Simple and Effective Multi-Paragraph Reading Comprehension

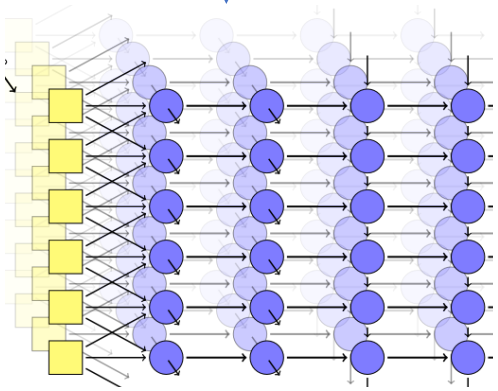
Christopher Clark and Matt Gardner

ALLEN INSTITUTE for ARTIFICIAL INTELLIGENCE

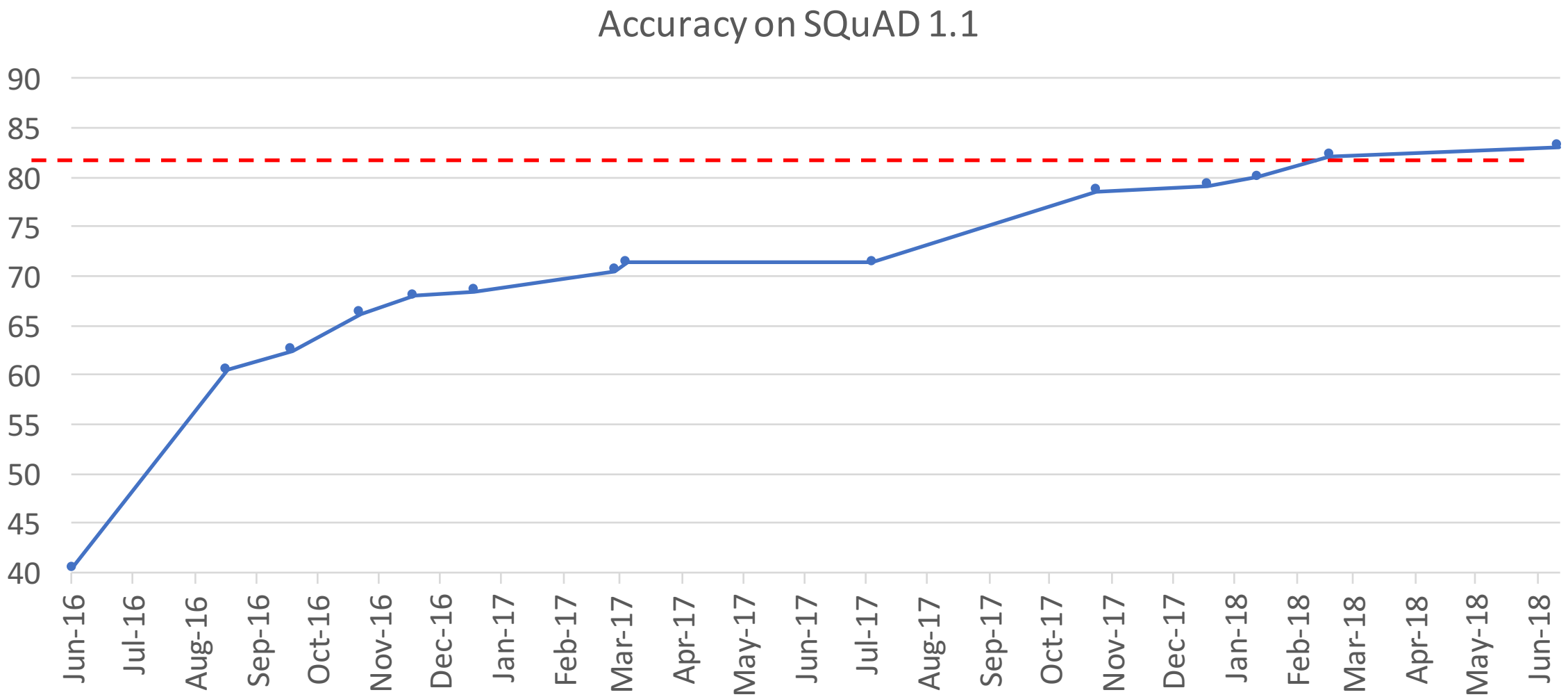
Neural Question Answering

Question: “What color is the sky?”

Passage: “Air is made mainly from molecules of nitrogen and oxygen. These molecules scatter the blue colors of sunlight more effectively than the green and red colors. Therefore, a clean sky appears blue.”



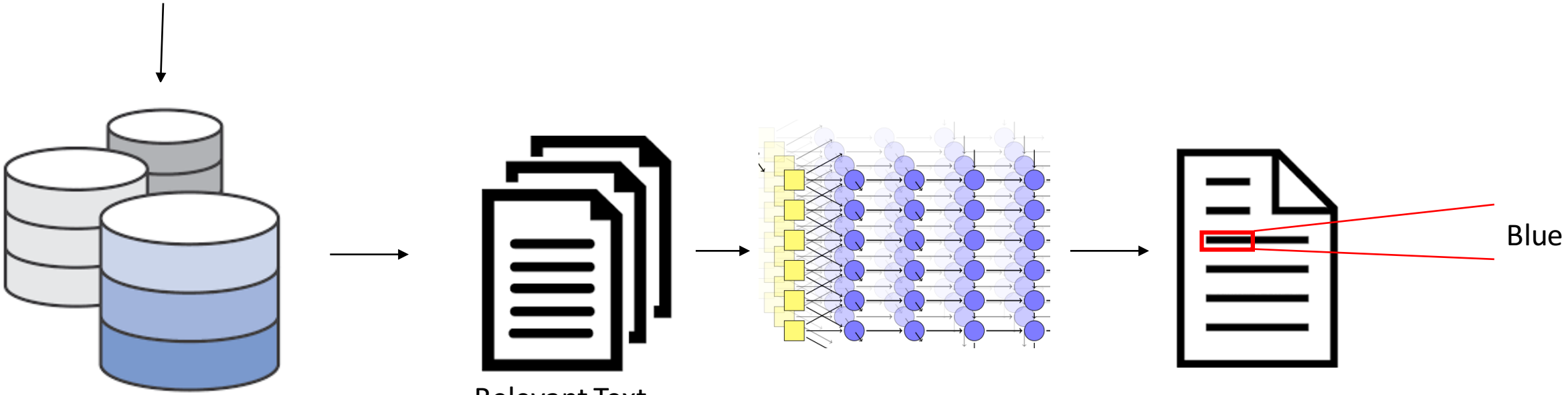
Fast Progress on Paragraph Datasets



What Next?

Open Question Answering

Question: “What color is the sky?”

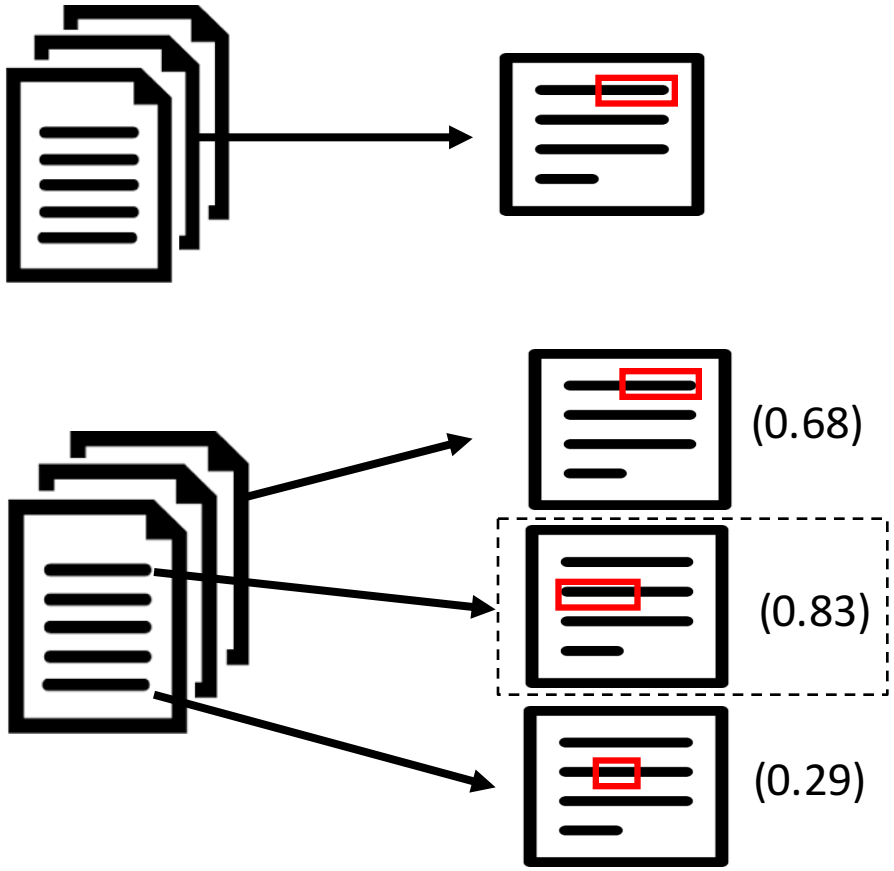


Challenge: Scaling Models to Documents

- Modern reading comprehension models have many layers and parameters
  - The trend is continuing in this direction, for example with the use of large language models
- Reduced efficiency as the paragraph length increases due to long RNN chains or transformers/self-attention modules
- Limits the model to processing short paragraphs

Two Possible Approaches

- Pipelined Systems
  - Select a single paragraph from the input, and run the model on that paragraph
- Confidence Systems
  - Run the model on many paragraphs from the input, and have itassign a confidence score to its results on each paragraph



This Work

Improved Pipeline Method

- Improve several of the key design decision that arise when training on document-level data

Improved Confidence Method

- Study ways to train models to produce correct confidence scores

Pipeline Method: Paragraph Selection

- Train a shallow linear model to select the best paragraphs
  - Features include TF-IDF, word occurrences, and its position within the document
- If there is just one document TF-IDF alone is effective
- Improves change of selecting an answering-containing paragraph from 83.0 to 85.1 on TriviaQA Web

Pipeline Method: Noisy Supervision

Document level data can be expected to be distantly supervised:

Question: Which British general was killed at Khartoum in 1885?

Passage:

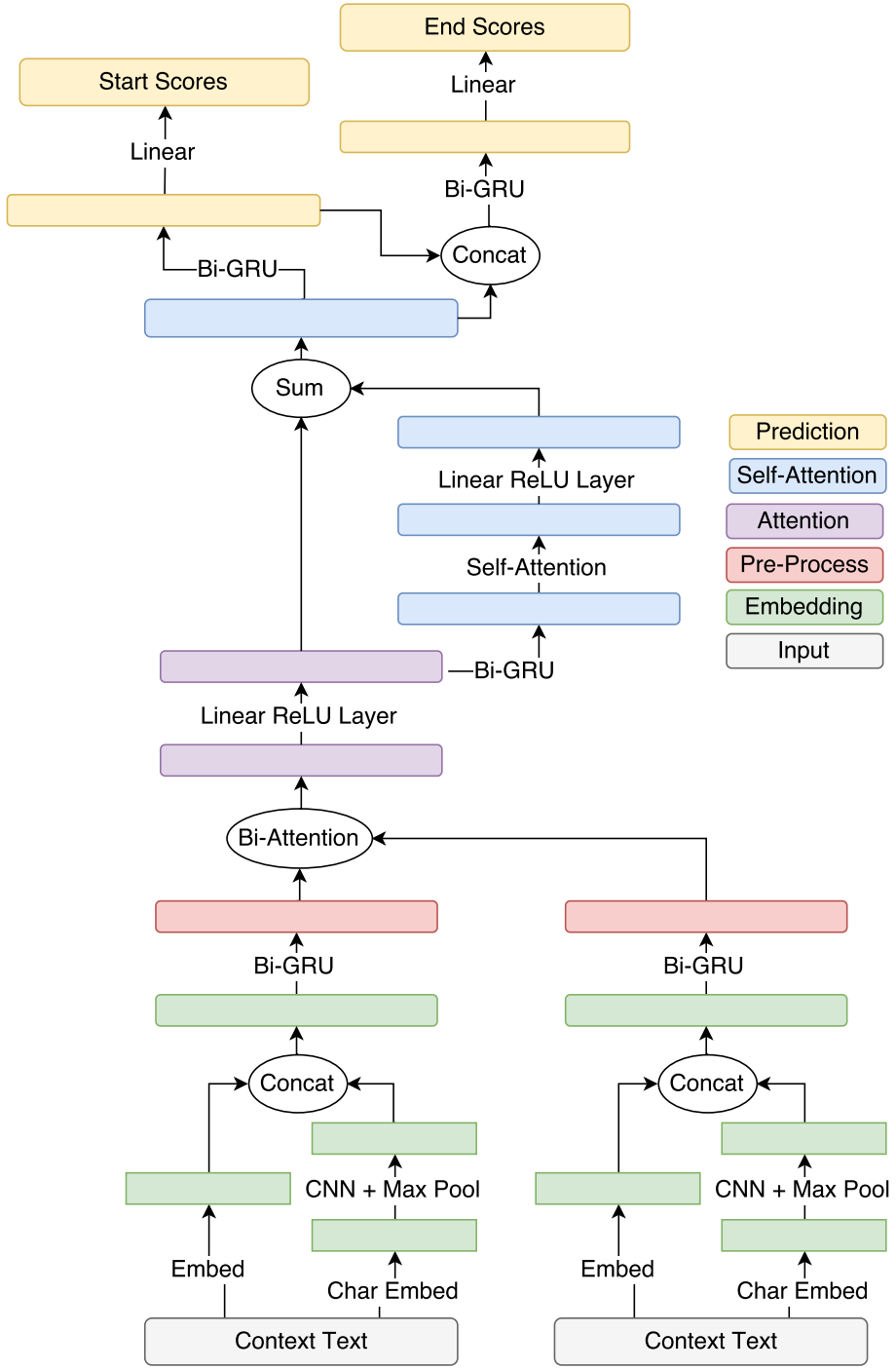
In February 1884 Gordon returned to the Sudan to evacuate Egyptian forces. Rebels broke into the city , killing Gordon and the other defenders. The British public reacted to his death by acclaiming ' Gordon of Khartoum , a saint. However, historians have since suggested that Gordon defied orders and….

Pipeline Method: Noisy Supervision

- Need a training objective that can handle multiple (noisy) answer spans
- Use the summed objective from Kadlec et al (2016), that optimizes the log sum of the probability of all answer spans
- Remains agnostic to how probability mass is distributed among the answer spans

Pipeline Method: Model

- Construct a fast, competitive model
- Use some keys ideas from prior work, bidirectional-attention, self-attention, character-embeddings, variational dropout
- Also added learned tokens for document and paragraphs starts
- < 5 hours to train for 26 epochs on SQuAD



Confidence Methods

- We can derive confidence scores from the logit scores given to each span by the model, i.e., the scores given before the softmax operator is applied
- Without re-training this can work poorly

Example from SQuAD

Question: “When is the Members Debate held?”

Model Extraction: “..majority of the Scottish electorate voted for it in a referendum to be held on **1 March 1979** that represented at least... ”

Correct Answer: **“Immediately after Decision Time** a “Members Debate” is held, which lasts for 45 minutes... ”

Learning Well-Calibrated Confidence Scores

- Train the model on both answering-containing and non-answering containing paragraph and use a modified objective function

- **Merge**: Concatenate sampled paragraphs together
- **No-Answer**: Process paragraphs independently, and allow the model to place probability mass on a "no-answer" output
- **Sigmoid**: Assign an independent probability on each span using the sigmoid operator
- **Shared-Norm**: Process paragraphs independently, but compute the span probability across spans in all paragraphs

Results

Datasets

- **TriviaQA**: Datasets of trivia questions and related documents found by web-search
  - Includes three setting, Web (a single document for each questions) Wiki (multiple wikipedia documents for each questions) and Unfiltered (Multiple documents for each questions)

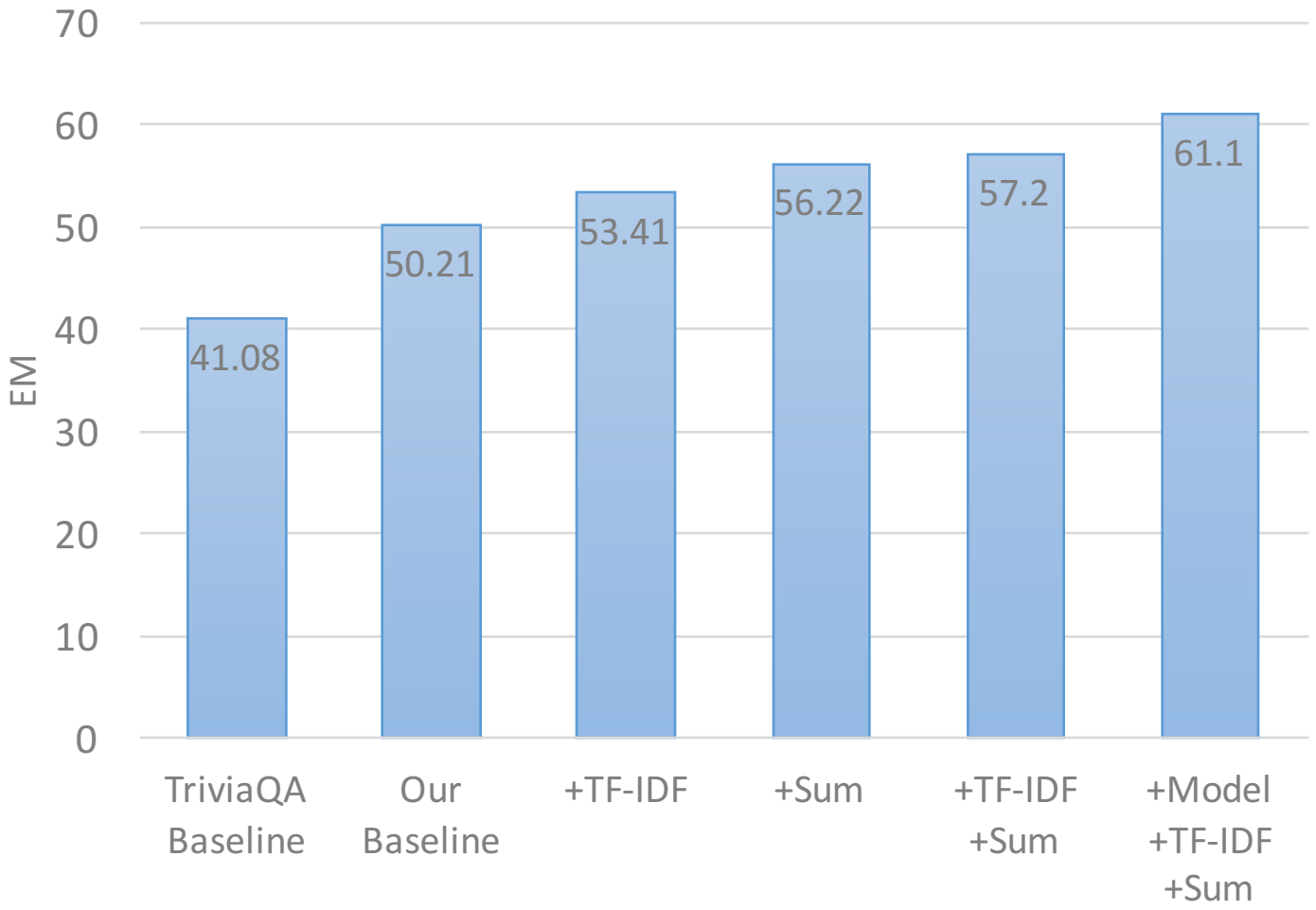
- **SQuAD:** Turker-generated questions about Wikipedia articles
  - We use the questions paired with the entire article
  - Manual annotation shows most (90%) of questions are answerable as given the document it was generated from

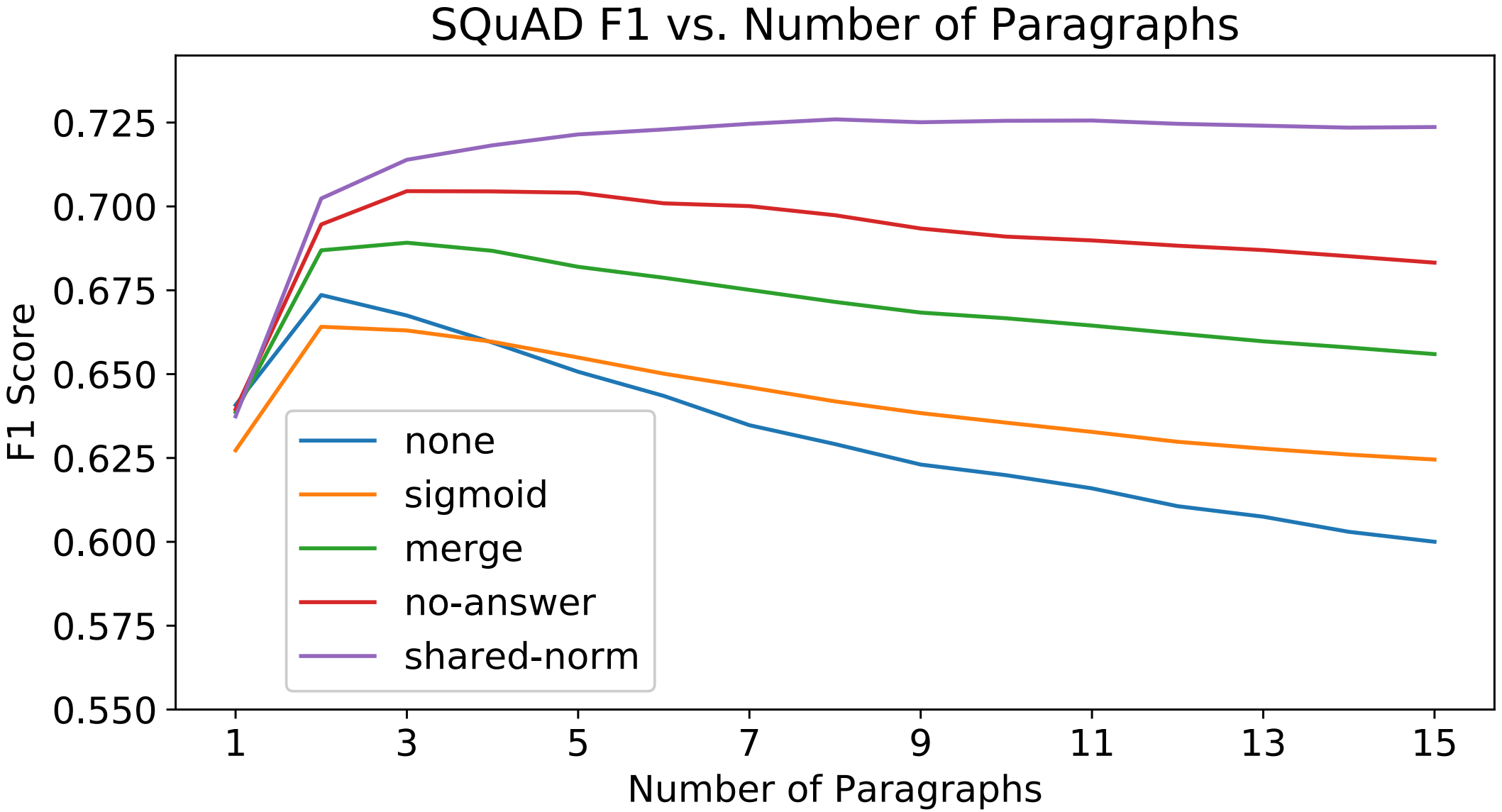
Pipeline Method: Results on TriviaQA Web

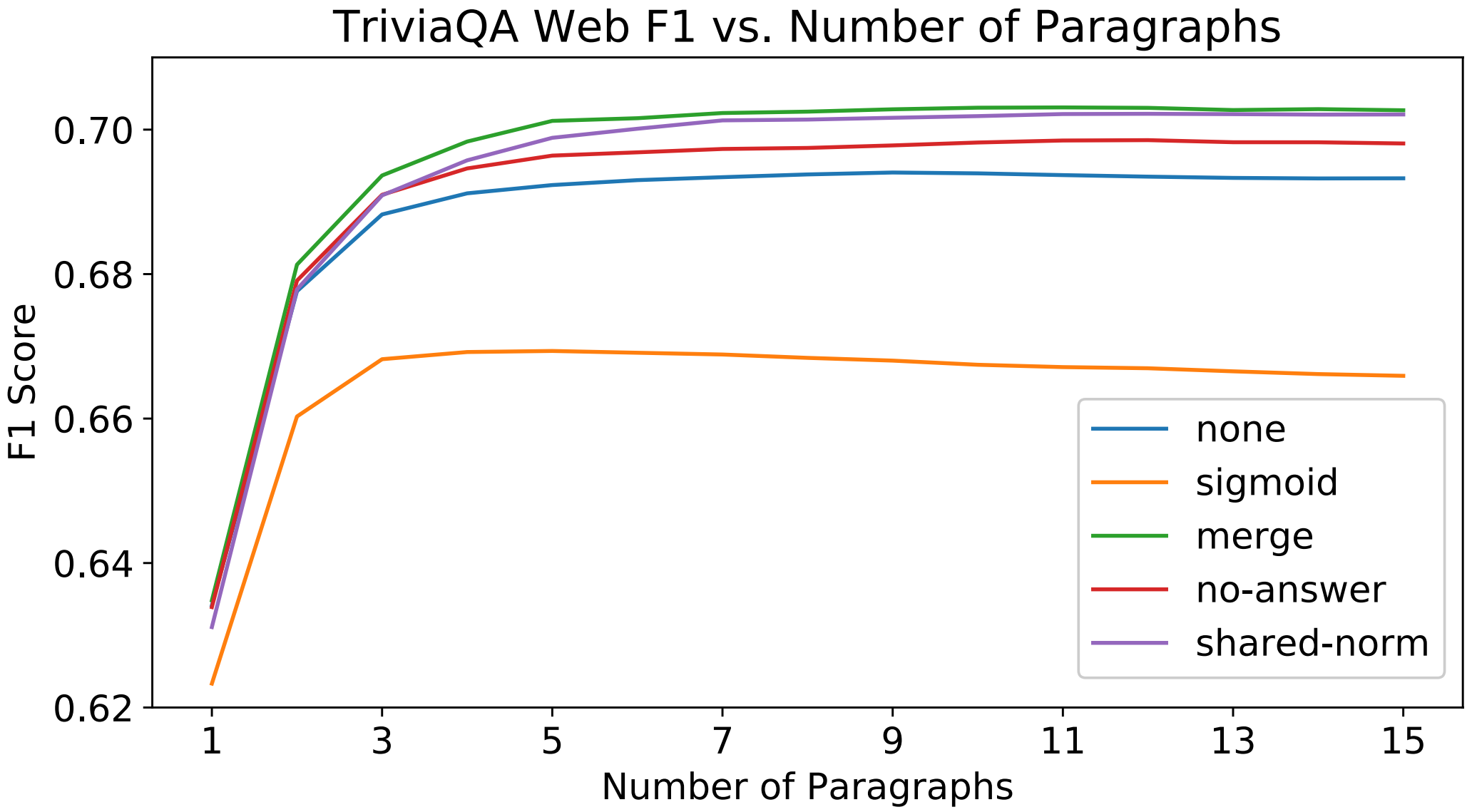
Baseline implementation:

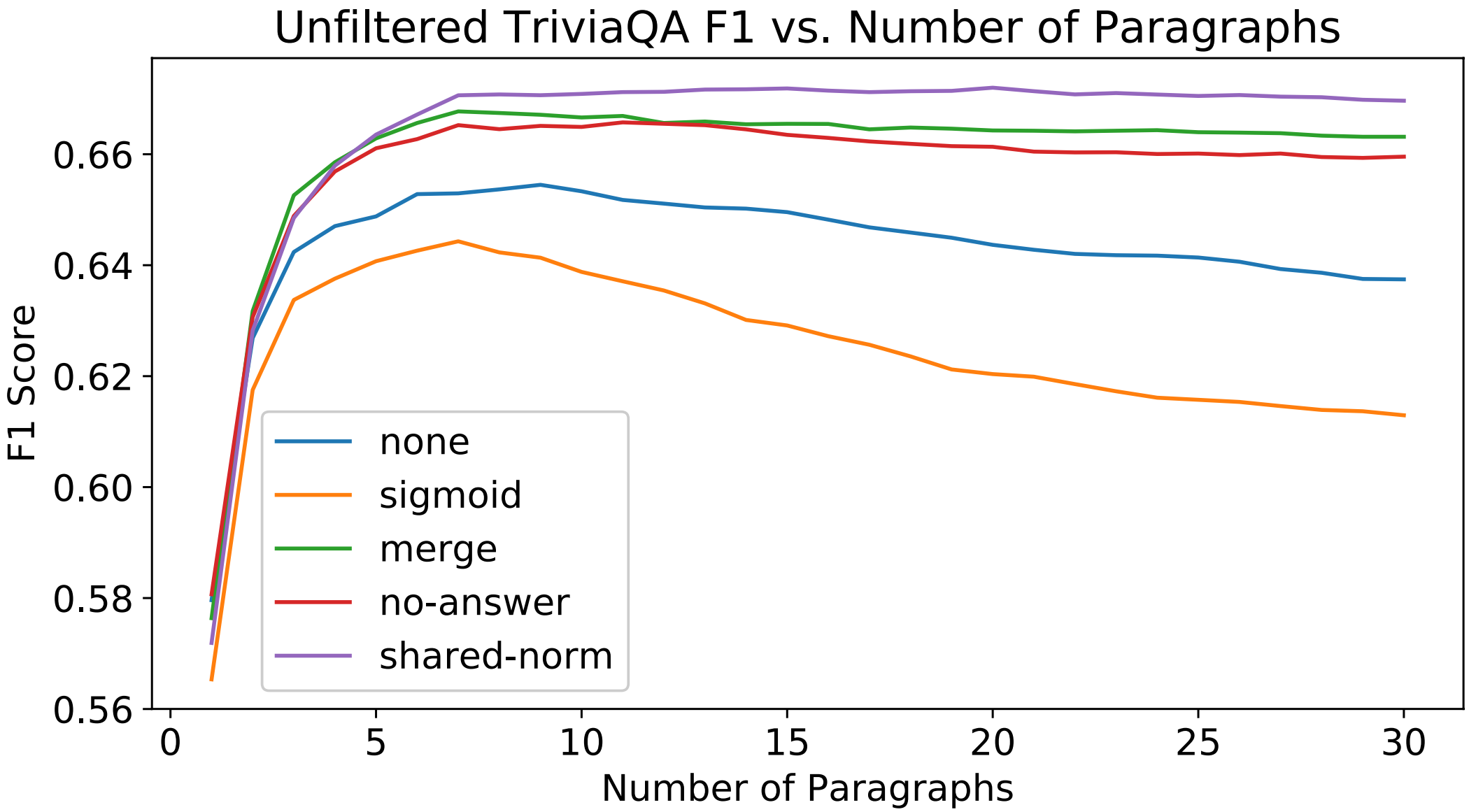
- Uses BiDAF as the model
- Select paragraphs by truncating documents
- Select answer-spans randomly

- 72.14 EM / 81.05 F1 on SQuAD
- 78.58 EM / 85.83 F1 with contextualized word embeddings (Peters et al., 2017)







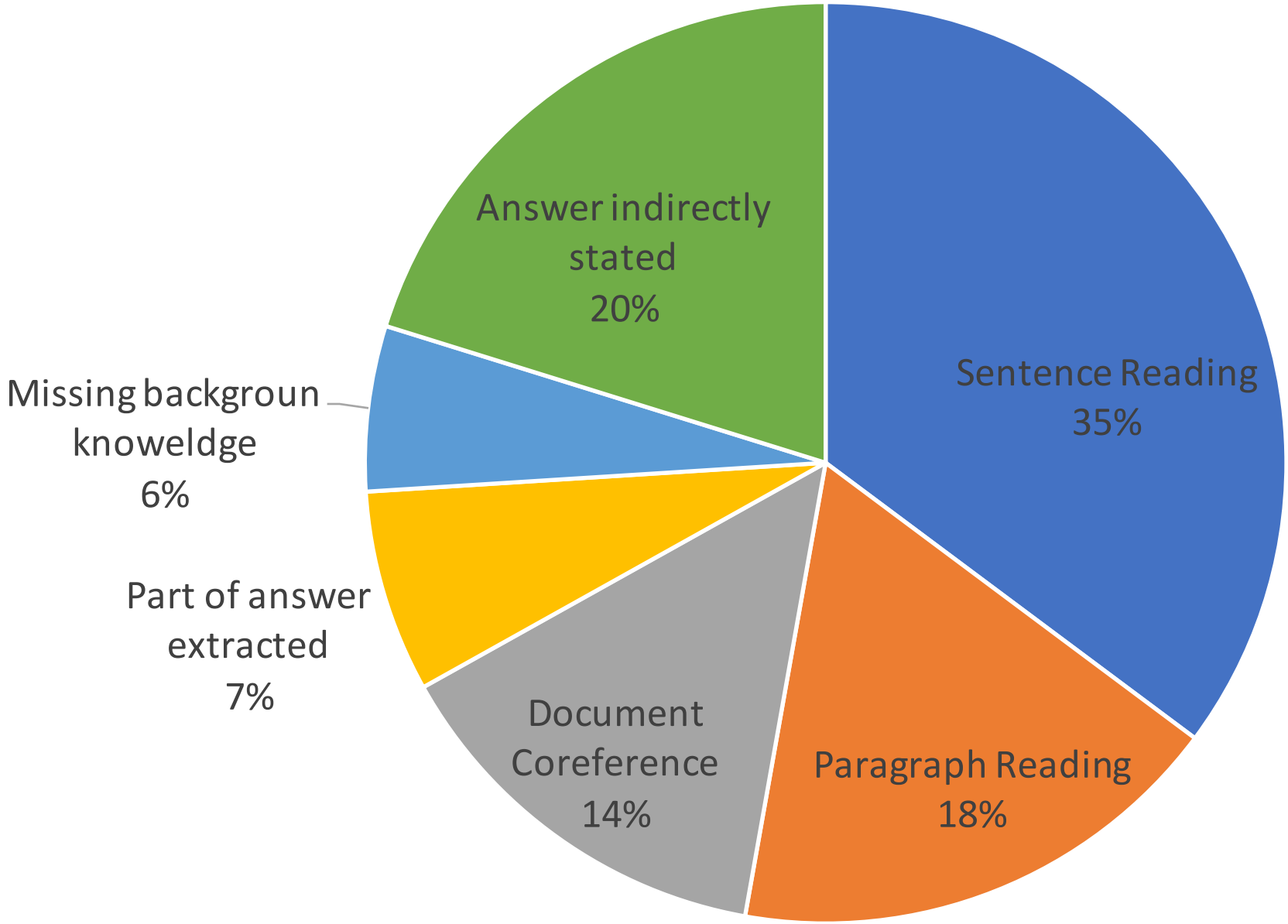


TriviaQA Leaderboard (Exact Match Scores)

Model	Web-All	Web-Verified	Wiki-All	Wiki-Verified
Best leaderboard entry (“mingyan”)	68.65	82.44	66.56	74.83
Leaderboard entry (“dirkweissen”)	67.46	77.63	64.60	72.77
Shared-Norm (Ours)	66.37	79.97	63.99	67.98
Dynamic Integration of Background Knowledge (Weissenborn et al., 2017a)	50.56	63.20	48.64	53.42
Neural Cascades (Swayamdipta et al., 2017)	53.75	63.20	51.59	58.90
Mnemonic Reader (Hue et al., 2017)	46.65	56.96	46.94	54.45
SMARNET (Chen et al., 2017	40.87	51.11	42.41	50.51

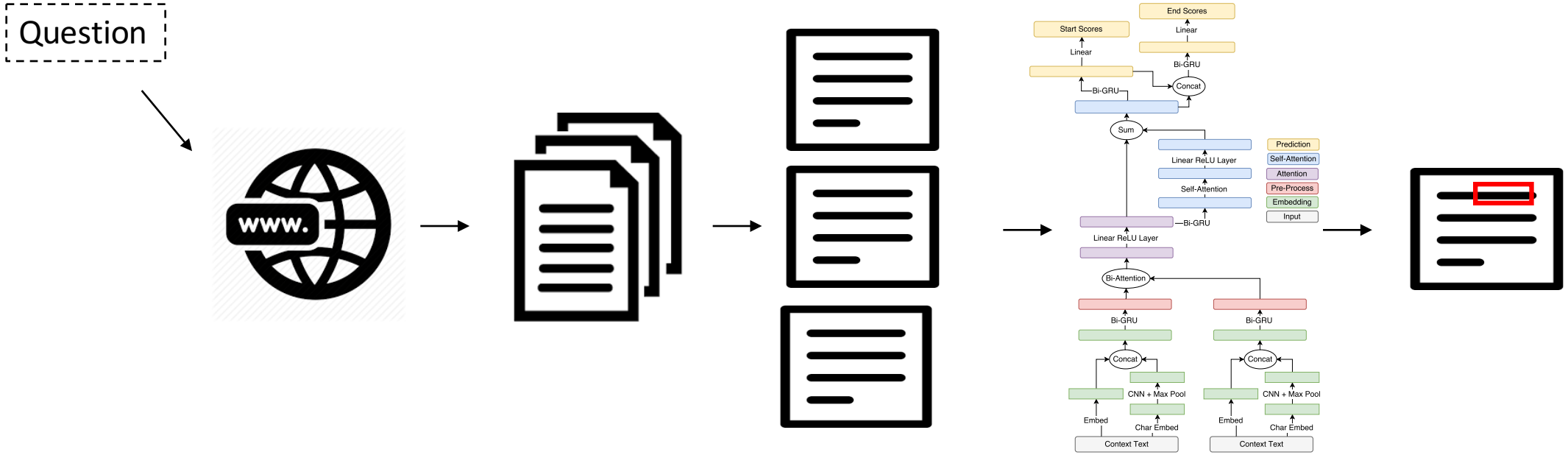
Error Analysis

- Manually annotated 200 errors made by the TriviaQA Web model
  - 40.5% are due to noise or lack of context in the relevant documents
  - Of the remaining….



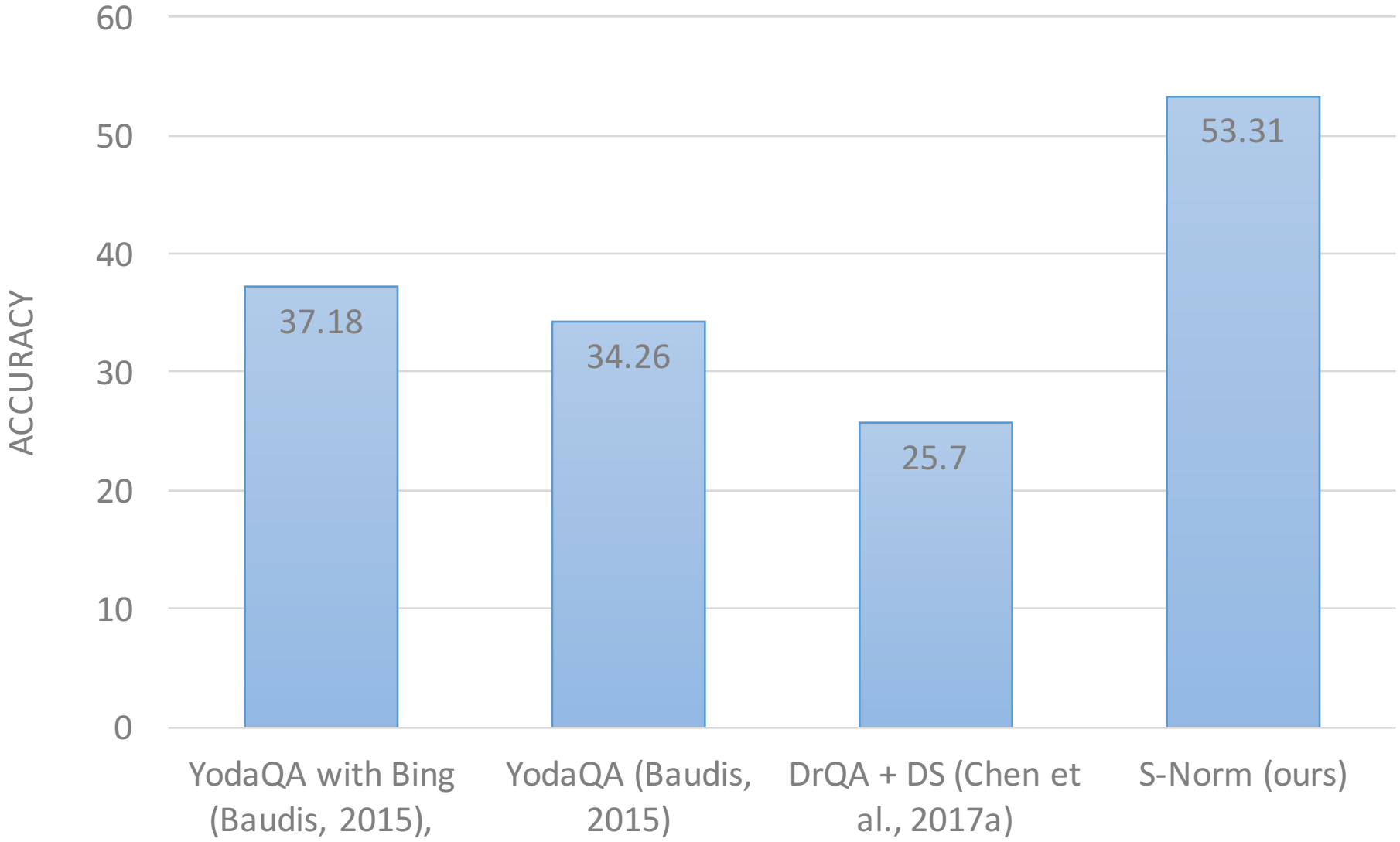
Building an Open Question Answering System

- Use Bing web search and a Wikipedia entity linker to locate relevant documents
- Extract the top 12 paragraphs, as found using the linear paragraph ranker
- Use the model trained for TriviaQA Unfiltered to find the final answer



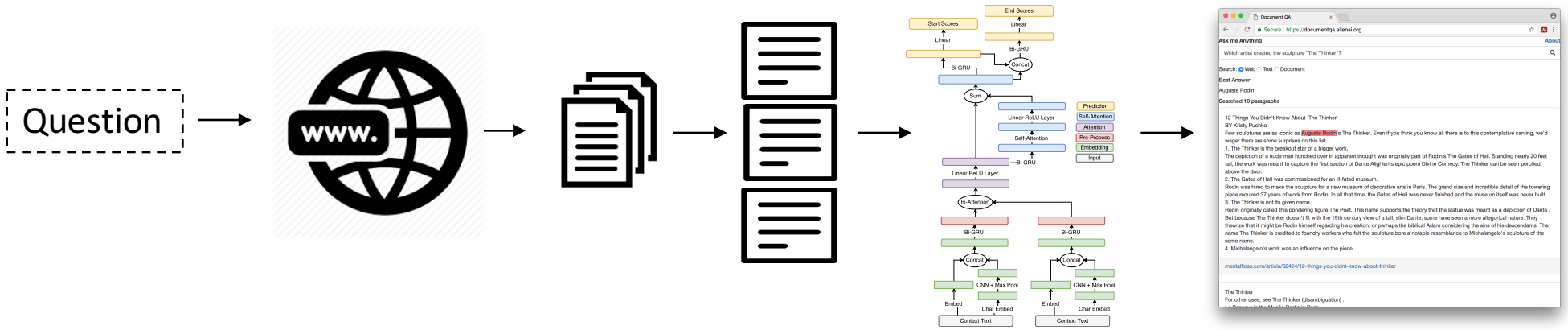
Demo

Curated Trec Results



Thank You

Demo: https://documentqa.allenai.org/



Github: https://github.com/allenai/document-qa

ALLEN INSTITUTE *for* ARTIFICIAL INTELLIGENCE