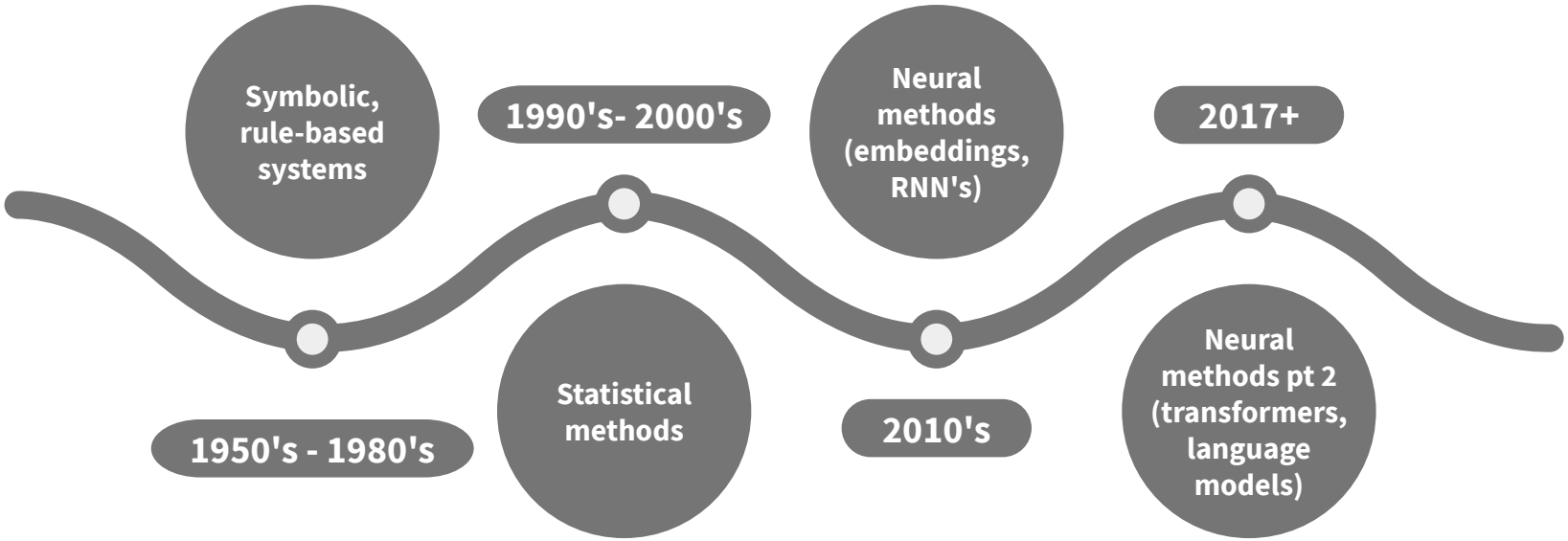
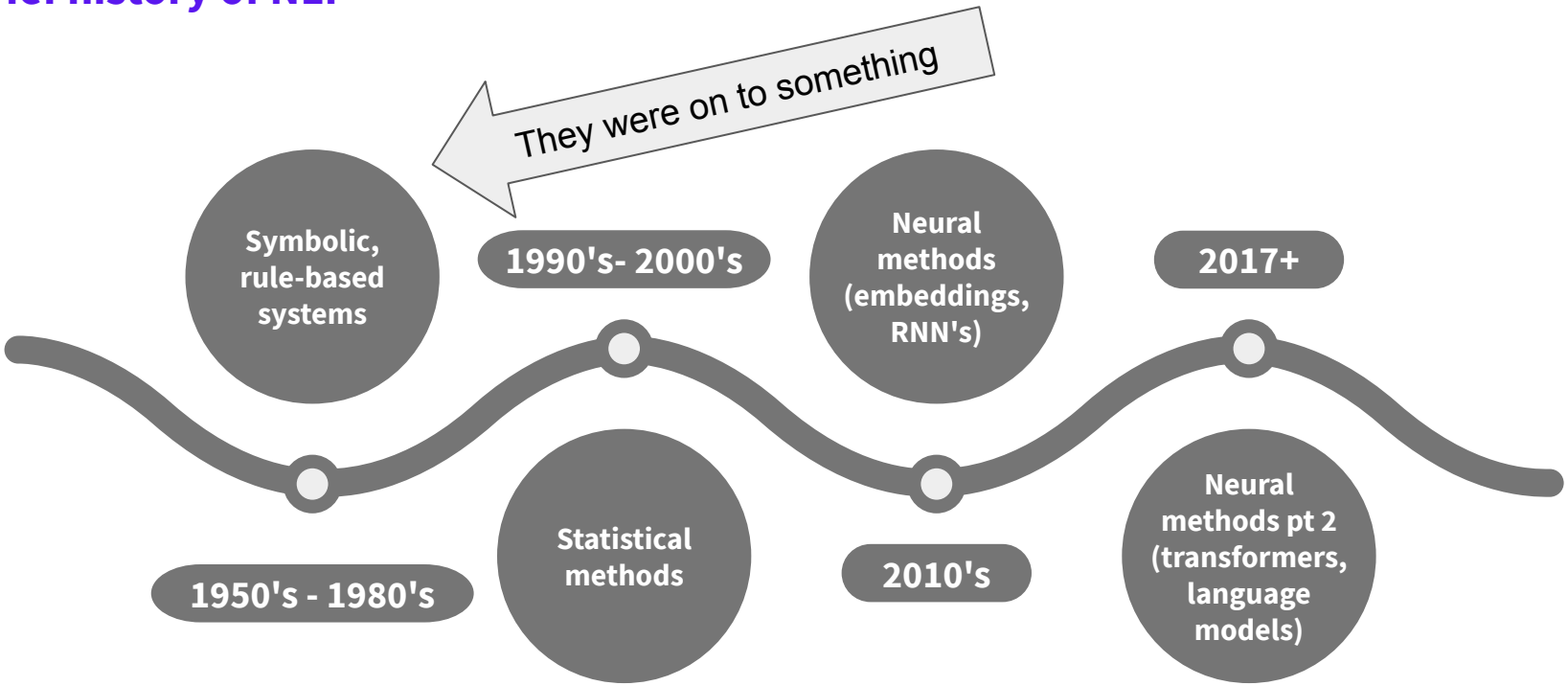
Sure, transformers are cool... but have you tried rules?

Dr. Rachael Tatman
Senior Developer Advocate, Rasa

A brief history of NLP



A brief history of NLP



Rules vs. Neural Methods

Neural methods are:

- Flexible
- Good at handling unseen data
- Probabilistic

But also:

- Unpredictable (wouldn't recommend them for generating text to serve to users)
- Require a lot of training data

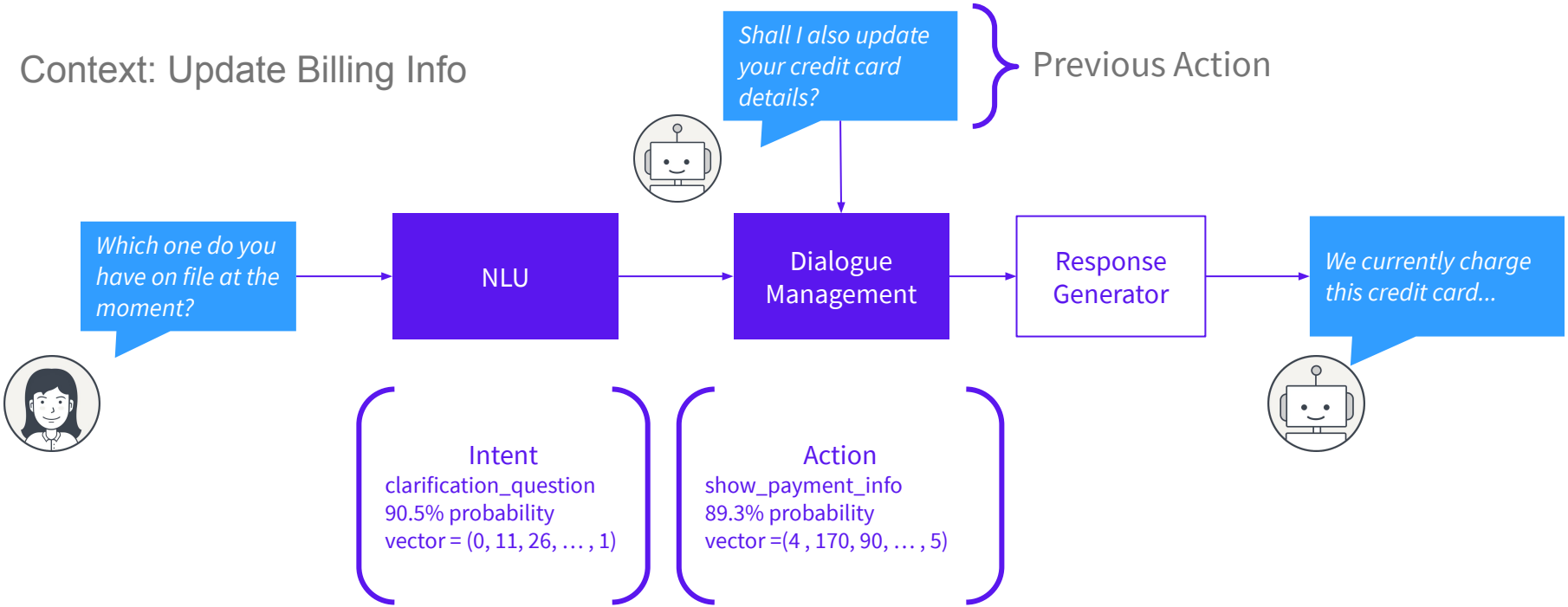
Rule based methods are:

- Reliable
- Easy to interpret
- Require no training data
- Very predictable

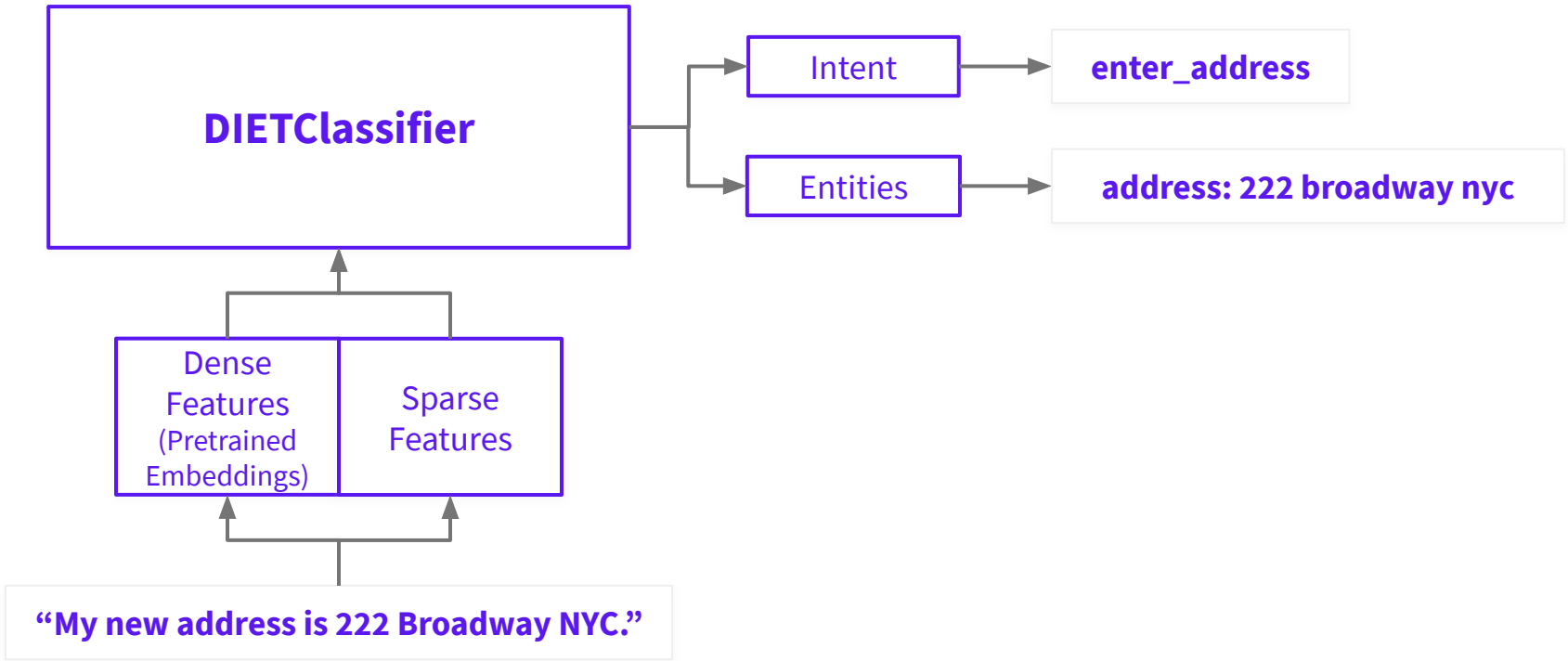
But also:

- Very narrowly defined
- Can become unwieldy
- Don't adapt to unseen situations

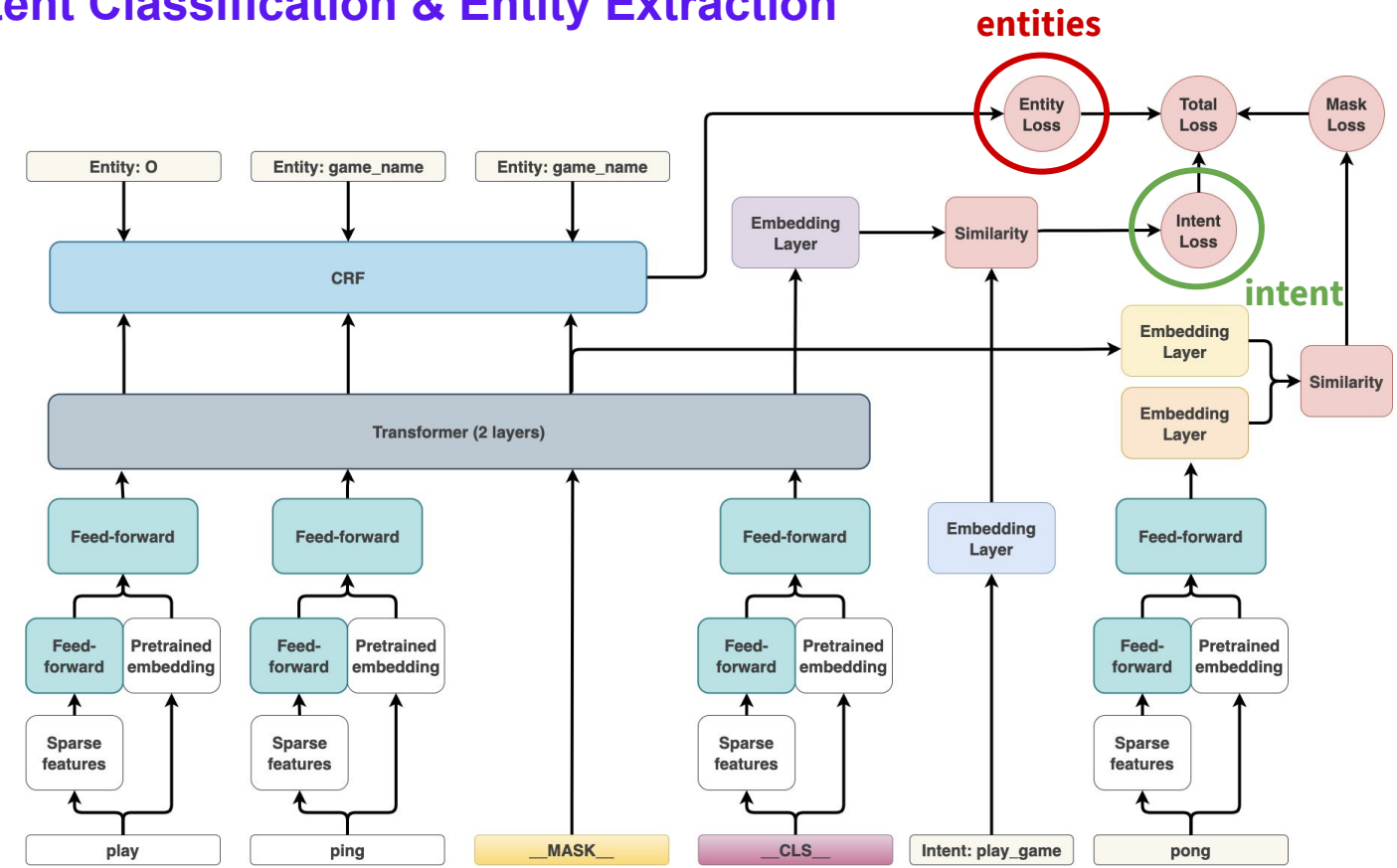
The Importance of Context



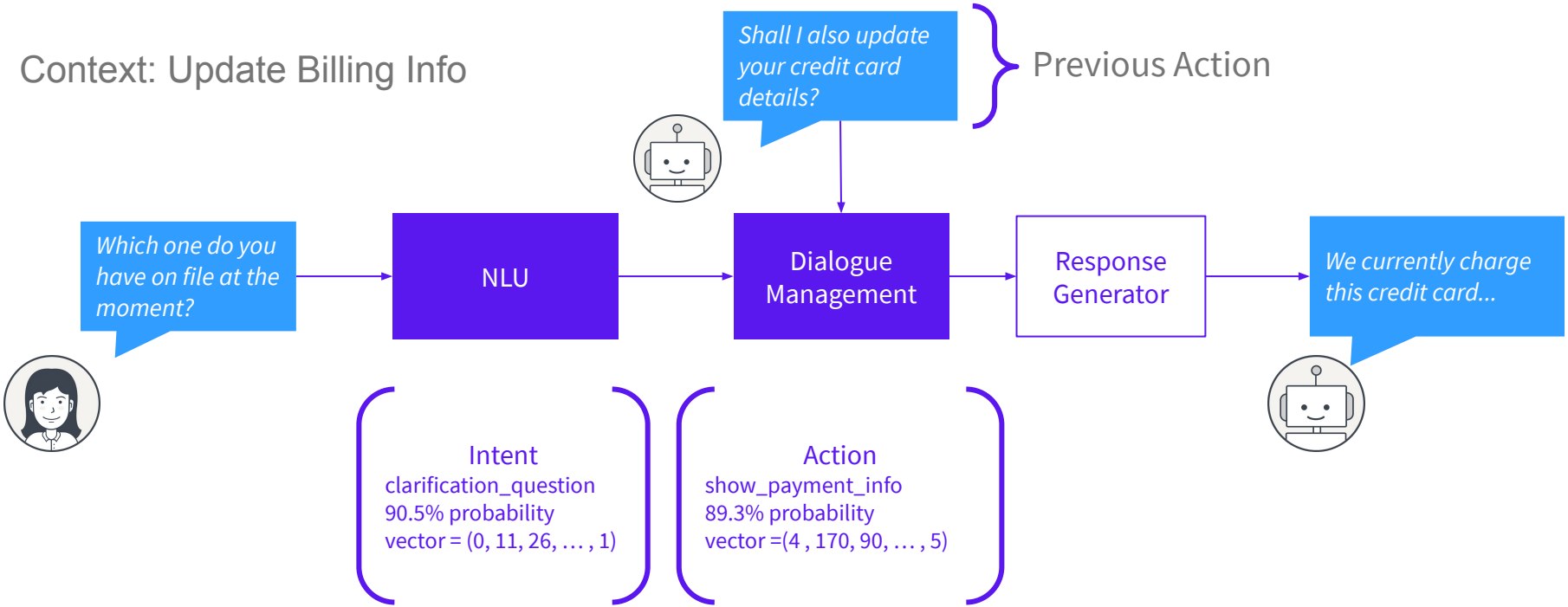
DIETClassifier: Combined Intent Classification & Entity Extraction



DIET: Intent Classification & Entity Extraction



The Importance of Context



Machine Learning Based Policies

These policies should be used in conjunction with rule-based policies

- **KerasPolicy:** Uses a standard LSTM to predict the next action
  - Learns the patterns of your stories
  - Good for handling stories that don't exactly match your training data
- **TED Policy:** Uses Attention to Handle Uncooperative Dialogue
  - Requires fewer story examples of uncooperative user dialogue
    - e.g. users who go off on tangents instead of providing the requested information
  - Effectively "ignores" irrelevant parts of the dialogue

Machine Learning Based Policies

These policies should be used in conjunction with **rule-based policies**

- **KerasPolicy:** Uses a standard LSTM to predict the next action
  - Learns the patterns of your stories
  - Good for handling stories that don't exactly match your training data
- **TED Policy:** Uses Attention to Handle Uncooperative Dialogue
  - Requires fewer story examples of uncooperative user dialogue
    - e.g. users who go off on tangents instead of providing the requested information
  - Effectively "ignores" irrelevant parts of the dialogue

Rules

Rules are used to train the `RulePolicy`

Rules look like short stories, but have some additional properties:

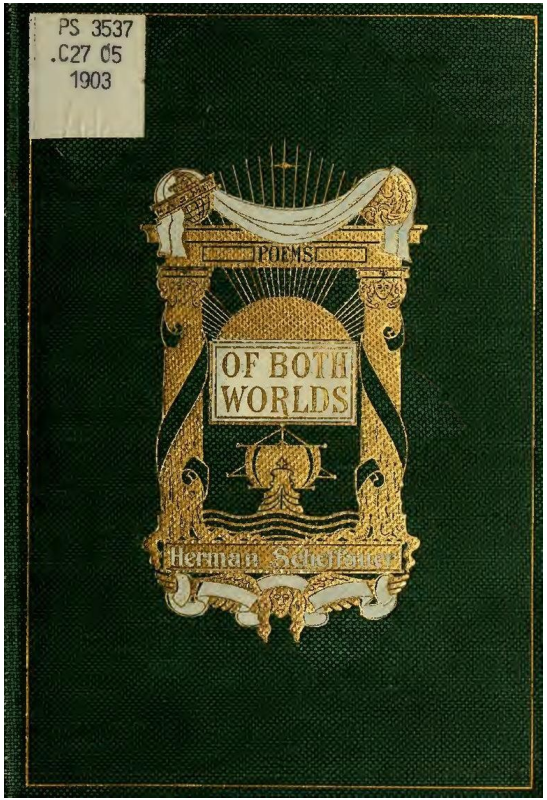
- `conditions` that must be met for the rule to apply
- `wait_for_user_input: false` at the end of a rule prevents automatically appending `action_listen` and allows further action prediction

```
rules:
- rule: greet
  steps:
  - intent: greet
  - action: utter_greet

- rule: greet by name
  conditions:
  - slot_was_set:
    - name: "something"
  steps:
  - intent: greet
  - action: utter_greet_name

- rule: faq interruption
  steps:
  - intent: faq
  - action: utter_faq
  Wait_for_user_input: false
```

What's next for NLP?

- Both rules and neural methods have a place in Conversational AI
  - Pure neural methods are too unpredictable for high stakes applications & training data isn't always available
  - Pure rule-based systems are too exact to cover all situations, neural methods are more extensible
- (Partially) rule based systems may not be fashionable in research but they aren't going anywhere in commercial applications

Public domain meme substitute



Thanks! Questions?