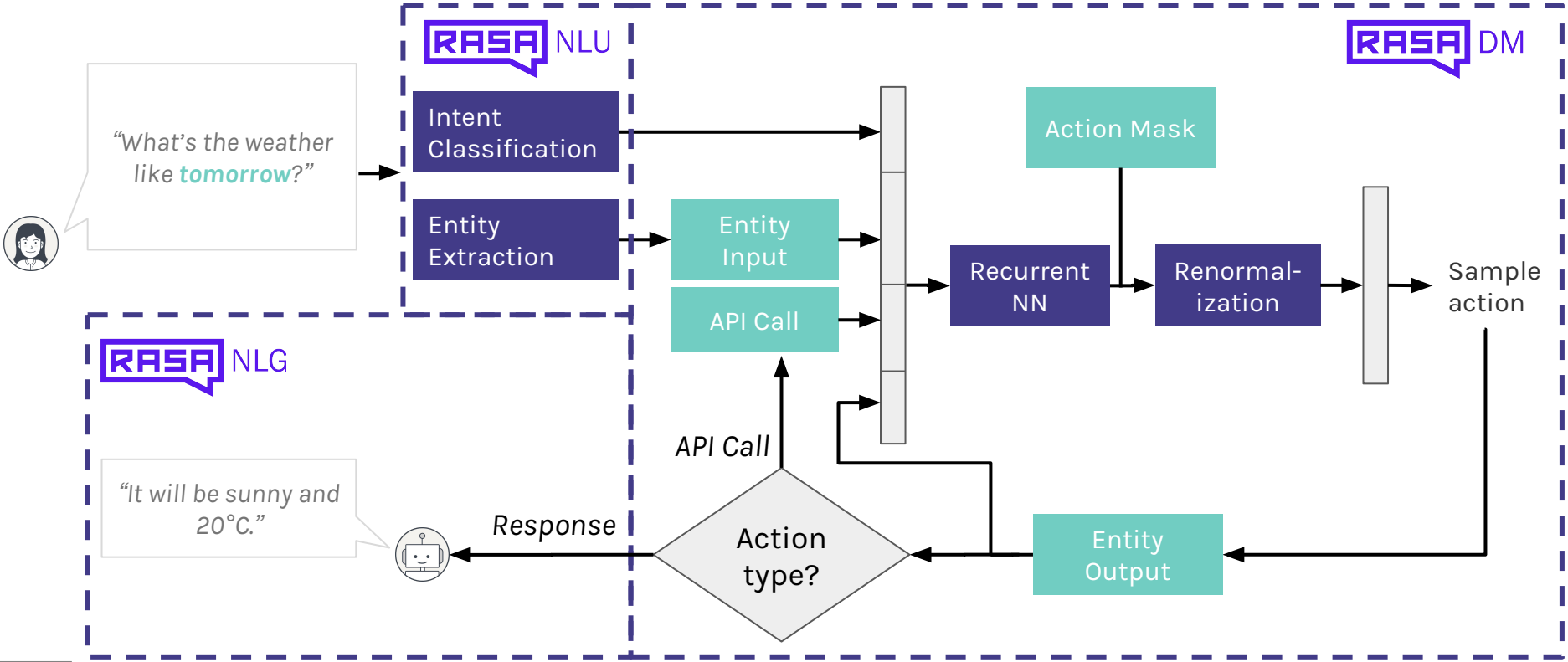
Under The Hood

Dialogue Handling



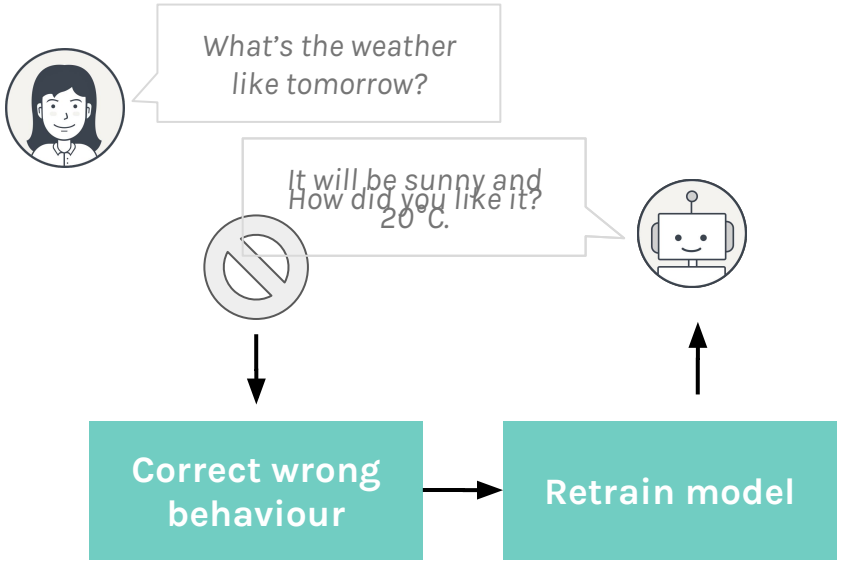
Detailed Dialogue Handling

Similar to LSTM-dialogue prediction paper: https://arxiv.org/abs/1606.01269



Dialogue Training

Issue: How to get started? → Online Learning



Demo

Under The Hood

Dialogue Training

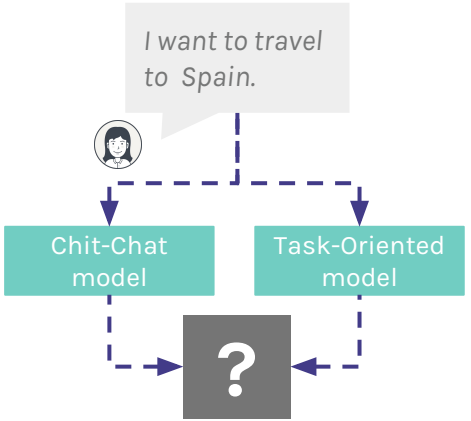
```
        user said:      _inform[price=expensive]


                whose intent is:       inform

        with price:     expensive

we currently have slots: info: None, cuisine: spanish, people: six, matches: None,

------
what is the next action for the bot?

         0                                listen    0.24
         1                               default    0.00
         2                                 greet    0.00
         3                               goodbye    0.00
         4                          ack_dosearch    0.13
         5                        ask_howcanhelp    0.05
         6                          ask_location    0.00
         7                         ask_numpeople    0.31
         8                             ask_price    0.00
         9                          ask_cuisine    0.05
        10                 ack_findalternatives    0.00
        11                 ack_makereservation    0.00
        12                     ask_moreupdates    0.00
        13                       ask_helpmore    0.00
        14                              on_it    0.04
        15                search_restaurants    0.13
        16                           suggest    0.01
```

Final Thoughts

Open Challenges

Challenges for curious minds:

- Combination of different dialogue models



- Unsupervised multi-language entity recognition

- Dialogue generalisation (e.g. optional questions)

Current Research

Good reads for a rainy day:

- Last Words: Computational Linguistics and Deep Learning (blog)
  https://goo.gl/IGSRuj
- Memory Networks (paper)
  https://arxiv.org/pdf/1410.3916
- End-to-End dialogue system using RNN (paper)
  https://arxiv.org/pdf/1604.04562.pdf
- MemN2N in python (github)
  https://github.com/vinhkhuc/MemN2N-babi-python

Summary

3 take home thoughts:

- Techniques to handle small data sets are key to get started with conversational AI
- Deep ML techniques help advance state of the art NLU and conversational AI
- Abandonen flow charts → Data driven dialogue analysis is the future!

Get in touch!

Tom Bocklisch
Lead Engineer
tom@rasa.ai

Introduction

We work on the core technology for next-generation conversational AI

is a **technology company** developing **conversational AI**.

Goal: next-generation intelligent bots

Team: tight-knit, fast-moving team of researchers, engineers, designers and product people

Location: everywhere (honestly: Berlin, Edinburgh, Beijing)

Founders:
Dr. Alan Nichol (CTO)
Alexander Weidauer (CEO)

UNIVERSITY OF CAMBRIDGE

Advisory Board:
Chad Fowler (MD & CTO @ Wunderlist)
Matthaus Krzykowski (former Co-Founder @ Xyo)
Cat Noone (Designer & Founder @ Iris)

Investors:

techstars

Reference customers:

RAIFFEISEN
ERGO
SwissLife