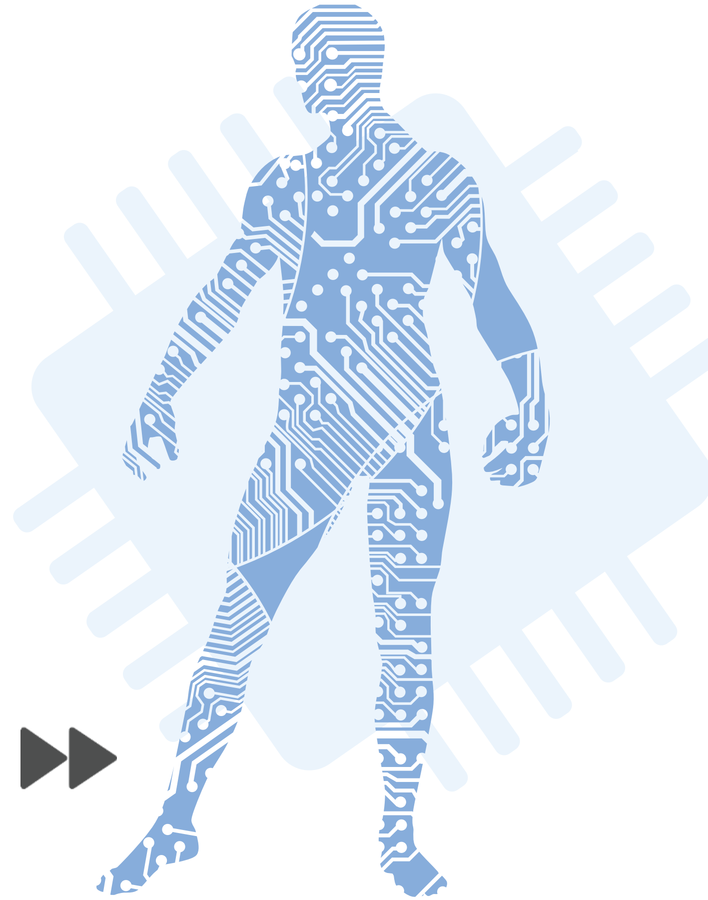
Deep Learning in NLP

Constructing a Machine Question Answering Model

Sam Cheung
chfsam@hku.hk
Supervisor: Dr. Gilbert Lui

Agenda

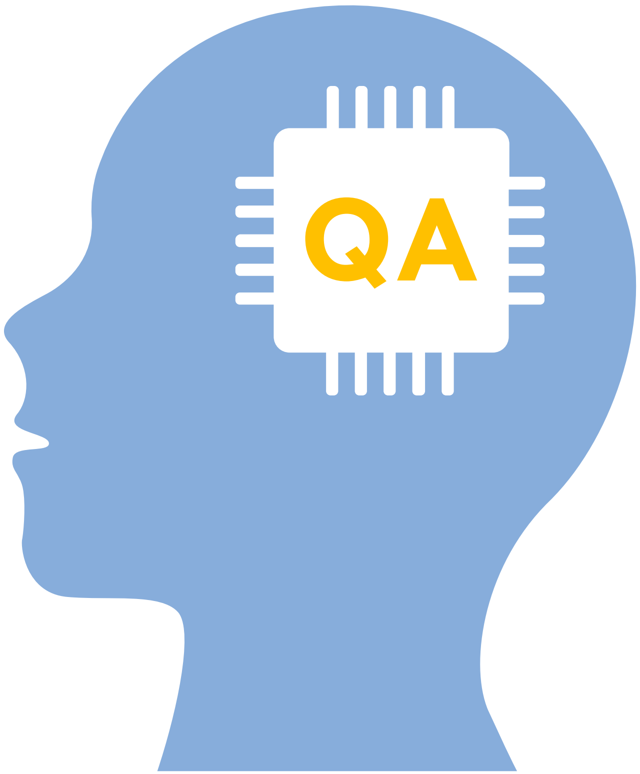
1. Research Objective
2. Data
3. Technicality
4. Experiment Results
5. Limitations & Way Forward
6. Q&A

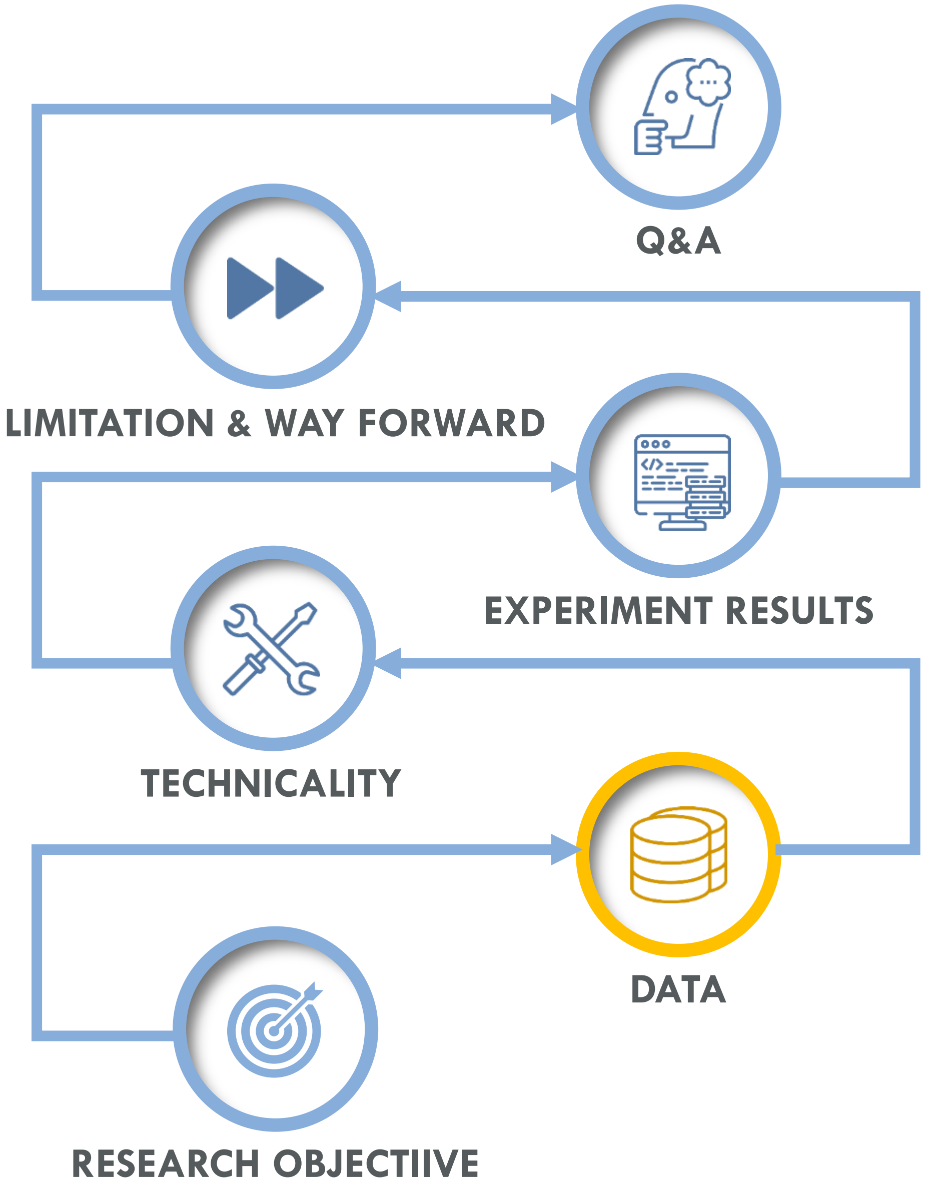


RESEARCH OBJECTIVE

Build a Machine Question Answering (QA) model to comprehend a given textual information and answer questions that are either answerable or unanswerable.

Possible applications:
Document Q&A tools, chat bot



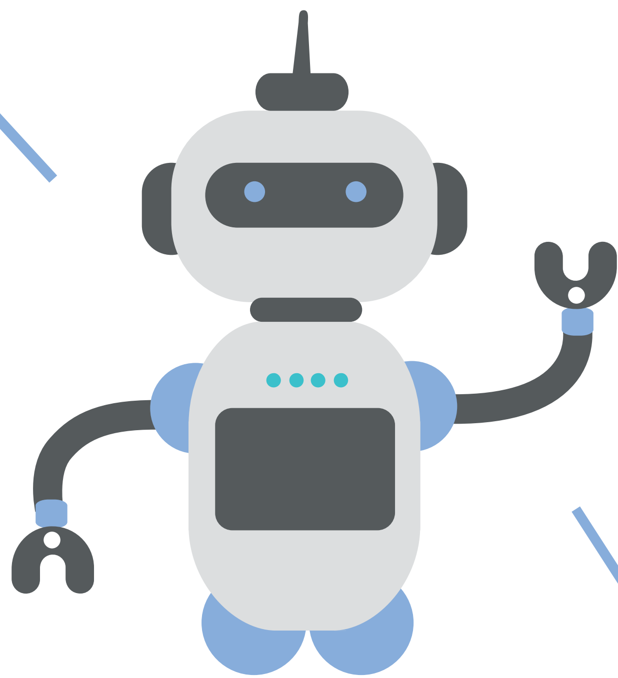

Q&A
LIMITATION & WAY FORWARD
EXPERIMENT RESULTS
TECHNICALITY
DATA
RESEARCH OBJECTIIVE

DATA

- The version 2 of the Stanford Question Answering Dataset (SQuAD)
- 150,000 context-question-answer trio from over 400 English Wikipedia articles
- Answer: a segment of texts in the given context paragraph or no answer

Answerable

Context Paragraph
Oxygen is a chemical element with symbol O and atomic number 8. It is a member of the chalcogen group…
Question
The atomic number of the periodic table for oxygen?
Answer
8

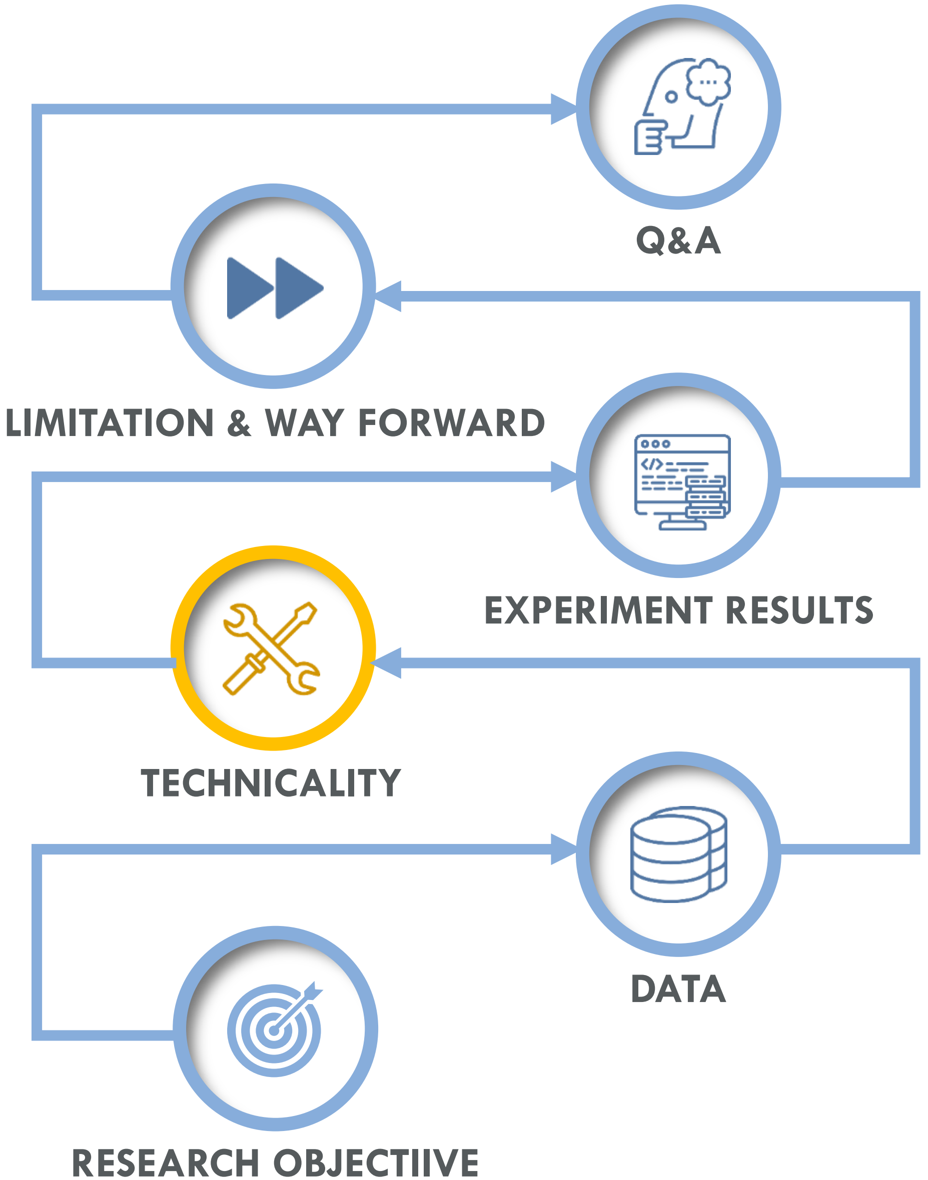


Context Paragraph
Spreading throughout the Mediterranean and Europe, the Black Death is estimated to have killed 30–60% of Europe's total population.…
Question
What percentage of people died of the Black Death in Central Asia?
Answer
Nil

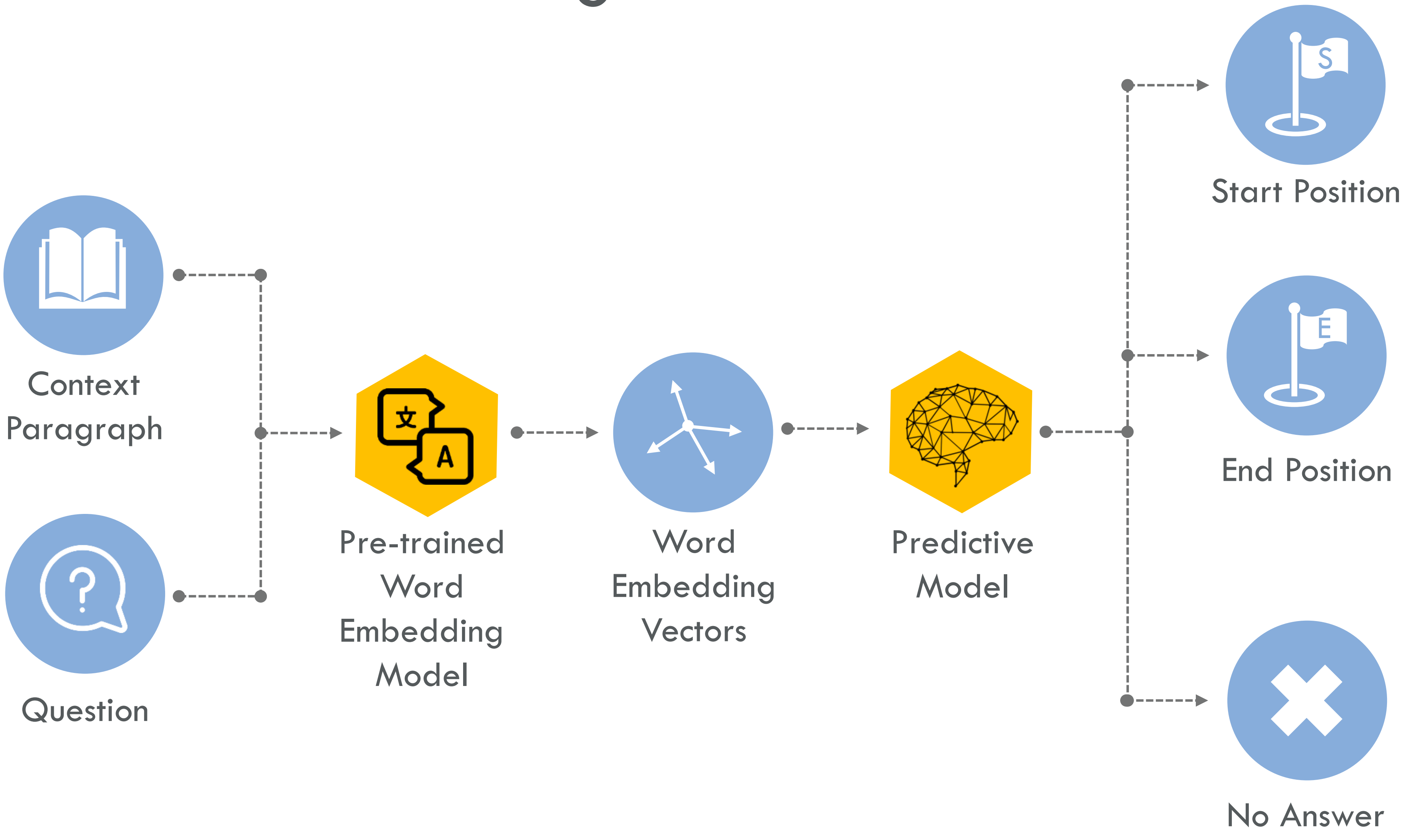
Unanswerable

DATA

	Train	Development	Test
Total examples	130319	11873	8862
Negative examples	43498	5945	4432
Total articles	442	35	28
Articles with negatives	0	35	28
Range of number of word tokens in context paragraph	[23,408]	[27,448]	-
Mean number of context tokens	116	112	-
Percentage of examples with number of context tokens > 300	0.9%	3.5%	-
Range of number of word tokens in question	[4,28]	[4,17]	-
Mean number of question tokens	10	10	-


Q&A
LIMITATION & WAY FORWARD
EXPERIMENT RESULTS
TECHNICALITY
DATA
RESEARCH OBJECTIIVE

Program Flow



Technicality

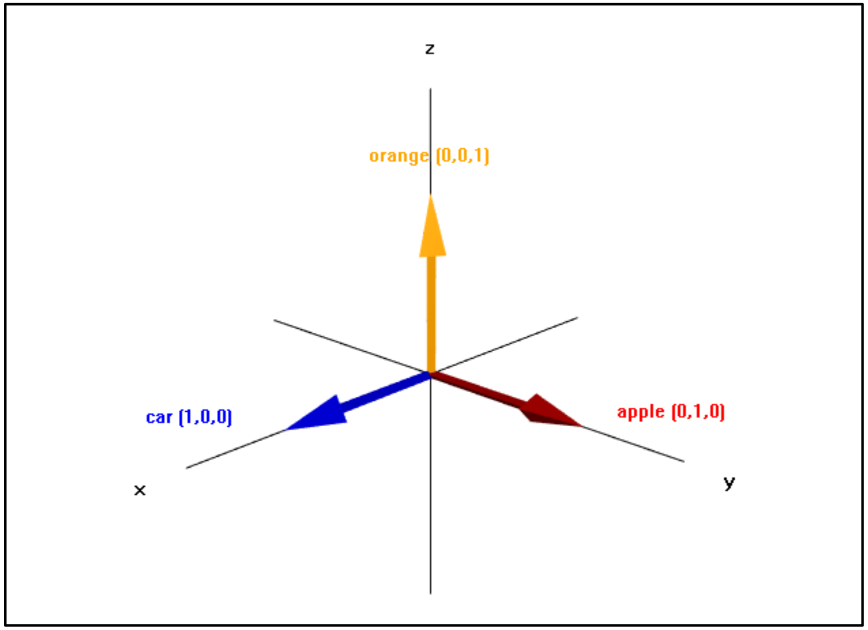
TECHNICALITY

- Pre-trained Word Embedding Models
  - Global Vectors (GloVe)
  - Embeddings from Language Models (ELMo)
  - Bidirectional Encoder Representations from Transformers (BERT)

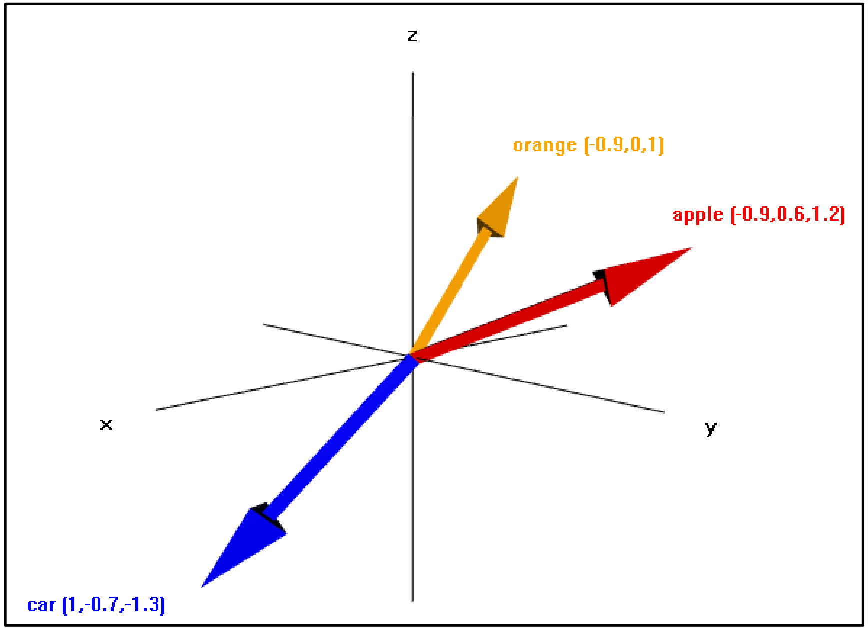
- Predictive Model
  - Bidirectional Attention Flow (BiDAF)
  - BERT-finetuning
    - Feedforward Neural Network (FNN)
    - Gating Mechanism
    - Highway Network
    - Residual Learning

TECHNICALITY – Word Embedding

- Represent each word token in a fixed-length numeric vector
- Relatively low dimension → Efficient representation
- Capture semantic and syntactic information in word tokens

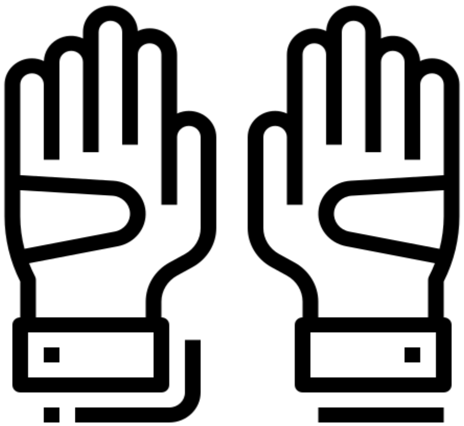


One-hot vectors



Word Embedding vectors

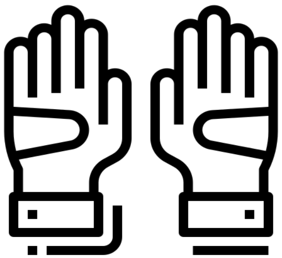
TECHNICALITY - GloVe



Global Vectors (GloVe) (Pennington et al., 2014)

- Train word vectors based on the number of co-occurrence of word pairs obtained from training text corpus
- Computationally efficient
- Context-free embedding
- Cannot handle polysemy, e.g. “bank account”, “river bank”
- Cannot handle out-of-vocabulary tokens

TECHNICALITY – GloVe



K : Number unique words in training text corpus

X : Word-word co-occurrence matrix

X_{ij} : number of times word w_i co-occur with word w_j in text corpus

GloVe model trains 2 vectors v_i and $\widetilde{v_i}$ for the same word w_i

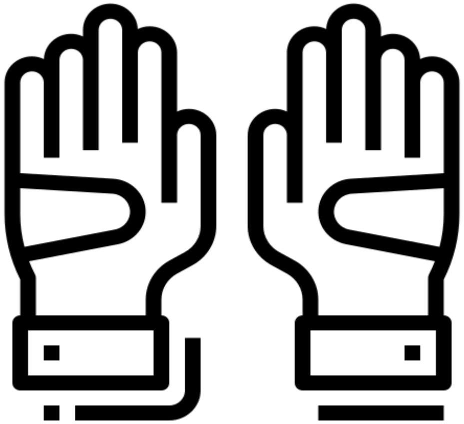
Final vector representation for word $w_i = v_i + \widetilde{v_i}$

Objective function:

$$J = \sum_{i=1}^{K} \sum_{j=1}^{K} f(X_{ij}) \left(v_i \widetilde{v_j} + b_i + \widetilde{b_j} - \log X_{ij} \right)^2, \text{ where } X_{ij} \neq 0.$$

- $f(x)$ is a weighting function to restrain the influence of the common word pairs e.g. “this is”, “I am” in word vectors training.
- b_i : bias term for word w_i

TECHNICALITY - GloVe



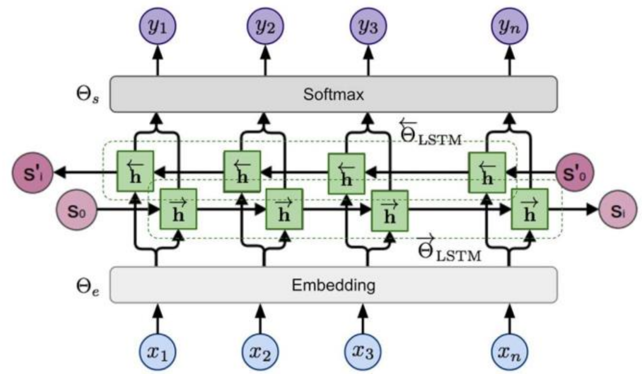
- Pre-trained GloVe model vector dimension: 300
- Fix every context paragraph to have 300 word tokens and every question to have 30 word tokens
- GloVe output: $\mathbf{C} \in \mathbb{R}^{300\times300}$ (context); $\mathbf{Q} \in \mathbb{R}^{300\times30}$ (question)

TECHNICALITY – ELMo

Embeddings from Language Models (ELMo) (Peters et al., 2018)

- ELMo vectors are functions of the intermediate states of a deep bidirectional language model (biLM)
- The biLM is indeed bidirectional Long Short Term Memory (biLSTM)
- Character-based language model
- Contextualized embedding
- Pre-trained ELMo vector dimension: 1024
- ELMo Output: $\mathbf{C} \in \mathbb{R}^{1024 \times 300}$ (context); $\mathbf{Q} \in \mathbb{R}^{1024 \times 30}$ (question)

TECHNICALITY – ELMo



Step	Details	Parameters
1	A series of word tokens is first represented by context insensitive vector $x_j, j = 1,2, \dots, K$	Θ_e
2	$x_1, x_2, \dots, x_K$ are fed into the 2 biLSTM layers to obtain 4 sets of context-dependent hidden states 1st layer forward LSTM: $\overrightarrow{\mathrm{h}_{1,1}}, \overrightarrow{\mathrm{h}_{2,1}}, \dots, \overrightarrow{\mathrm{h}_{K,1}}$ 1st layer backward LSTM: $\overleftarrow{\mathrm{h}_{1,1}}, \overleftarrow{\mathrm{h}_{2,1}}, \dots, \overleftarrow{\mathrm{h}_{K,1}}$ 2nd layer forward LSTM: $\overrightarrow{\mathrm{h}_{1,2}}, \overrightarrow{\mathrm{h}_{2,2}}, \dots, \overrightarrow{\mathrm{h}_{K,2}}$ 2nd layer backward LSTM: $\overleftarrow{\mathrm{h}_{1,2}}, \overleftarrow{\mathrm{h}_{2,2}}, \dots, \overleftarrow{\mathrm{h}_{K,2}}$	$\overrightarrow{\Theta}_{LSTM}$; $\overleftarrow{\Theta}_{LSTM}$
3	$\overrightarrow{\mathrm{h}_{j,2}}$ is used to predict the next word w_{j+1} via a softmax layer while $\overleftarrow{\mathrm{h}_{j,2}}$ is used to predict the previous word w_{j-1} via a softmax layer.	Θ_s

Objective Function: $\ell = \sum_{j=1}^{K} \left[\log \Pr\left(w_j \middle| w_1, \dots, w_{j-1}; \Theta_e, \overrightarrow{\Theta}_{LSTM}, \Theta_s\right) + \log \Pr\left(w_j \middle| w_{j+1}, \dots, w_N; \Theta_e, \overleftarrow{\Theta}_{LSTM}, \Theta_s\right) \right]$

TECHNICALITY – ELMo

- In downstream NLP task like QA, we want to generate representation vector for the context and question word tokens
- The ELMo vectors are functions of the internal states of the biLM
- $\mathrm{ELMo}_j = \gamma^{task} \sum_{l=0}^{2} s_j^{task} \mathrm{h}_{j,l}$, where $\mathrm{h}_{j,0} = x_j$, $\mathrm{h}_{j,l} = \left[\overrightarrow{\mathrm{h}_{\mathrm{j},l}} , \overleftarrow{\mathrm{h}_{\mathrm{j},l}} \right]$

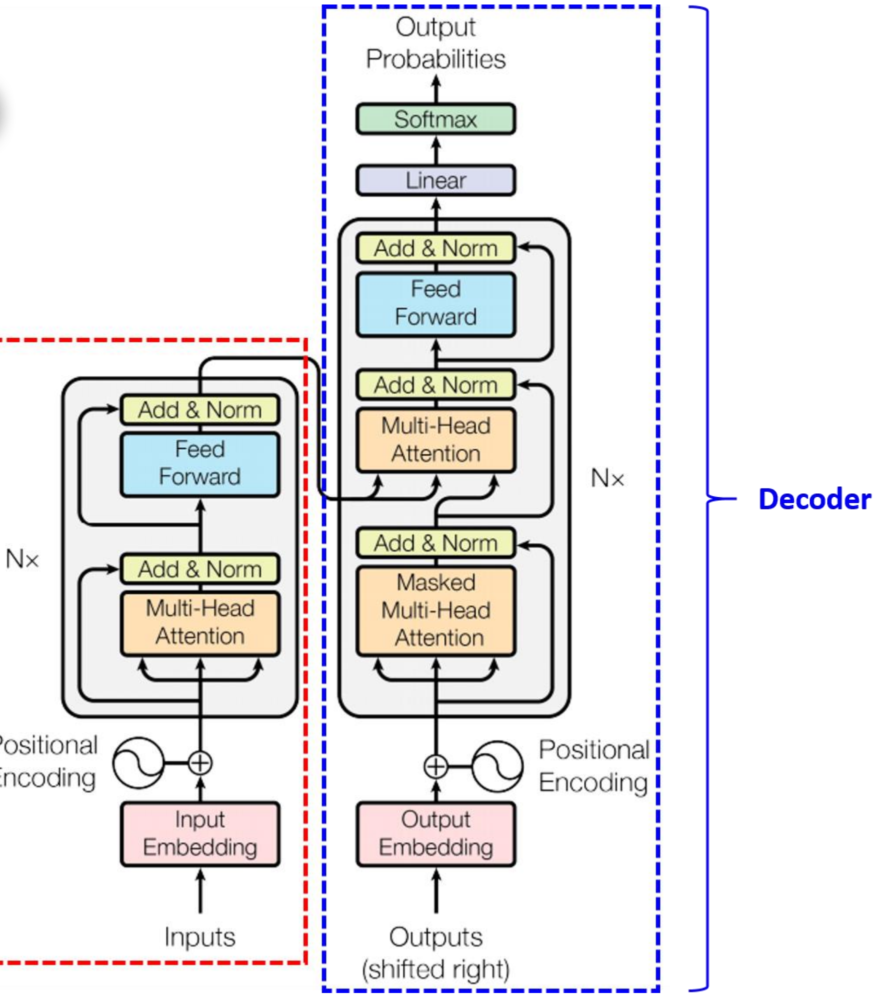
TECHNICALITY – BERT

Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018)

- The backbone of BERT is Transformer, which is a deep neural network with extensive multihead-attention mechanism
- Contextualized embedding

Pre-trained BERT model	Number of transformer layer	Dimension of output vector	No. of self-attention heads	Total no. of parameters
BERT-Base	12	768	12	110M
BERT-Large	24	1024	16	340M

TECHNICALITY – BERT



Transformer

- Deep neural network
- Consist of Encoder and Decoder block
- Multihead self-attention mechanism
- Attention: Dot product of pairs of vector representations
- Multihead: Performing several self-attentions simultaneously
- Residual connection

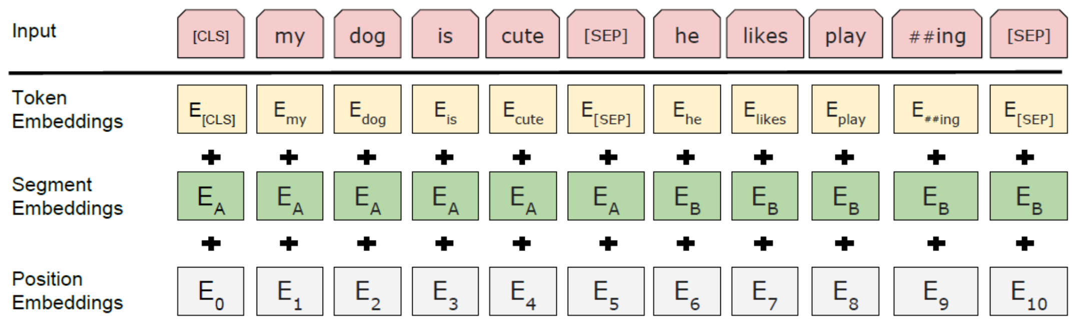
TECHNICALITY - BERT

- Training of BERT
  - **Masked Language Model (MLM)**
    - A word's meaning should be conditioned on both the left and right context simultaneously
    - 15% of word piece tokens in the training corpus for BERT are randomly masked and the BERT model
  - **Next Sentence Prediction**
    - A binary classification task
    - given an input sentence A, the model has to predict whether sentence B is the next sentence to A.

TECHNICALITY – BERT

Characteristics of BERT

- Input to Transformer
  - Word Piece Embeddings
    - e.g. "largely" → "large" & "##ly"
  - Segment Embeddings
  - Position Embeddings



- Output
  1. **BERT-Base** $\mathbf{C} \in \mathbb{R}^{320\times768}$ (context) and $\mathbf{Q} \in \mathbb{R}^{40\times768}$ (question) [for BiDAF]
  2. **BERT-Base** $\mathbf{X} \in \mathbb{R}^{384\times768}$ or **BERT-Large** $\mathbf{X} \in \mathbb{R}^{384\times1024}$
     (combine context and question tokens in 1 sequence) [for BERT finetuning]

TECHNICALITY – PREDICTIVE MODEL

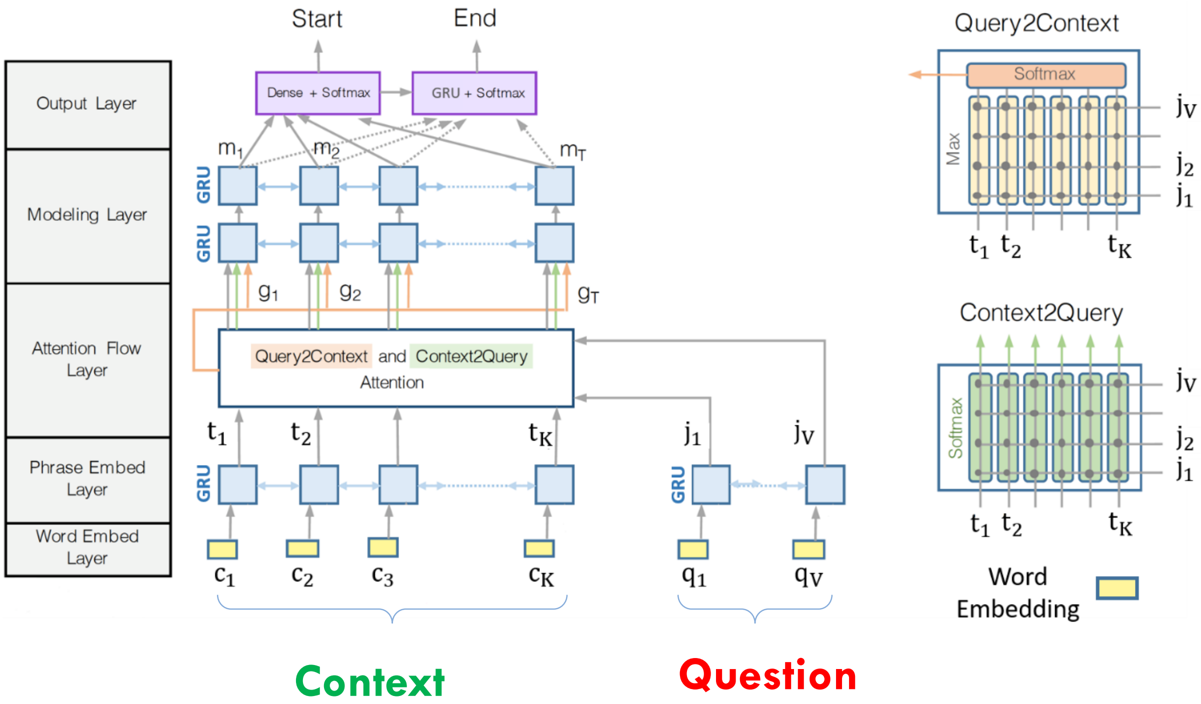
Bidirectional Attention Flow (BiDAF)

- Apply to GloVe/ELMo/BERT-Base

BERT finetuning with modified classification layer

- Applicable to BERT-Base and BERT-Large only
- Recall that BERT vectors have gone through the multihead self-attention under Transformer's architecture
- A simple classification layer can already achieve satisfactory performance

TECHNICALITY – BiDAF



Bidirectional Attention Flow (BiDAF) (Seo et al., 2016)

- Utilizes the attention mechanism bidirectionally in order to obtain
  - question-aware representation of tokens in the context paragraph
  - context-aware representation of the question tokens

TECHNICALITY – BiDAF

1. Before applying softmax to the **logit vector** of (starting/ending) poistion to obtain the final probability vector,
2. Add a trainable weight (w_s/w_e) (indicating for no answer) to the **logit vector**.
3. Threshold for determining whether the question is answerable or not

- Objective Function
- $\ell = -\frac{1}{N}\sum_i^N \left[\log\left(\mathbf{p}^s_{y_i^s}\right) + \log\left(\mathbf{p}^e_{y_i^e}\right)\right]$
- where y_i^s and y_i^e are the ground truth starting and ending position of the *i*-th training sample respectively and $\mathbf{p}_k$ is the k-th entry in the predicted probability vector $\mathbf{p}$.

TECHNICALITY – BiDAF

1. After obtaining a list of starting and ending position probabilities, invalid starting and ending position pairs are filtered away.
2. Given that the majority of answer tokens' length is less than 20, for computational efficiency the maximum length of answer tokens is restricted to be 16.
3. The starting and ending position pair with the maximum value of $p_s \times p_e$ is chosen.
4. In addition, if any of the starting and ending position in the pair falls into the no answer position, the question is determined as unanswerable.

TECHNICALITY – BiDAF

Hyperparameters of BiDAF model using GloVe/ELMo/BERT input

	GloVe	ELMo	BERT-Base
Sequence length of context paragraph tokens (K)	300	300	320
Sequence length of question tokens (V)	30	30	40
Dimension of word vector (d)	300	1024	768
Learning rate	0.001	0.001	3e-5
Batch size	60	32	16
Optimizer	Adam	Adam	Adam
Training Device	12GB GPU	12GB GPU	12GB GPU

TECHNICALITY – BERT Finetuning

- BERT output $\mathbf{X} \in \mathbb{R}^{384\times768}$ or $\mathbf{X} \in \mathbb{R}^{384\times1024}$ (including context and question)

- Simple Feedforward neural network layer **(FNN)**

$$\mathbf{Y} = \mathbf{W}\mathbf{X} + \mathbf{b}\,,$$
$$\text{where } \mathbf{Y} \in \mathbb{R}^{2\times\boldsymbol{K}}(\text{logit}), \mathbf{W} \in \mathbb{R}^{2\times d}, \mathbf{b} \in \mathbb{R}^{2}$$

- If the sum of starting and ending logits of the best answer do not pass the pre-determined threshold, the question is determined as unanswerable.

TECHNICALITY – BERT Finetuning

Gating Mechanism (Xue & Li, 2017)

$$\mathbf{Y} = \mathrm{ReLU}(\mathbf{W_R X} + \mathbf{b_R}) \odot \tanh(\mathbf{W_T X} + \mathbf{b_T})$$

- $\odot$ is element-wise vector multiplication
- Note that $\mathrm{ReLU}(x) = \max(0, x)$
- Selectively output features that are crucial to the prediction task and increase model performance.

Highway Network (Srivastava et al., 2015)

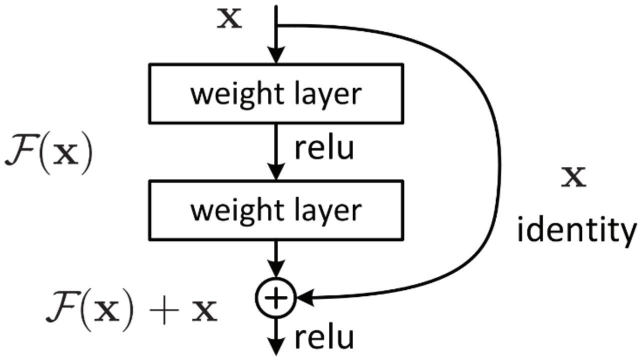
$$\mathbf{Y} = \mathrm{ReLU}(\mathbf{W}_H \mathbf{X} + \mathbf{b_H}) \odot \sigma(\mathbf{X W_T} + \mathbf{b_T}) + \mathbf{X} \odot [1 - \sigma(\mathbf{X W_T} + \mathbf{b_T})]$$

- Used to optimize the training of deep neural networks
- Learn to regulate the flow of information through a network.

TECHNICALITY – BERT Finetuning

Residual Learning (He et al., 2016)

- Combat the inefficient training of deep neural network
- Characterized by directly forwarding and adding the output $\mathbf{X}$ from an earlier neural network layer L_k to the output of later layer L_{k+r}.
- Later layers only require learning the incremental/residual information $F(x)$ instead of learning the output from earlier neural network layers from scratch



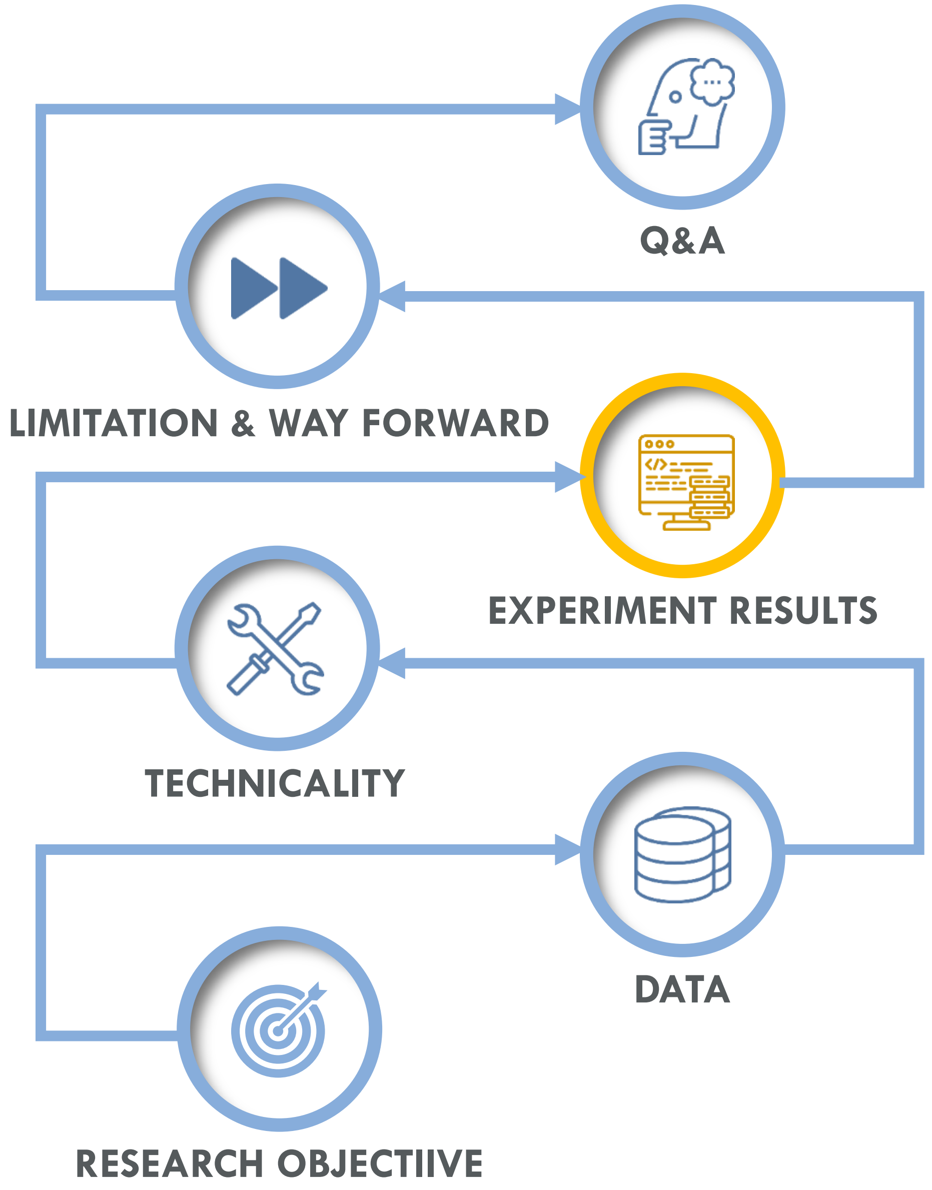
TECHNICALITY – BERT Finetuning

Objective function

- $\ell = -\frac{1}{N}\sum_i^N \left[\log\left(\mathbf{p}^s_{y_i^s}\right) + \log\left(\mathbf{p}^e_{y_i^e}\right)\right]$
- where y_i^s and y_i^e are the ground truth starting and ending position of the *i*-th training sample respectively and $\mathbf{p}_k$ is the k-th entry in the predicted probability vector $\mathbf{p}$.

Model Training

- Trained on Cloud Tensor Processing Unit (TPU)
- Batch size: 32
- Learning rate: 3e-5
- Optimizer: Adam



Q&A
LIMITATION & WAY FORWARD
EXPERIMENT RESULTS
TECHNICALITY
DATA
RESEARCH OBJECTIIVE

Evaluation Metrics

Assume N questions

Pred_i: Predicted answer tokens for Question i

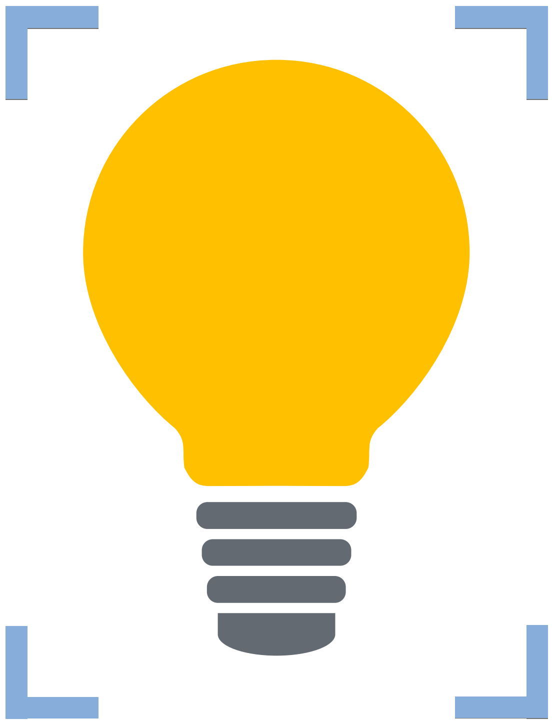
Truth_i: Ground truth answer tokens for Question i

Exact Match	F1 Score
Whether the predicted answer exactly match with the ground truth answer	Proportion of predicted answer tokens that match with the ground truth answer tokens
$\text{EM} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}\{\text{Pred}_i = \text{Truth}_i\}$	$\text{F1} = \frac{1}{N}\sum_{i=1}^{N} \frac{2 \times \text{Recall}_i \times \text{Precision}_i}{\text{Recall}_i + \text{Precision}_i}$ $\text{Recall}_j = \frac{\text{No. of tokens in Pred}_j \in \text{Truth}_j}{\text{No. of tokens in Truth}_j}$ $\text{Precision}_j = \frac{\text{No. of tokens in Pred}_j \in \text{Truth}_j}{\text{No. of tokens in Pred}_j}$

Model Results on Development Set

No.	Model	F1	EM	Ensemble F1	Ensemble EM
1	**BiDAF + GloVe**	54.622	50.678	-	-
2	**BiDAF + ELMo**	62.388	59.244	-	-
3	**BiDAF + BERT-Base Uncased**	59.695	56.565	-	-
4	**BERT-Base Uncased + 1FFN**	77.158	74.050	-	-
5	**BERT-Large Uncased + 1FFN**	80.884	77.899	81.450	78.573
6	**BERT-Large Uncased + 2FFN**	81.168	78.253	81.841	79.121
7	**BERT-Large Uncased + 1FFN + ReLU**	80.628	77.756	81.429	78.624
8	**BERT-Large Uncased + Highway networks**	80.720	77.975	81.211	78.691
9	**BERT-Large Uncased + 1 Residual learning block**	81.243	**78.371**	81.914	79.045
10	**BERT-Large Uncased + 5 Residual learning blocks**	81.233	78.278	82.016	79.272
11	**BERT-Large Uncased + Gating**	**81.404**	78.169	82.001	79.205
12	**Ensemble of 10 and 11 (6 models)**	-	-	**82.294**	**79.542**

Observations and Findings



- Contextualized word embedding better than context-free's
- BERT finetuning approach significantly outperforms BiDAF
- BERT does not synergize with BiDAF
- BERT-Large outperforms BERT-Base
- Ensemble gives extra boost to prediction performance

Error Analysis (From Ensemble Model)

01

Failure to handle certain questions with lexical variation

Context

Instead, Kublai Khan, the founder of the Yuan dynasty, favored Buddhism, especially the Tibetan variants. As a result, Tibetan Buddhism was established as the **de facto** state religion...

Question

What was the Yuan's **unofficial** state religion?

Ground Truth Answer

Tibetan Buddhism

Predicted Answer

Nil

Error Analysis (From Ensemble Model)

02

Trapped by multiple sentence reasoning

Context

...In front of the field of macrocilia, on the mouth "lips" in some species of Beroe, is a pair of narrow strips of adhesive epithelial cells on the stomach wall that "zip" the mouth shut when the animal is not feeding, by forming intercellular connections with the opposite adhesive strip. This tight closure streamlines the front of the animal when it is pursuing prey.

Question

What does the beroe do when pursuing prey?

Ground Truth Answer

"zip" the mouth shut

Predicted Answer

Nil

Error Analysis (From Ensemble Model)

03

Tricked by small twists in the questions

Context

Southern California is also home to a large home grown surf and skateboard culture. Companies such as Volcom, Quiksilver, No Fear, RVCA, and Body Glove are all headquartered here. Professional skateboarder Tony Hawk … and professional snowboarder Shaun White live in southern California...

Question

Where does professional surfer Tony Hawk live?

Ground Truth Answer

Nil

Predicted Answer

southern Califonia


Q&A
LIMITATION & WAY FORWARD
EXPERIMENT RESULTS
TECHNICALITY
DATA
RESEARCH OBJECTIIVE

Limitations and Way Forward

Design of training dataset
Question words too similar to context paragraph words

Model Training
Include more training source

Hyperparamters tuning
Learning rate, dropout, number of neural network layers, etc.

Additional variables
Add input variables like part-of-speech or name-entity tagging

Q&A