DevRev

Final Presentation

Expert Answers In A Flash: Improving Domain Specific QA

TeamID: 23

Overview

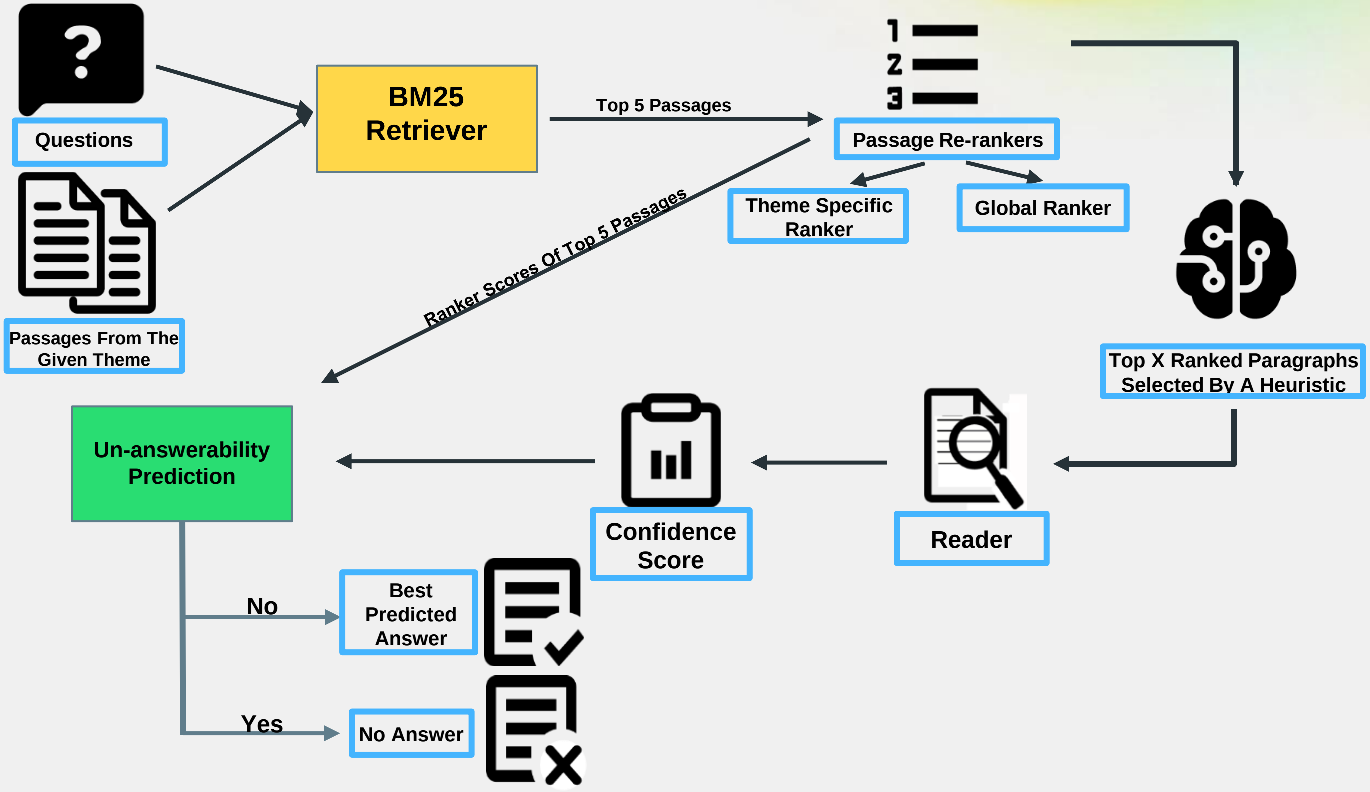
- Pipeline for domain-specific question answering in a open-QA setting
  - Involves the use of retrievers, rankers and readers

Challenges:

- Efficient resource allocation
  - Providing reader appropriate number of passages

- Ranking suffers from generalization and can improved with domain knowledge

Pipeline Overview



BM25: First Level Filter

- A probabilistic model
- Intuition: Paragraphs can be easily distinguished based on the query keywords
- Fast and effective filter
  - 16ms on average per query
  - Top 5 accuracy of BM25 is nearly 95.4%

- Alternatives: DPR (Bi-encoder)
  - Pro: captures semantics
  - Con: requires precomputing dense, data-specific vector representations
  - Doesn't provide considerable improvements

Rankers

- Essentially cross-encoders trained on query, paragraph pair
  - Classifier head to determine the semantic similarity
- Re-ranks the narrowed paragraphs

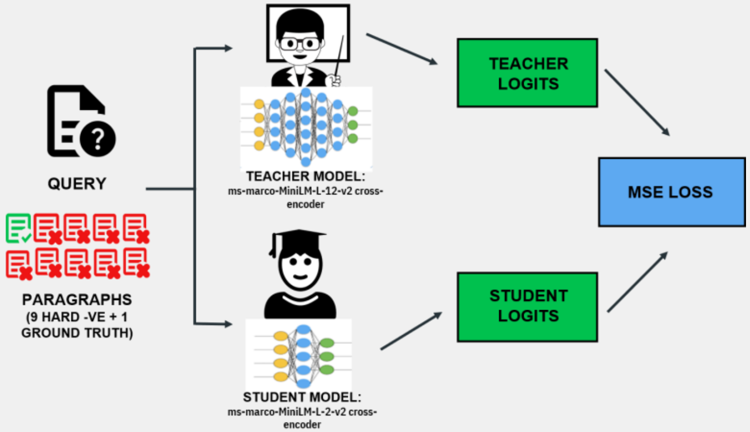
Model	Top k	Accuracy	Time per query (s)
TinyBERT cross-encoder applied on all paragraphs in the theme	1	85.02%	0.05
TinyBERT cross-encoder applied on top 10 paragraphs based on BM25	1	86%	0.02
MiniLM cross-encoder applied on top 10 paragraphs based on BM25	1 (top 1 of MiniLM)	88.71%	0.76
	2 (top 1 of BM25 and MiniLM each)	90.92%	0.76
MiniLM, and TinyBERT cross-encoder applied on top 10 paragraphs based on BM25 separately	3 (top 1 of BM25, MiniLM and TinyBERT each)	92.51%	0.82

Domain Adaptive Rankers

With Knowledge Distillation and Contrastive Loss

Knowledge Distillation

- Hard negatives are mined using BM25 retrieved documents
- Smaller student model learns from the output logits of the teacher model
  - Minimize the mean square loss (MSE)



- MiniLM teacher (L-12) and student model (L-2)
  - Task-transfer: pretrained on the ms-marco dataset for the task of passage re-ranking

Knowledge Distillation

	Top 1 Accuracy	Inference Time per Query (Colab CPU)
Student Model (MiniLM-L-2)	85.79%	305 ms
Finetuned Model	89.31%	305 ms
Teacher Model (MiniLM-L-12)	90.27%	1010 ms

- Trained for 13 epochs on 80:20 train-test split with overlapping themes
- Approaches top 1 accuracy of a pre-trained teacher model

Contrastive Loss Training

- Minimising loss translates to simultaneously maximize the similarity between the positive pairs while minimizing the same for negative pairs
- 1 positive and 9 BM25 hard negatives
- $\text{sim}(z_i, z_j)$ is the logits score of the ranker

$$\ell_{i,j} = -\log \frac{\exp(\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_k)/\tau)}$$

Epochs	Top 1 Accuracy
Pretrained	81.02%
Epoch 1	81.65%
Epoch 2	82.71%

Theme Specific Rankers

- Outperforms universal ranker when fine-tuned with specific themes
  - Considerable difference for enough training examples
- Loaded at inference time
- Fine-tuned universal ranker for 2 epochs on specific themes

	Universal Ranker	Theme Finetuning
New York City	0.747	0.761
IPod	0.804	0.885
2008 Sichuan Earthquake	0.843	0.862

Heuristic

For Difficulty Prediction

Heuristic for Difficulty Prediction

1. Not all questions are equally difficult
2. 1s is the **average** time limit per question

Varied amount of passages can be passed to reader based on question difficulty

Theme Specific Rankers

- $p(X, i)$: probability that the answer lies among the top i of the ranker's final ranked paragraphs
- $q(X, i)$: probability that the reader will solve the top I question correctly
- Expected number of correctly answered questions:

$$\sum p(X\ i\ , z[i]) \cdot q(X\ i\ , z[i])$$

- $z[i]$: number of passages passed for the ith question
- Maximize expectation over the constraint- $\sum z[i] \leq K$ for some K
  - Upperbound is on the total number of passages passed to the reader

Model for P(X,i)

- Correlation exists between ranker/retriever scores distribution and the probable location of the answer paragraph
- X is taken as the concatenated ranker-retriever scores
- Neural network with one hidden layer used to predict the p(X,i)

Algorithm

1. Initialize z[i] as 1 (assume one paragraph for each query)
2. Greedily increment the $z[j]$ variable that locally increases the expectation by the maximum amount
    1. $O(K\ log\ n)$ (with min heap)

3. Followed with a random algorithm:
    1. Randomly choose a j with $z[j]$ > 0, decrement $z[j]$ and then again increase the $z[k]$ value
        1. Redo the greedy operation

Results

- Constraint K dynamically based on **time remaining**
  - Based on time left after retrieving and ranking and average reader latency

Results:

- Average time per question: 0.97s

	Top 2	**Heuristic Approach**
Accuracy	0.854	0.898

Readers & Answerability

For answer extraction

Readers Intro

- Purpose of the reader is to apply reading comprehension algorithms to retrieved paragraphs
- Used transformer based readers which are composed of encoders and decoders that employ extractive spans

Total_loss = (Start_loss + End_loss)/2

- Where the start_loss and end_loss are the cross entropy losses for the start and end logits respectively.

Pre-Trained Readers

Model	Time (per query)	Accuracy (Exact Match)	Memory	F1 Score
Retro Reader	13.96s	90.56%	3.86 GB	87.76%

Retro Reader

- Performs well due to Sketchy reading(E-FV), Intensive reading(I-FV), and Rear Verification(RV)

Model	Accuracy (Exact Match)	Time (per query)	Memory	F1 score
roberta-base-squad2	83.27%	2020 ms	496 MB	62.33
roberta-large-squad2	89.98%	6500 ms	1420 MB	70.32
tinyroberta-squad2	79.26%	630 ms	326 MB	66.73
minilm-uncased-squad2	78.85%	305 ms	134 MB	64.85
distilbert-base	51.67%	474 ms	261 MB	45.50

Experiments on MiniLM

- Distilling BERT-base's last layer attention module - student flexibility
- Scaled dot product between last layer attention modules - similarity
- Offers the best performance-latency ratio
- Pre trained on squad 2.0

Split type	Details of fine-tuning	Exact match accuracy
Theme Independent Split	Pre-trained Minilm	78.142%
Theme Independent Split	Minilm fine-tuned on the train-split	74.890%
Theme Dependent Split	Pre-trained Minilm	78.142%
Theme Dependent Split	Minilm fine-tuned on the train-split	75.217%
Data-augmentation	Minilm fine-tuned on the train-split	70.126%
Data-augmentation	2nd model fine-tuned again on the train-split	65.515%

Challenges and Inferences

- Absence of relevant training data causes overfitting

Split type	Details of fine-tuning	Exact match accuracy
Theme Independent Split	Pre-trained Minilm	78.142%
Theme Independent Split	Minilm fine-tuned on the train-split	74.890%

- Improvement on theme-wise finetuning rather than normal split

Theme Dependent Split	Pre-trained Minilm	78.142%
Theme Dependent Split	Minilm fine-tuned on the train-split	75.217%

Data Augmentation

- Tried two kinds of data augmentations:

1. Hard negatives: generated by pairing the wrong paragraphs with each questions to extend the dataset.
2. Inserting the sentence containing the correct answer of a question in another paragraph and pairing up with corresponding question

- Can be attributed to complete change in context and latency as compared to heavier models

Data-augmentation	Minilm fine-tuned on the train-split	70.126%
Data-augmentation	2nd model fine-tuned again on the train-split	65.515%

Decoding Strategy

We designed three different t

- Find the top_n best answers - maximizing the sum of start_logits and end_logits - vectorization, time-optimal solution.
- O(nlogn) - binary search and a type of sliding window – maximum answer length.
- commonly used simple searching algorithm of O(n^2) time complexity.

Answerability

- Baseline: Reader confidence scores with threshold 0.5
- Proposed novel method uses confidence score of reader, retriever and ranker with perceptron classifier
  - Intuition: correct answer's passage reader, retriever and ranker scores must be placed higher in their score distributions.

Method	Data	Accuracy	F1
0.5 Threshold	Reader Score	95.80%	96.34%
Perceptron Classifier	Top 10 Retriever Score	69.57%	79.41%
Perceptron Classifier	Top 10 Retriever + Reader Score	97.46%	98.17%
Perceptron Classifier	Top 10 Retriever + Ranker + Reader Score	97.61%	98.53%

Here Retriever is BM25, Reader is TinyRoBERTa and Ranker is miniLM cross-encoder.

Conclusion

1. Domain Adaptable Rankers with knowledge distillation
2. Novel difficulty prediction heuristic to dynamically determine the number of passages to be read
3. Signals from retriever, ranker and reader for answerability

- Domain-Adaptability ☑
- Low Latency ☑
- High Precision ☑