Siamese Networks

FPT Education
FPT UNIVERSITY

- Introduction
- Model architecture
- Lost function
- One-shot learning

- A Siamese Network is a neural network which uses the same weights while working in tandem on two different input vectors to compute comparable output vectors
- Question Duplicates

How old are you? = What is your age?

Where are you | from? ≠ Where are you | going?

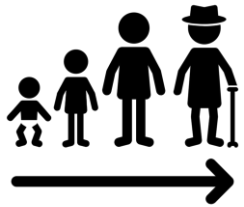
What do Siamese Networks learn?

FPT Education
FPT UNIVERSITY

I am happy because I am learning

Classification: categorize things

Siamese Networks: Identify similarity between things



What is your age?
How old are you?

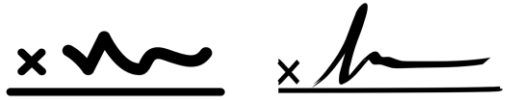
Difference or Similarity

- Aiming to identify what's makes two input similar and what makes them different.
- Computing a single similarity score representing the relationship between the two input

Siamese Networks in NLP

FPT Education
FPT UNIVERSITY

Handwritten checks

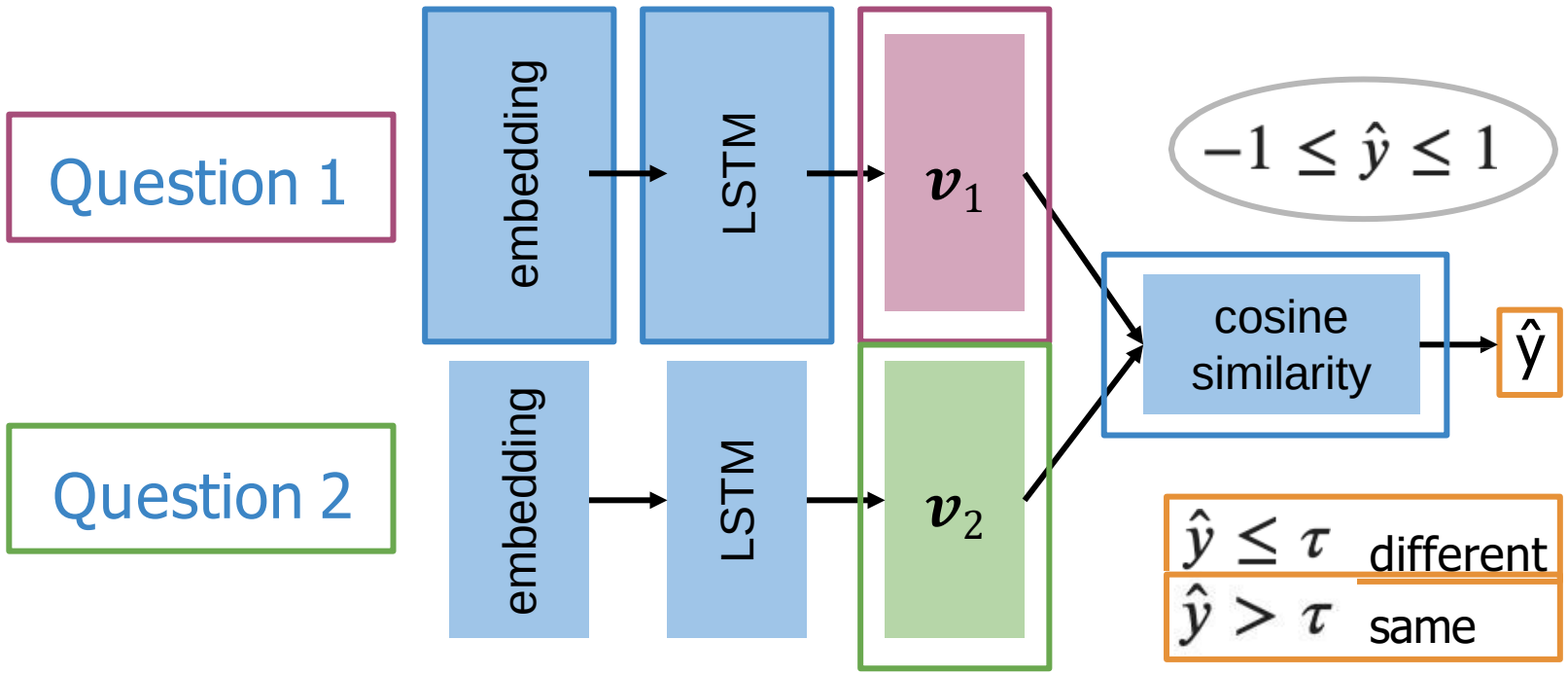


What is your age?
How old are you?

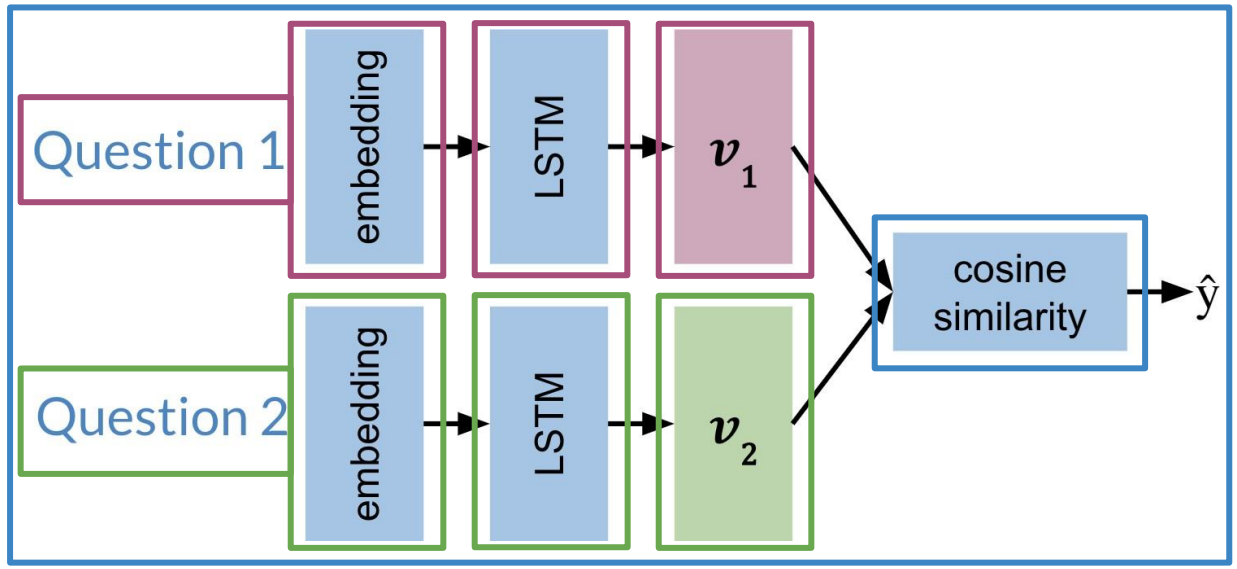
Question duplicates

Queries

Model Architecture



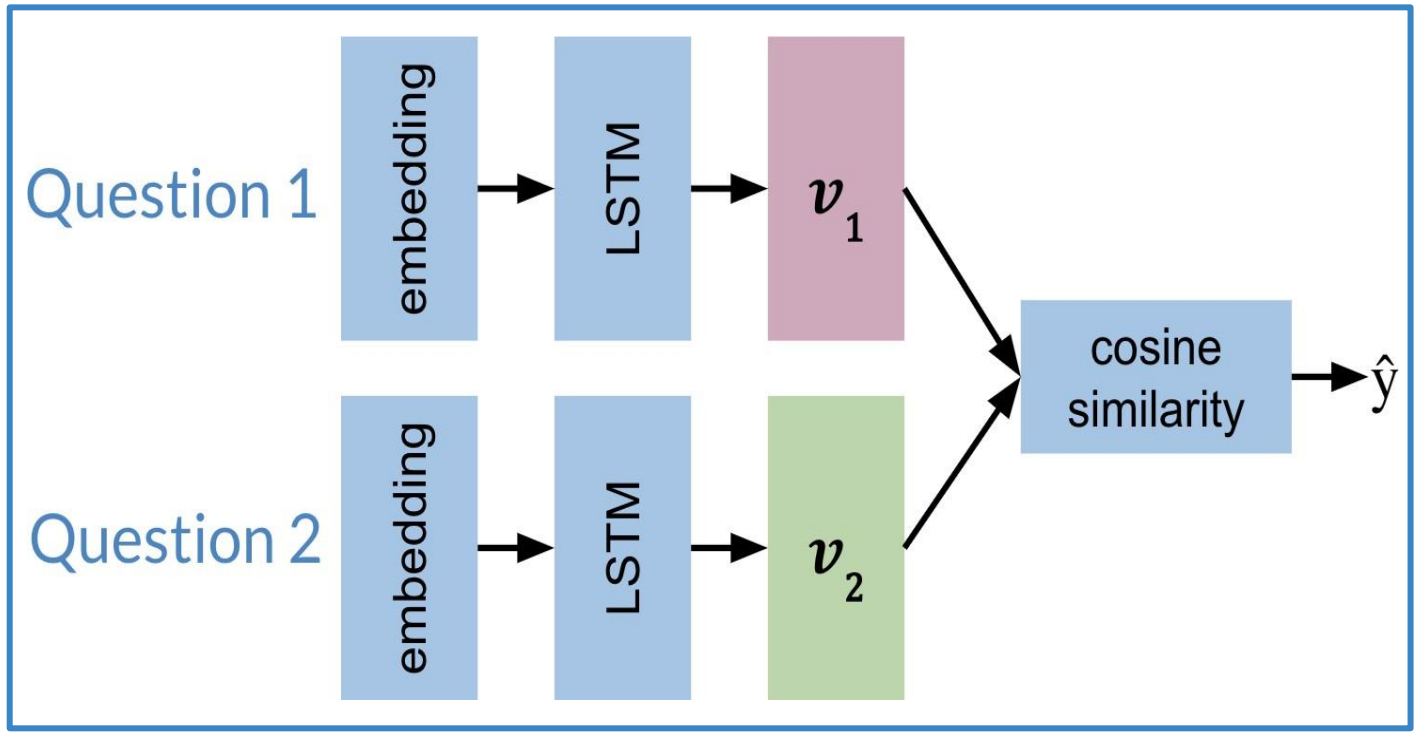
Model Architecture



1) Inputs
2) Embedding
3) LSTM
4) Vectors
5) Cosine Similarity

- a siamese network made up of two identical subnetworks
- each question gets transformed into an embedding and passed through an LSTM layer,
- take the outputs of each of the subnetworks and compare them using cosine similarity

Loss Function



$$\hat{y} = s(v_1, v_2)$$

- Calculate the similarity between the two questions

Loss Function

$$\cos(v_1, v_2) = \frac{v_1 \cdot v_2}{||v_1||\,||v_2||}$$

$$s(v_1, v_2)$$

How old are you? — Anchor

What is your age? — Positive

$s(A, P)$

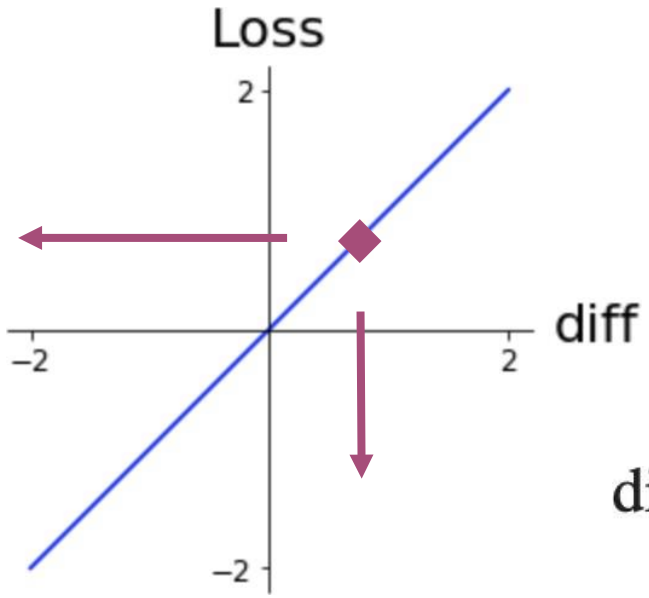
Where are you from? — Negative

$s(A, N) \approx -1$

$s(A, N) - s(A, P)$

- Anchor is use to compare against two other questions
- Questions that have the same meaning as the anchor are called positive questions
- questions do not have the same meaning as the anchor are called negative questions

Loss Function



$$\text{diff} = \text{s}(A, N) - \text{s}(A, P)$$

Triplets

How old are you? **A**nchor

What is your age? **P**ositive

Where are you from? **N**egative

Triplets

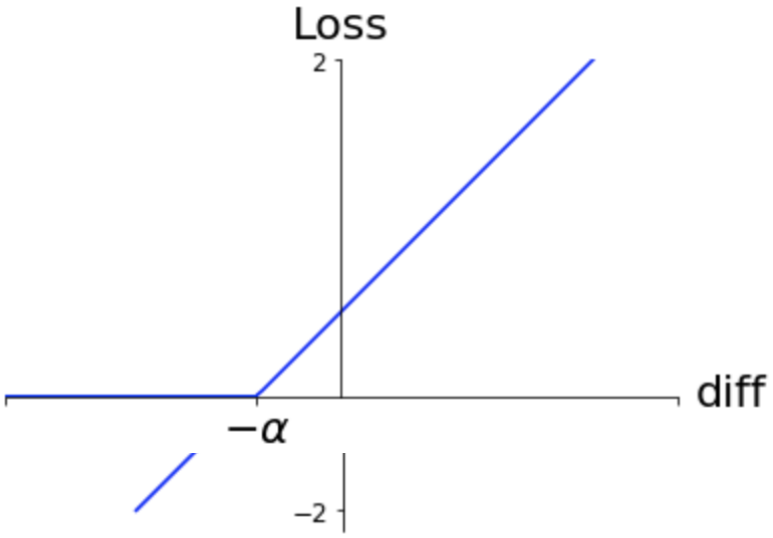
Whether or not a question has the same meaning as the anchor

Triplet Loss

How old are you? **A**nchor

What is your age? **P**ositive

Where are you from? **N**egative



Simple loss:

$$\text{diff} = \text{s}(A, N) - \text{s}(A, P)$$

With non-linearity

$$\mathcal{L} = \begin{cases} 0; & \text{if } diff \leq 0 \\ diff; & \text{if } diff > 0 \end{cases}$$

With alpha margin

$$\mathcal{L} = \begin{cases} 0; & \text{if } diff + \alpha \leq 0 \\ diff + \alpha; & \text{if } diff + \alpha > 0 \end{cases}$$

Triplet Loss

$$\mathcal{L} = \begin{cases} 0; & \text{if } diff + \alpha \leq 0 \\ diff; & \text{if } diff + \alpha > 0 \end{cases}$$

Simplified

$$\mathcal{L}(A, P, N) = \max(diff + \alpha, 0)$$

From the neural network

- You can use any similarity function or distance metric

Triplet Selection

Triplet A, P, N { duplicate set: A, P; non-duplicate set: A, N

$$\mathcal{L} = \max(diff + \alpha, 0)$$

$$\text{diff} = \text{s}(A, N) - \text{s}(A, P)$$

Random

Easy to satisfy. Little to learn

$$\text{s}(A, N) \approx \text{s}(A, P)$$

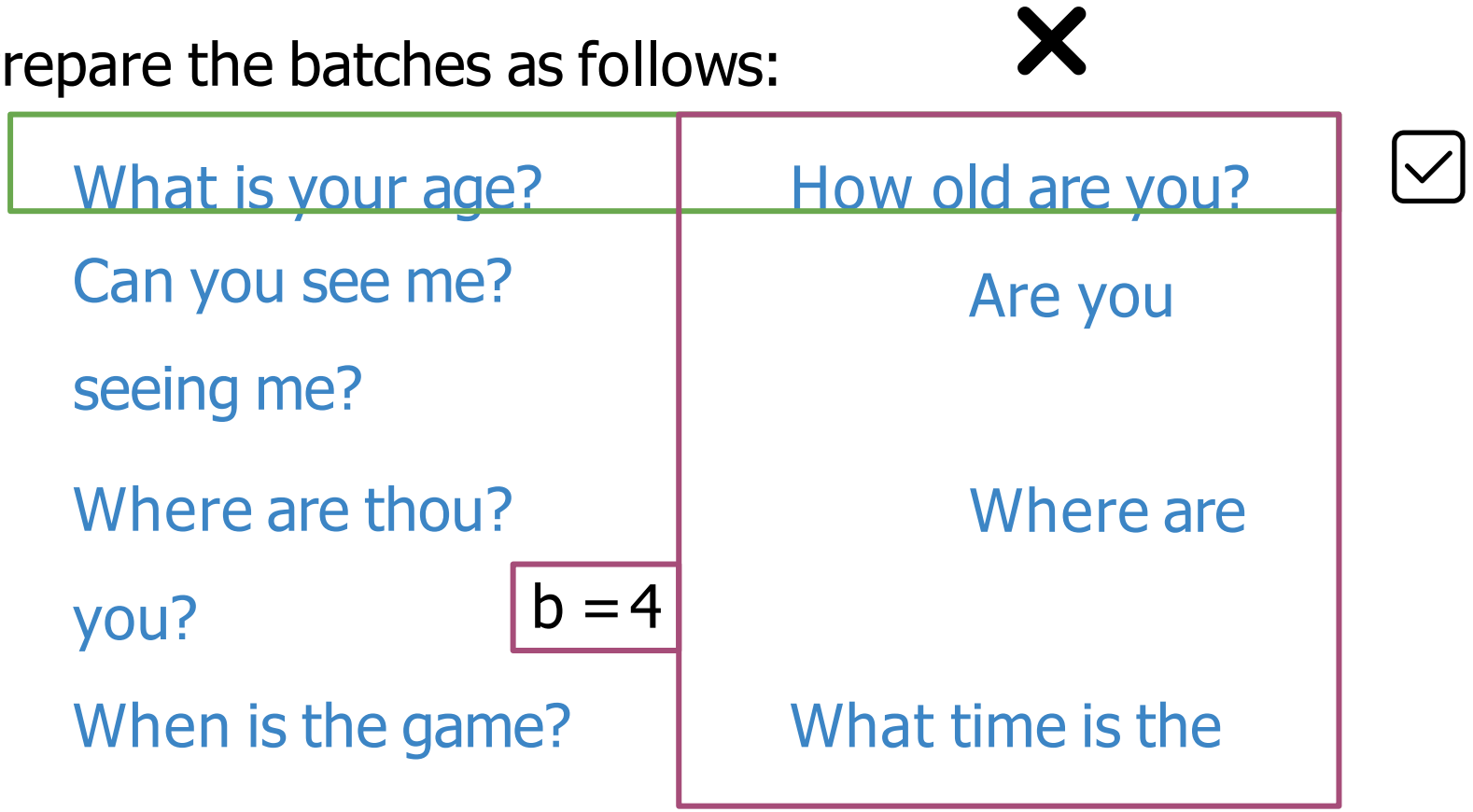
Hard

Harder to train. More to learn

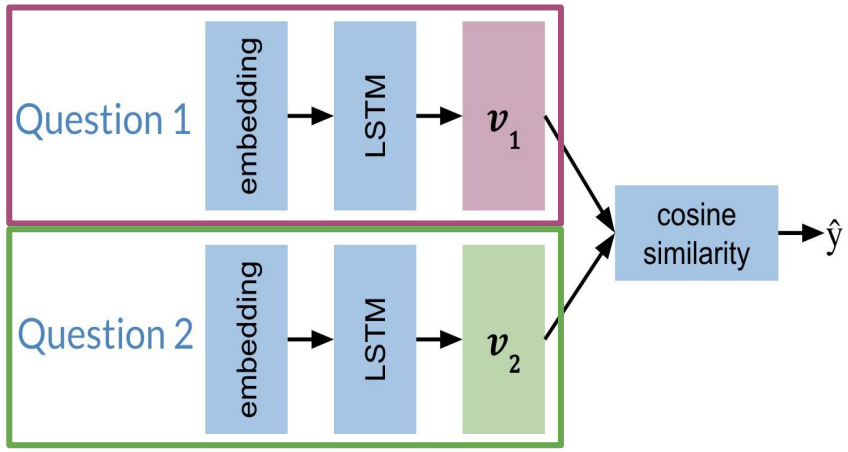
- select a pair of duplicates questions as the anchor and positive example from the training set.
- select a question that is known to be different in meaning from the anchor to form the anchor and the negative pair.

Computing The Cost I

Prepare the batches as follows:



Computing The Cost



Batch 1

What is your age?

Can you see me?

Where are thou?

When is the game?

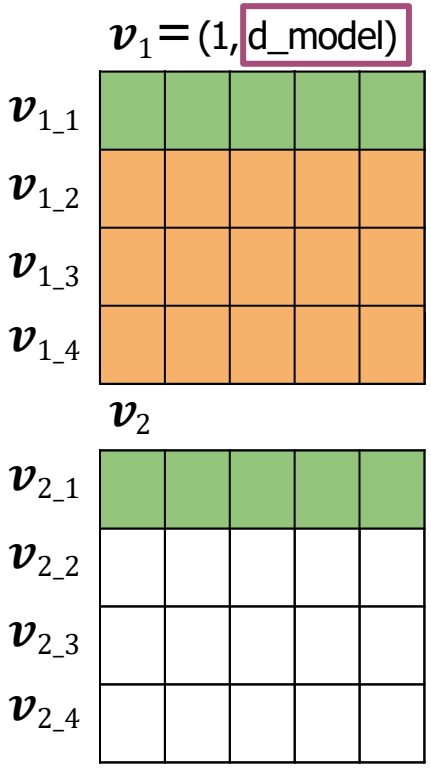
Batch 2

How old are you?

Are you seeing me?

Where are you?

What time is the game?



Computing The Cost

FPT Education
FPT UNIVERSITY

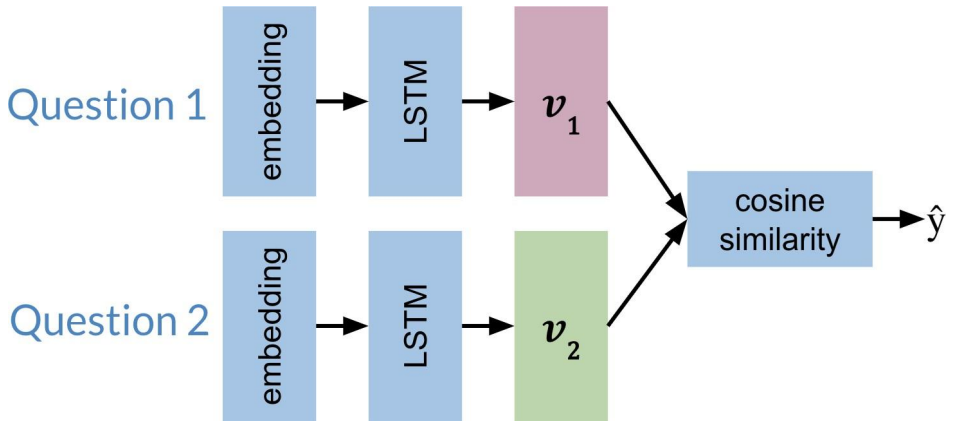
$s(v_1, v_2)$

v_2 \ v_1	_1	_2	_3	_4
_1	0.9	-0.8	0.3	-0.5
_2	-0.8	0.5	0.1	-0.2
_3	0.3	0.1	0.7	-0.8
_4	-0.5	-0.2	-0.8	1.0

$$\mathcal{L}(A, P, N) = \max(diff + \alpha, 0)$$
$$\text{diff} = s(A, N) - s(A, P)$$

$$\mathcal{J} = \sum_{i=1}^{m} \mathcal{L}(A^{(i)}, P^{(i)}, N^{(i)})$$

Computing The Cost II



Batch 1

What is your age?

Can you see me?

Where are thou?

When is the game?

Batch 2

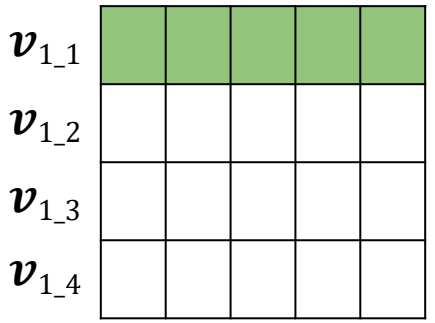
How old are you?

Are you seeing me?

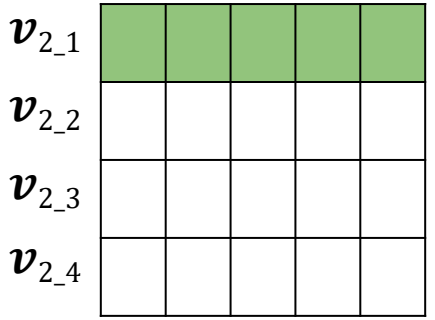
Where are you?

What time is the game?

v_1 = (1, d_model)



v_2



Hard Negative Mining

$s(v_1, v_2)$

v_2 \ v_1	_1	_2	_3	_4
_1	0.9	-0.8	0.3	-0.5
_2	-0.8	0.5	0.1	-0.2
_3	0.3	0.1	0.7	-0.8
_4	-0.5	-0.2	-0.8	1.0

mean negative:

mean of off-diagonal values in each row

closest negative:

off-diagonal value closest to (but less than) the value on diagonal in each row

Hard Negative Mining

mean negative: mean of off-diagonal values

closest negative: closest off-diagonal value

$$\mathcal{L}_{\text{Original}} = \max\left(\underbrace{s(A, N) - s(A, P)}_{\text{diff}} + \alpha, 0\right)$$

$$\mathcal{L}_1 = \max\left(mean\_neg - s(A, P) + \alpha, 0\right)$$

$$\mathcal{L}_2 = \max\left(closest\_neg - s(A, P) + \alpha, 0\right)$$

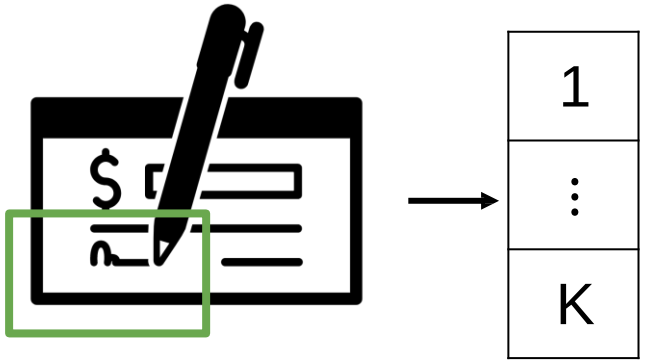
$$\mathcal{L}_{\text{Full}}(A, P, N) = \mathcal{L}_1 + \mathcal{L}_2$$

$$\mathcal{J} = \sum_{i=1}^{m} \mathcal{L}_{\text{Full}}(A^{(i)}, P^{(i)}, N^{(i)})$$

One Shot Learning

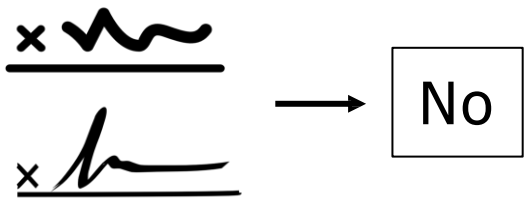
- Classification vs One Shot Learning

Classification



Classify as 1 of K classes

One Shot Learning



Measure similarity between 2 classes

One Shot Learning

FPT Education
FPT UNIVERSITY

No need for retraining !

Learn a similarity score!

$$s(sig1, sig2) > \tau \quad ☑$$

$$s(sig1, sig2) \leq \tau \quad ✗$$

Training / Testing

- Dataset

Question 1	Question 2	is_duplicate
What is your age?	How old are you?	true
Where are you from?	Where are you going?	false
⋮	⋮	⋮

Prepare Batches

Question 1:
batch size b

Batch 1

- What is your age?
- Can you see me?
- Where are thou?
- When is the game?

q_1_a
q_1_b

v_1 = (1, d_model)

v_{1_1}
v_{1_2}
v_{1_3}
v_{1_4}

Question 2:
batch size b

Batch 2

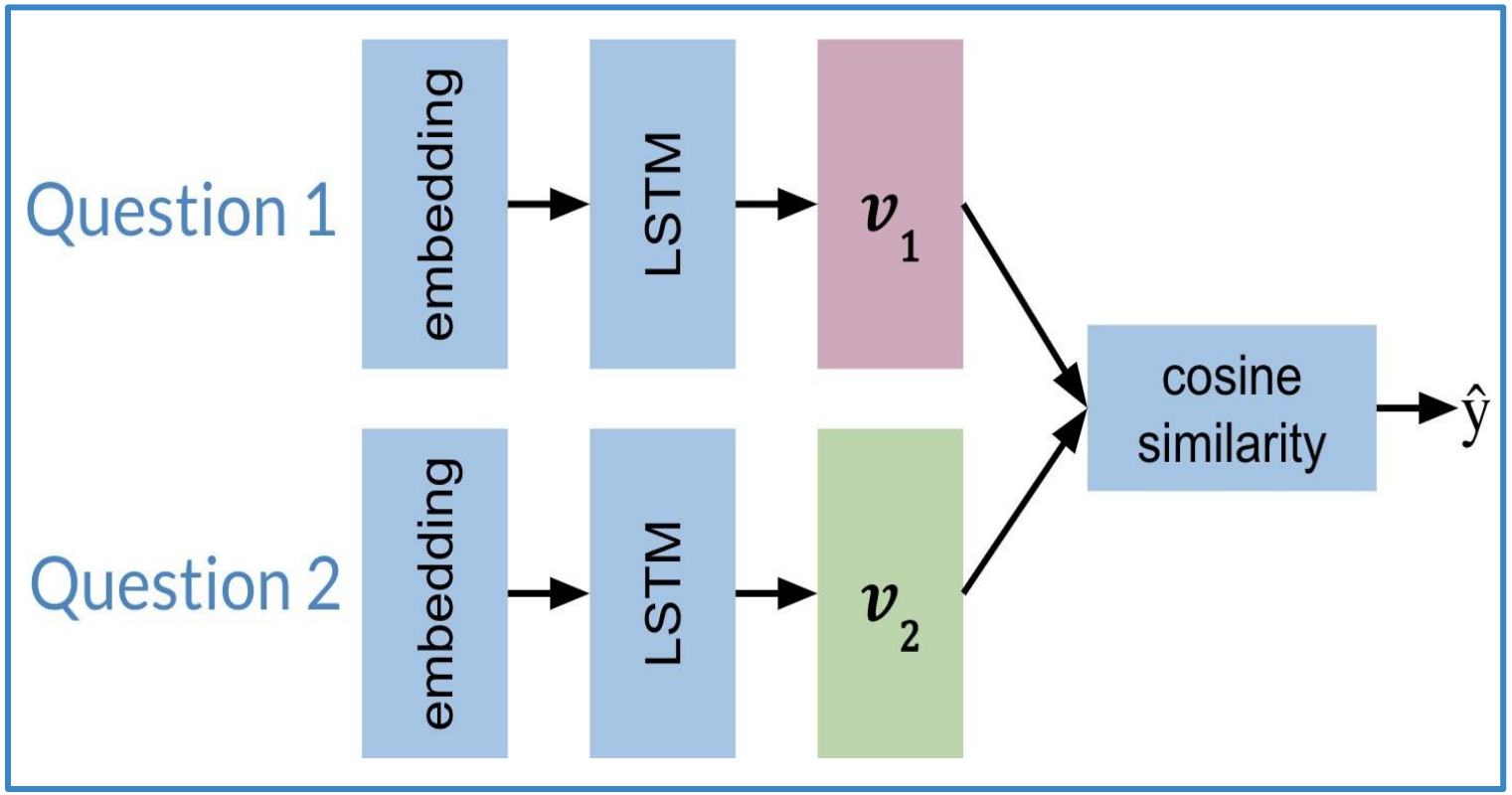
- How old are you?
- Are you seeing me?
- Where are you?
- What time is the game?

q_2_a
q_2_b

v_2

v_{2_1}
v_{2_2}
v_{2_3}
v_{2_4}

Siamese Model



Create a subnetwork:

1) Embedding
2) LSTM
3) Vectors
4) Cosine Similarity

Testing

- The goal is to find a similarity score between two inputs questions.
- Then test the score against some threshold τ and if the cosine similarity is greater than Tau, then the questions are classified as duplicates.

1. Convert each input into an array of numbers
2. Feed arrays into your model
3. Compare $\boldsymbol{v}_1, \boldsymbol{v}_2$ using cosine similarity
4. Test against a threshold τ