Dynamic Coattention Networks For Question Answering

Caiming Xiong, Victor Zhong, Richard Socher

Houman Mehrafarin
Mahsa Razavi

Introduction

- Many models have been proposed for Question Answering, however due to their single pass nature they cannot recover from local maxima corresponding to incorrect answers

- DCN, an end-to-end neural network for question answering solves this problem

Why is DCN useful?

- The DCN first fuses co-dependent representations of the question and the document in order to focus on relevant parts of both.
- Afterwards a dynamic pointing decoder iterates over potential answer spans.
- This iterative procedure allows us to recover from the local maxima

Related Work

- Dynamic Chunk Reader

  It uses a similar attention mechanism (calculating attention values for each pair of words in the question and context paragraph) but aggressively filtering possible answers by selecting an answer from candidate "chunks". The candidate chunks are determined by linguistic parsing of the roles that the various words play in a sentence.

- Seq2Seq Model

Our Baseline

- Our baseline is a simple Seq2Seq model consisting of 2 bi-directional LSTMs. One for questions and another for the documents

The model’s recipe

- Document and Question Encoder
- Co-Attention Encoder
- Dynamic Pointing Decoder

Question and Document Encoder

- This step is actually really simple:

  We just take the documents and questions as inputs, pass them to a pre-trained embedding layer (with GloVe pre-trained weights) and encode them with a shared LSTM

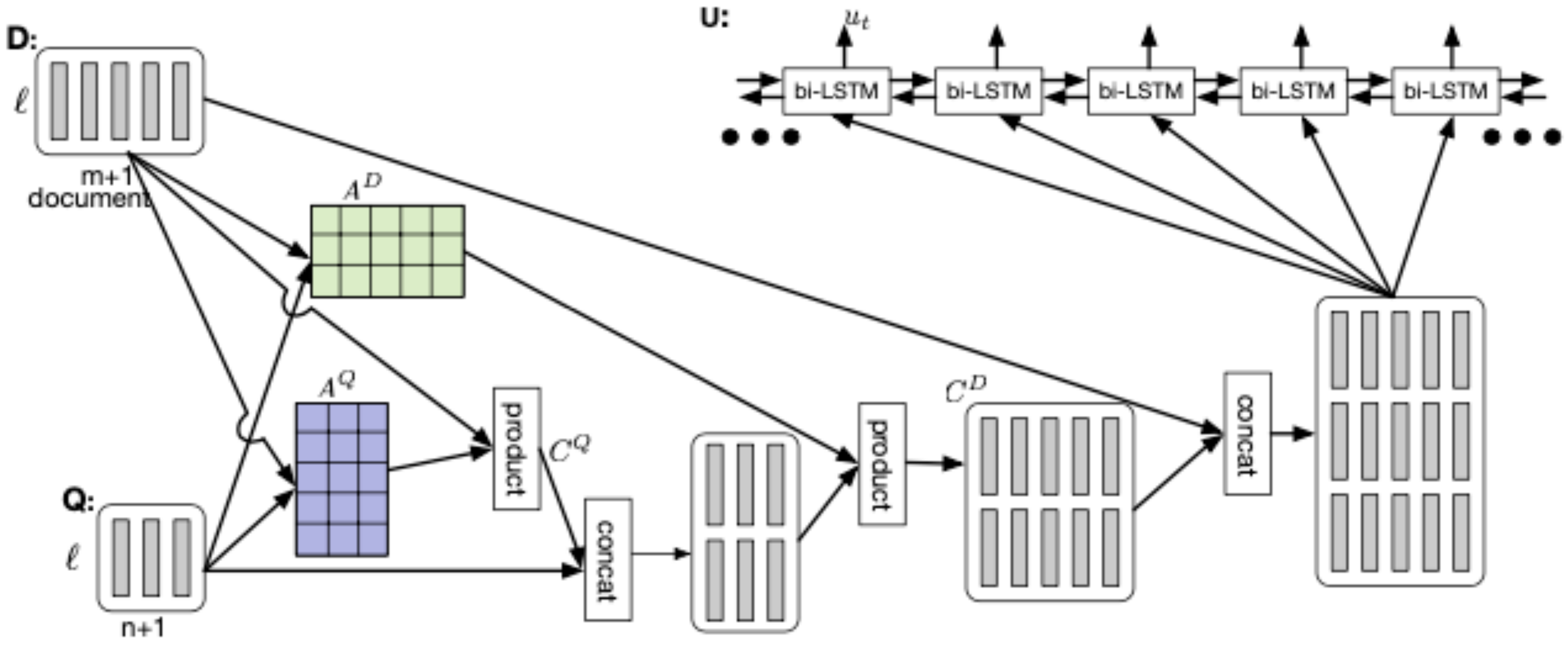
- What are the outputs of this layer?

  The outputs are the document states and question states retrieved from the LSTM layer and the output of a dense layer which we passed the question states to

Co-Attention Encoder

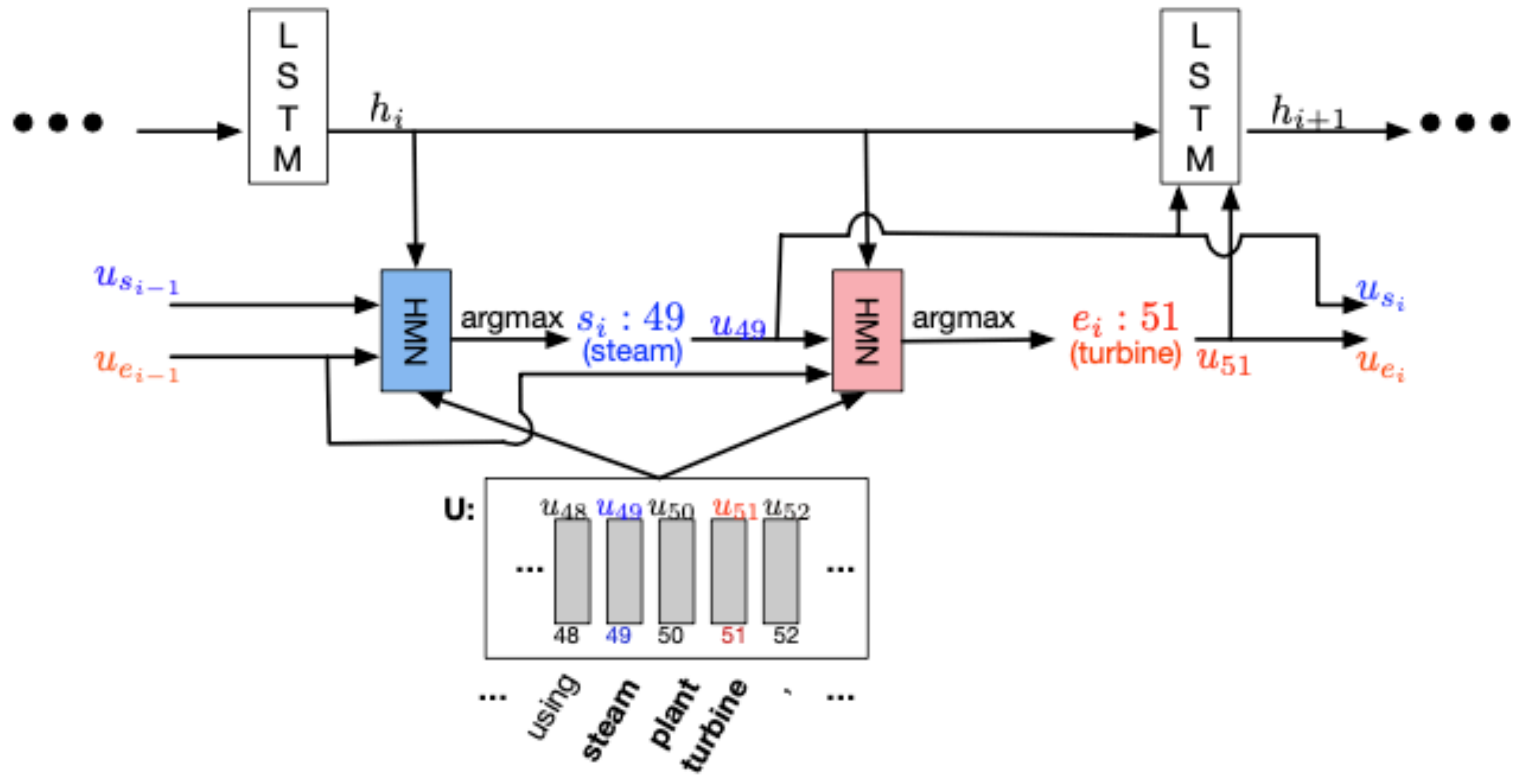
- The beating heart of the model is the Co-Attention Encoder
- Here we do all the fun stuff!

By that I mean lots of concatenation. We explained it all in the documentation



Dynamic Pointing Decoder

- What we actually expect as outputs for model is indexes of the beginning and the end of the answer span.
- In this layer, during each iteration, the decoder update its current state taking into account the Coattention encoding corresponding to current estimates of the start and end positions and produces the new estimates



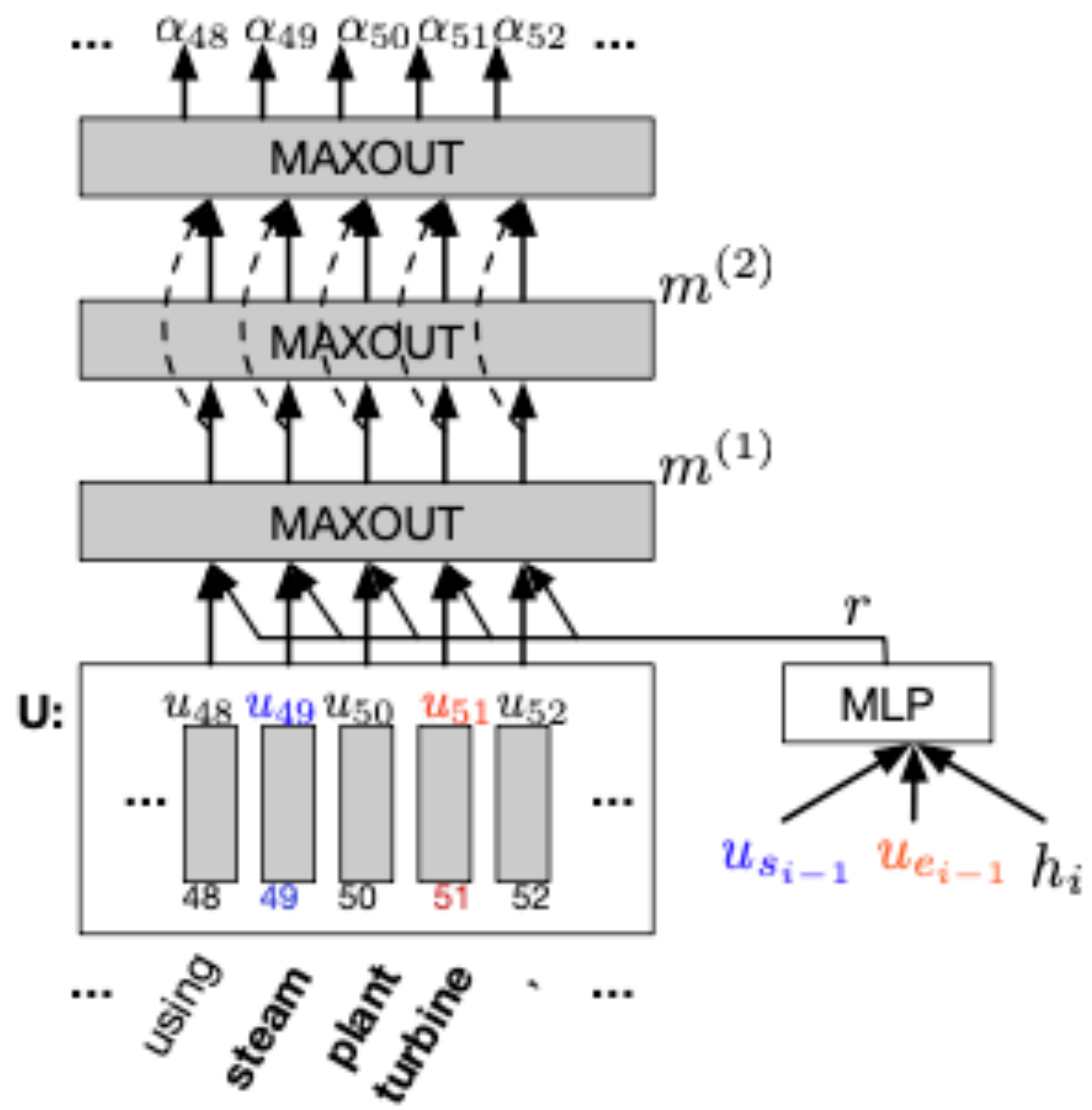
Highway Maxout Network

- Highway Maxout Network (HMN) is made up of two networks. Highway Networks and Maxout Networks
- The intuition behind using such models is the nature of QA tasks, which have multiple question types and document topics.
- These variations may require different models to estimate the answer span.
- Maxout provides a simple and efficient way to pool across different multiple model variations

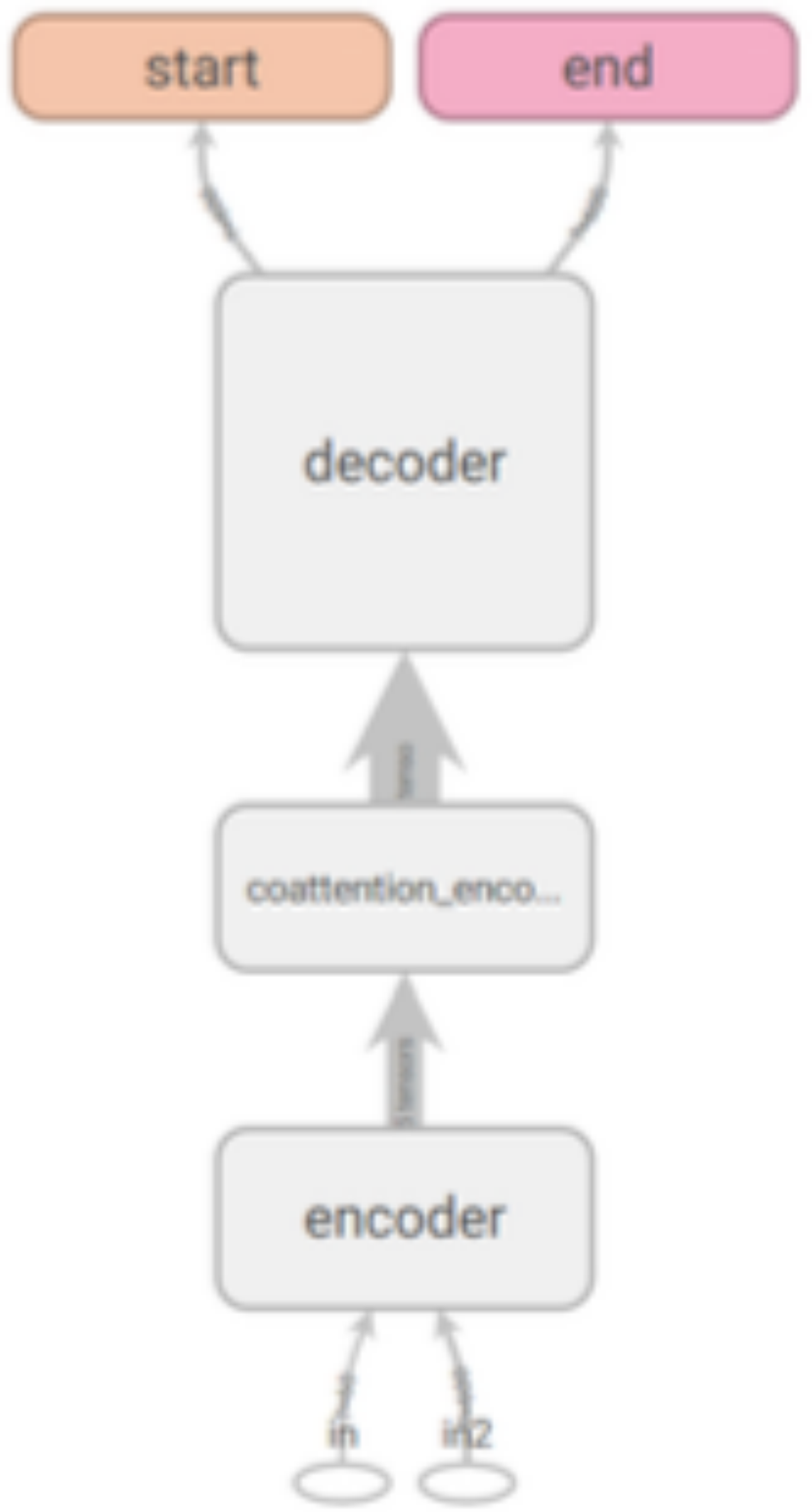
Where is HMN used?

- HMN is used inside the decoder layer. It takes ending and starting encodings, GRU cell state and Coattention layer's output.
- HMN actually consists 4 layers.
- We use to different HMNs for predicting the starting position and the ending position (α and β)

Highway Maxout Network Architecture



The final model



Thank You!