Knowledge Graph And Fusion Based Transformer Approach for Multi-Hop Question Answering

Group 39

Akash Gujju, Arun Baalaaji, Balaji Chidambaram, Mukkesh Ganesh and Trisha Mandal

Email: gujju@usc.edu, arunbaal@usc.edu, bchidamb@usc.edu, mganesh@usc.edu, trishama@usc.edu

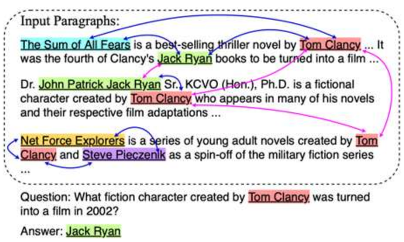
University of Southern California, Los Angeles, CA, United States of America

USC Viterbi School of Engineering

MOTIVATION

- Question Answering(QA) is an important downstream task in NLP
- Most real-world questions require multi-hop reasoning to arrive at an answer
- A machine must understand the question, identify supporting facts from multiple knowledge sources and use reasoning to generate an answer
- The goals of the project
  - Explore various fusion techniques
  - Experiment with knowledge-based information retrieval

EXAMPLE

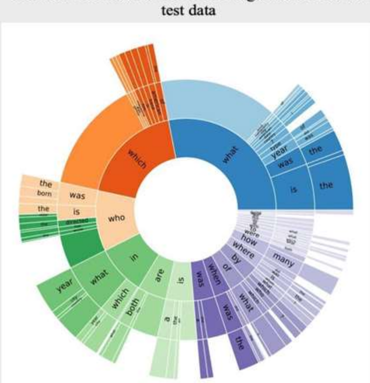


DATASET

- HotPotQA is a challenging dataset with Wikipedia Q&A pairs aimed at Multi-Hop Reasoning task
- HotPotQA has two evaluation modes:
  - Distractor setting
  - Full-Wiki setting
- Training set has 90K and Dev/Test set has 22K datapoints
- A single datapoint contains answer, question, supporting facts, contexts, level, and type

Name	Desc.	Usage	# Examples
train-easy	single-hop	training	18,089
train-medium	multi-hop	training	56,814
train-hard	hard multi-hop	training	15,661
dev	hard multi-hop	dev	7,405
test-distractor	hard multi-hop	test	7,405
test-fullwiki	hard multi-hop	test	7,405
Total			112,779

The table contains the different categories of train and test data



The figure consists of the categories of questions

EXPERIMENTS

1. Fusion experiments
- In our model we need to fuse Question Attention and Context Attention embeddings.
- Complex fusion techniques has been shown to improve the convergence rate of the model and slightly increase the performance.
- We studied two fusion techniques:

A. Multi Modal Tucker Fusion

$$y = \left(\left(\mathcal{T}_c \times_1 \left(\mathbf{q}^\top \boldsymbol{W}_q\right)\right) \times_2 \left(\mathbf{v}^\top \boldsymbol{W}_v\right)\right) \times_3 \boldsymbol{W}_o$$

$$\mathbf{z} = \left(\mathcal{T}_c \times_1 \tilde{\mathbf{q}}\right) \times_2 \tilde{\mathbf{v}} \in \mathbb{R}^{t_o}$$

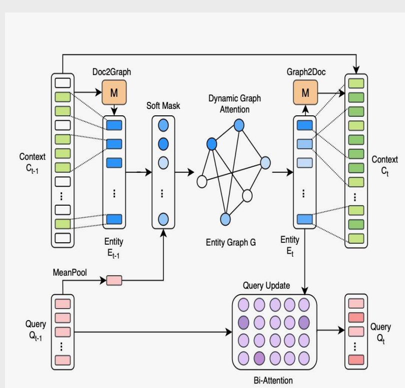
B. Multi Modal Factorized Bi-Linear pooling

$$\begin{aligned} z_i &= x^T U_i V_i^T y = \sum_{d=1}^{k} x^T u_d v_d^T y \\ &= \mathbb{1}^T (U_i^T x \circ V_i^T y) \end{aligned}$$

2. Fine-tuning Casual Language Model (CLM) with WikiText-2 corpus using hugging face trainers
3. Knowledge based information retrieval
- Entity graph creation
- Attention based context retrieval from the entity graph

ARCHITECTURE

- A paragraph selector is used to filter the relevant context information from the multiple input paragraphs
- The filtered context is used in creating an entity graph which is dynamically generated for every question
- The BERT encoder produces a representation from the input query and context paragraphs using a Bi-Attention layer
- The fusion block takes the output from the entity graph and the encoder input and fuses them together
- The LSTM prediction head produces supporting facts and the answer

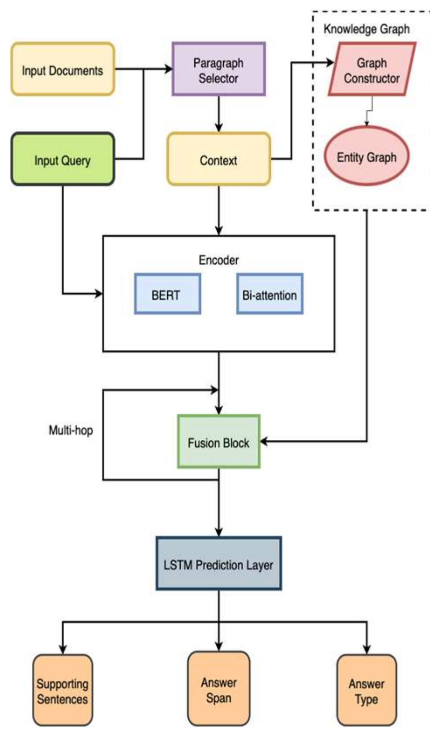


DFGN Fusion Block

RESULTS

Model	F1 Score	EM	Precision	Recall
Fusion Baseline	0.5104	0.3762	0.5300	0.5297
MFB	0.5186	0.3831	0.5393	0.5362
MFB + Tucker Fusion	0.5246	0.3941	0.5480	0.5442
Tucker Fusion	0.5155	0.3808	0.5363	0.5335
DFGN	0.6500	0.5088	0.6737	0.6672

- Fine-tuning the BERT model on the WIkiText-2 corpus improves the model F1 to 0.65
- Dynamically creating the entity graph for each input query helps the model in generating better contexts
- To achieve an F1 Score of 0.50, the baseline model takes 18 epochs while the MFB and Tucker Fusion models took only 12 and 14 epochs respectively



Architecture Diagram

FUTURE WORK

1. Incorporate fusion techniques on dynamic graph fused network
2. Fine tune the BERT model on DBPedia and ConceptNet
3. Experiment with BeerQA dataset