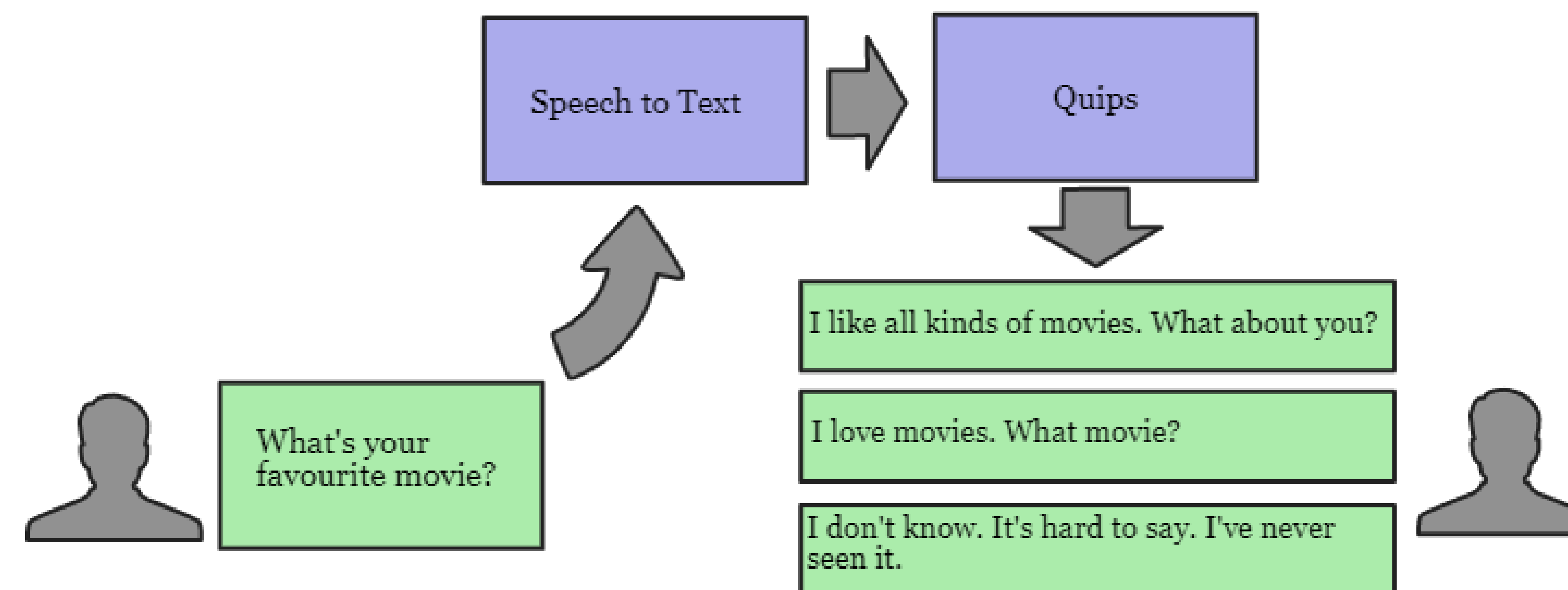


1. Quips: Problem Overview

Motor Neurone Disease (MND) is a fatal disease that causes rapid loss of muscle movement. Eventually this can lead to an almost **total loss of communication**. While assistive technologies for this exist, the long delays associated with spelling out each word causes frustration, to the point where many people begin limiting their social interactions.

In partnership with the Motor Neurone Disease Association, Roll's Royce AI hub is leading a project that aims to enhance such assistive technologies. So far the project has implemented a prototype - '**Quips**', which is based on a Sequence to Sequence architecture commonly used in Neural Machine Translation (NMT) using PyTorch.

2. Quips Prototype



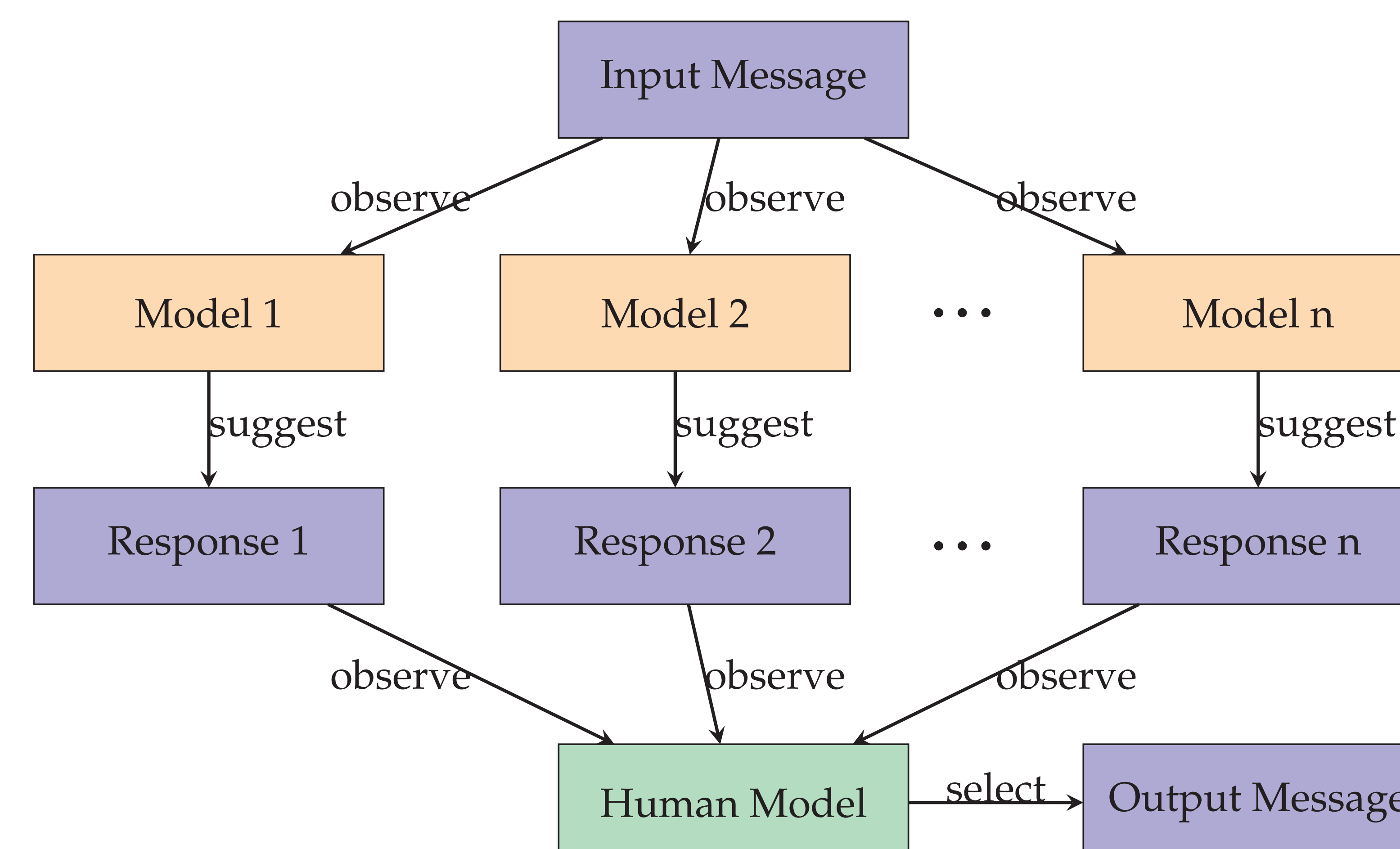
3. ImperialQuips

ImperialQuips is a prototype implemented (since May/June) in this project which is an **incremental upgrade** of Quips built by Rolls Royce. This is a joint work with MSc student Xin Li Huang.

In particular ImperialQuips enhances the architecture of Quips by making each interaction generate possible answers from multiple chatbots (or NLP models) rather than just implementing one chatbot.

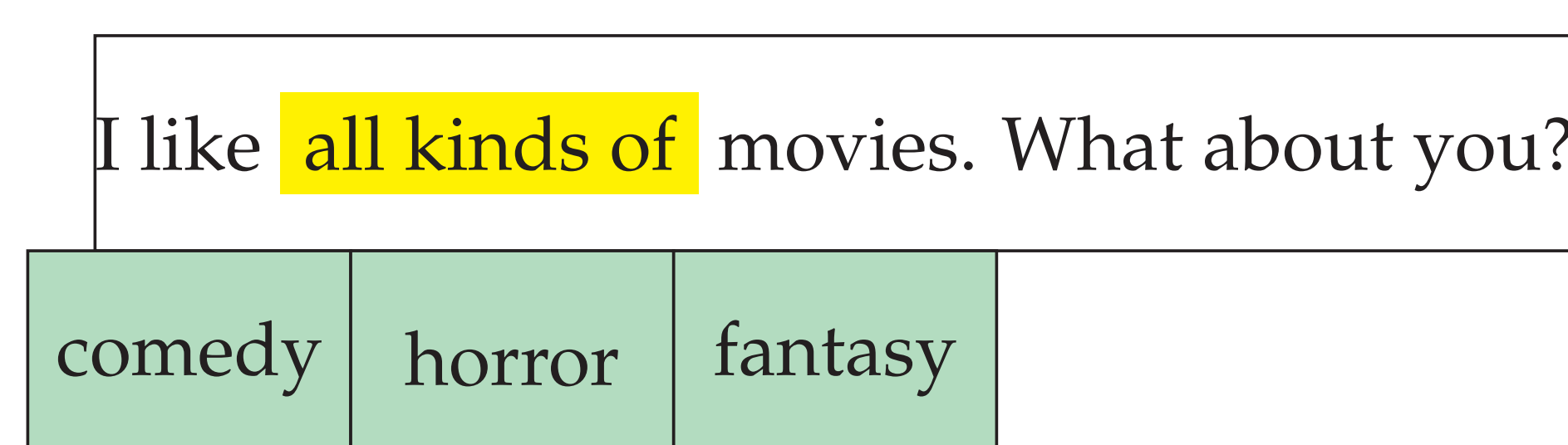
The models available in ImperialQuips include numerous dialogue systems implemented in Facebook's ParlAI framework (see Panel 7).

4. ImperialQuips Prototype



5. ImperialQuips 2.0

It is unlikely that any of the suggested responses in ImperialQuips 1.0 will exactly match what a user would like to say. So the next step is to implement a **personalisation** layer that helps the user to efficiently modify one of the suggested responses from ImperialQuips 1.0.

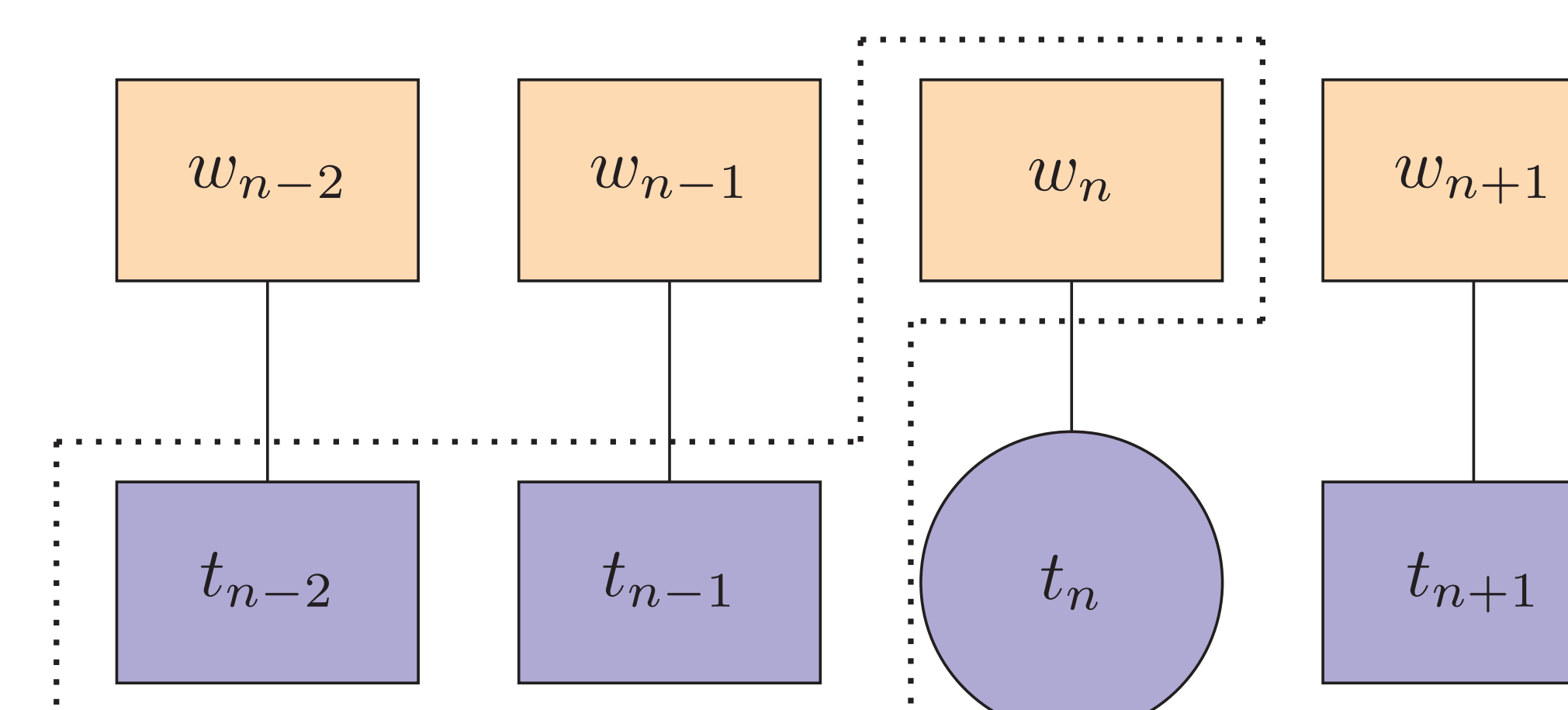


6. N-Gram Tagging

The modification of the responses can be done using **Part-of-speech (POS) tagging** - the process of classifying words (**tokens**) into their parts of speech (**tags**), such as nouns or adjectives.

An **n-gram tagger** picks the tag t_n for a token w_n that is most likely in the context of the $n - 1$ preceding tags. Thus we are looking for t_n that maximizes

$$P(t_n | t_1, \dots, t_{n-1}).$$



Example of n-gram tagger with n=3

7. ParlAI

ParlAI is a python framework for sharing, training and testing dialogue models developed by Facebook. It contains:

- 80+ popular datasets
- a set of reference models
- a zoo of pretrained models

Amongst those **pretrained models** there is: **Wizard Of Wikipedia (WoW)**

Dataset: Conversations discussing a randomly chosen topic using passages from Wikipedia articles

Theoretical Novelty: Generative transformer memory network selects the most relevant piece of knowledge and performs an encoding step by concatenating it with the dialogue context

Overview: This model usually offers accurate responses to factual questions.

Convai

Dataset: Conversations between paired crowd-workers getting to know each other, where each is given a role to play based on their persona

Theoretical Novelty: Generative Profile Memory Network stores profile of persona

Overview: This model usually shows a consistent personality.

Empathetic Dialogues (ED)

Dataset: Conversations grounded in an emotional situation related to a given feeling

Theoretical Novelty: Training over this dataset can improve the performance of an end-to-end dialogue system on empathetic dialogue

Overview: This model is usually able to display empathy.

Blended Skill Talk

Dataset: Instances from the three previous datasets

Theoretical Novelty: Existing Retrieval and Generator models are fine-tuned

Overview: This model aims to combine the skills from the models above seamlessly during dialogue.

References

- [1] Facebook AI Research. ParlAI Documentation. <https://parl.ai/docs/index.html>, 2020.
- [2] Roler, Dinan, Goyal, Ju, Williamson, Liu, Xu, Ott, Shuster, Smith, Boureau, Weston. Recipe for building an open-domain chatbot. Facebook AI Research, arXiv:2004.13637, 2020.