Neural Networks for Text Semantic Similarity Modeling

Zhe Hu

Outline

1. Introduction

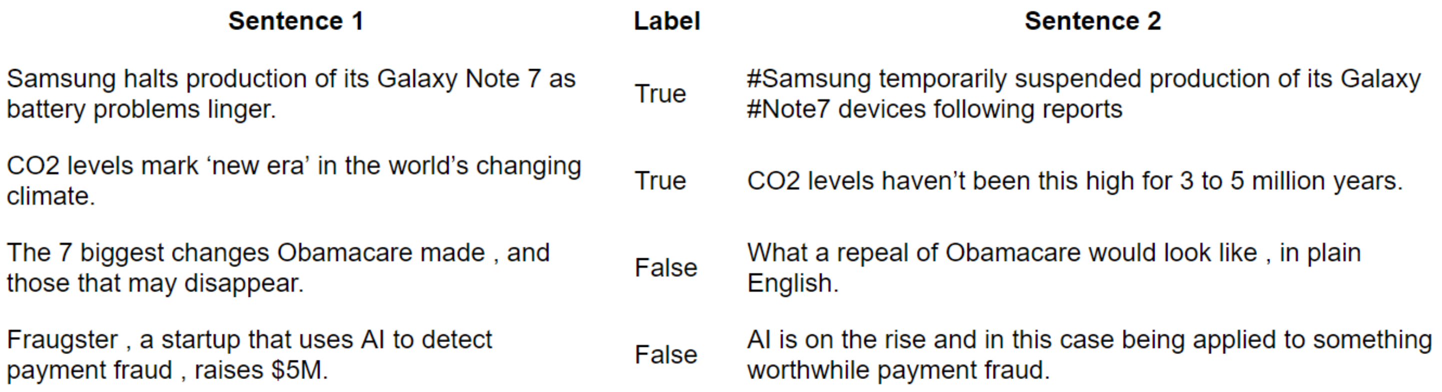
2. Models

3. Experiments

4. Analysis

1. Introduction

Given a sentence pair, we want to know where they have the same meaning or not?



Sentence 1	Label	Sentence 2
Samsung halts production of its Galaxy Note 7 as battery problems linger.	True	#Samsung temporarily suspended production of its Galaxy #Note7 devices following reports
CO2 levels mark ‘new era’ in the world’s changing climate.	True	CO2 levels haven’t been this high for 3 to 5 million years.
The 7 biggest changes Obamacare made , and those that may disappear.	False	What a repeal of Obamacare would look like , in plain English.
Fraugster , a startup that uses AI to detect payment fraud , raises $5M.	False	AI is on the rise and in this case being applied to something worthwhile payment fraud.

This task is useful in many NLP systems, like dialog system, question answering, etc.

Examples are taken from LanguageNet page: https://languagenet.github.io/

1. Introduction

- Naïve Way:

  If two sentences have many same words, they may have the same meaning.

However, in many cases it is not accurate.

1. What kind of fruit is this ? What kind of fruit do you eat ? -- not similar

2. What are some DIY ways to remove scratches from car ? How do I get scratches out of a vehicle ?
-- similar

- More Advanced Method:

  Let’s try NNs!

2. Models

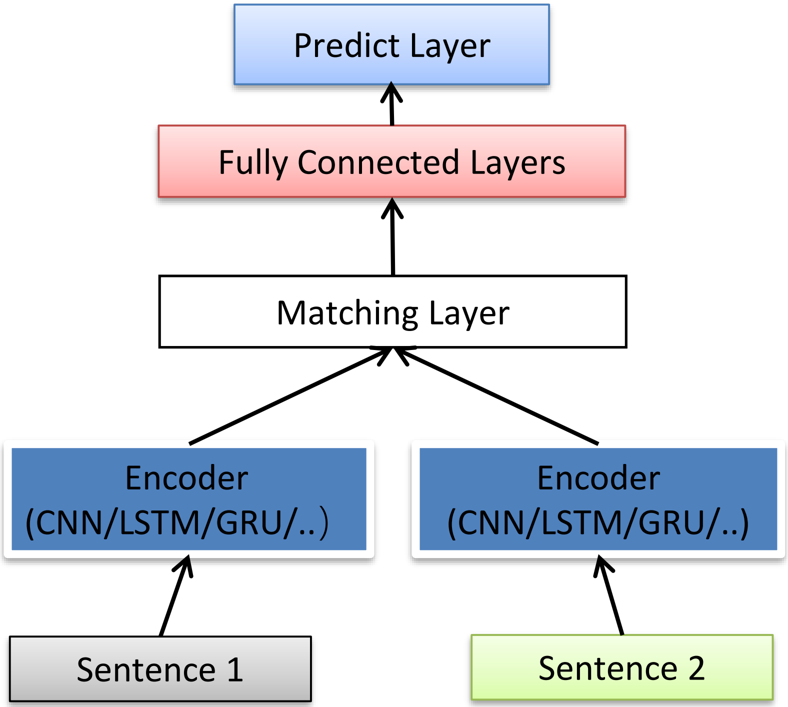
- Sentence Encoder Models

  Use neural nets to encoder each sentence into vector representation, and compare their semantic relationship.

- Sentence Interaction Models

  Calculate word-pair interactions to build the sentence interaction matrix, and use image recognition method to do classification.

Sentence Encoder Structure



Predict Layer

Fully Connected Layers

Matching Layer

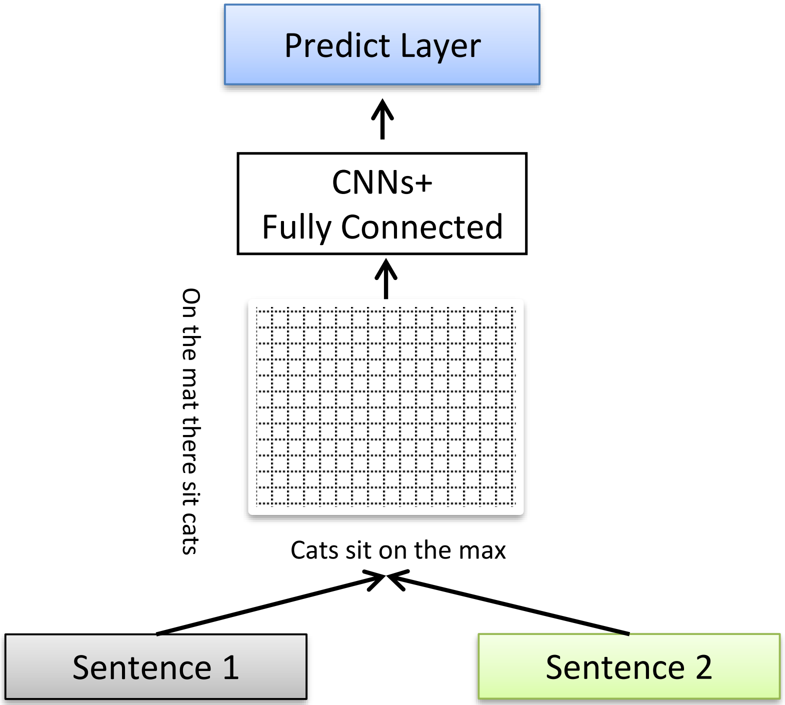
Encoder (CNN/LSTM/GRU/..)

Encoder (CNN/LSTM/GRU/..)

Sentence 1

Sentence 2

Sentence Interaction Structure



Predict Layer

CNNs+
Fully Connected

On the mat there sit cats

Cats sit on the max

Sentence 1

Sentence 2

Some state-of-art models for sentence pair modeling:

Model	structure
InferSent (Conneau, et al.)	BiLSTM encoder + row-max pooling
ESIM (Chen et al.)	BiLSTM encoder + soft alignment attention
MatchPyramid (Pang, et al.)	Interaction-based model with multi-CNN layers
……	……

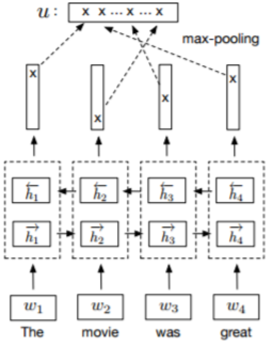


Figure 1. InferSent

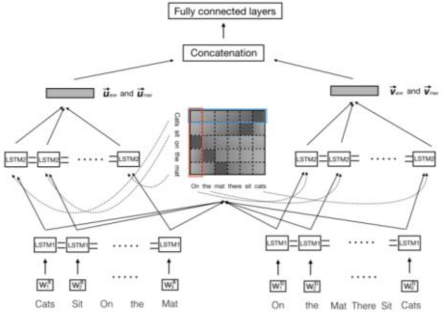
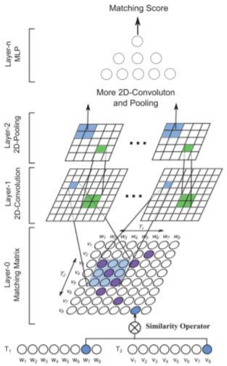


Figure 2. ESIM

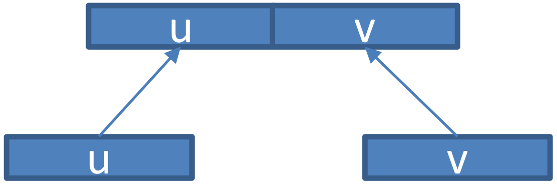


Matching Score
Layer-n MLP
More 2D-Convoluton and Pooling
Layer-2 2D-Pooling
Layer-1 2D-Convolution
Layer-0 Matching Matrix
T_1
T_2
Similarity Operator
T_1 w_1 w_2 w_3 w_4 w_5 w_6 w_7 w_8
T_2 v_1 v_2 v_3 v_4 v_5 v_6 v_7 v_8

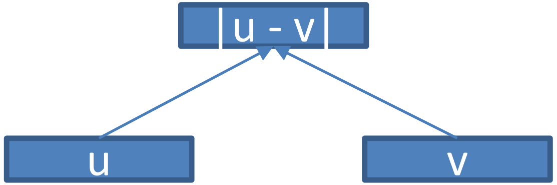
Figure 3. MatchPyramid

Note: figure1 and figure3 are from the original papers, and figure2 is from paper (Lan, et al.)

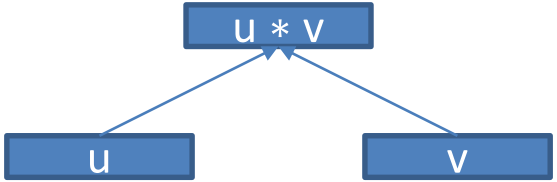
Matching layer



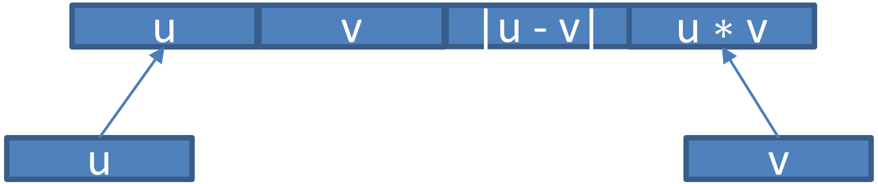
Matching 1



Matching 2

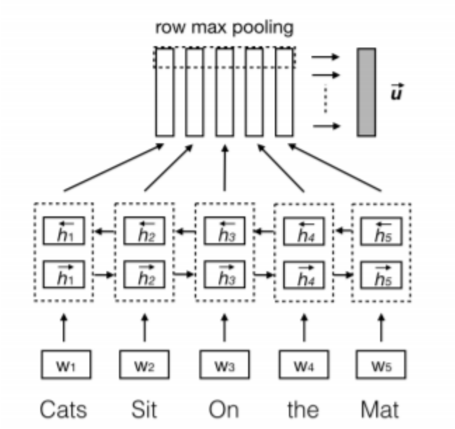


Matching 3



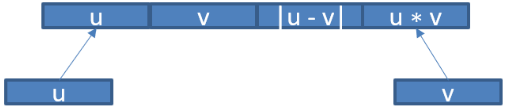
Matching 4: the best one

Model 1: InferSent


row max pooling
$\vec{u}$
$\overleftarrow{h_1}$ $\overleftarrow{h_2}$ $\overleftarrow{h_3}$ $\overleftarrow{h_4}$ $\overleftarrow{h_5}$
$\overrightarrow{h_1}$ $\overrightarrow{h_2}$ $\overrightarrow{h_3}$ $\overrightarrow{h_4}$ $\overrightarrow{h_5}$
w_1 w_2 w_3 w_4 w_5
Cats Sit On the Mat

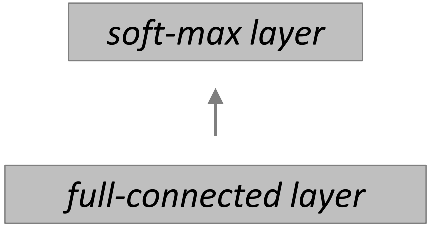
encoder

+



matching

+



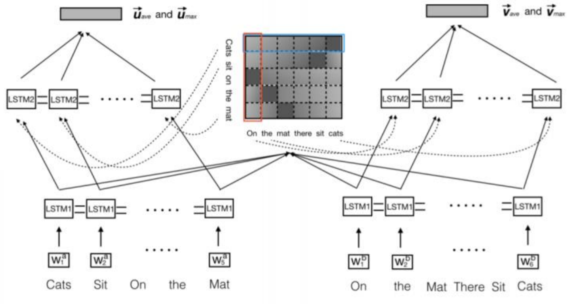
soft-max layer

full-connected layer

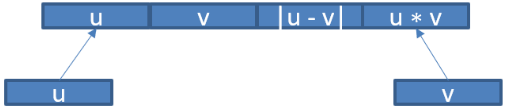
classifier

Note: figure for encoder is from paper (Lan, et al.)

Model 2: ESIM (Enhanced LSTM for Natural Language Inference)


$\vec{u}_{ave}$ and $\vec{u}_{max}$
$\vec{v}_{ave}$ and $\vec{v}_{max}$
LSTM2
LSTM1
Cats sit on the mat
On the mat there sit cats
W^a_1 W^a_2 W^a_5
Cats Sit On the Mat
W^b_1 W^b_2 W^b_6
On the Mat There Sit Cats

+

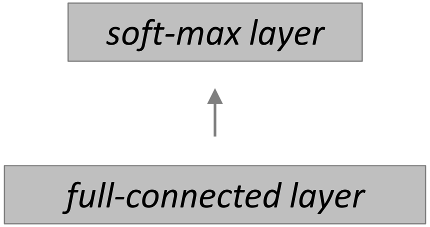


u | v | |u - v| | u * v

u

v

+



soft-max layer

full-connected layer

Encoder with soft-alignment attention

matching

classifier

Note: figure for encoder is from paper (Lan, et al.)

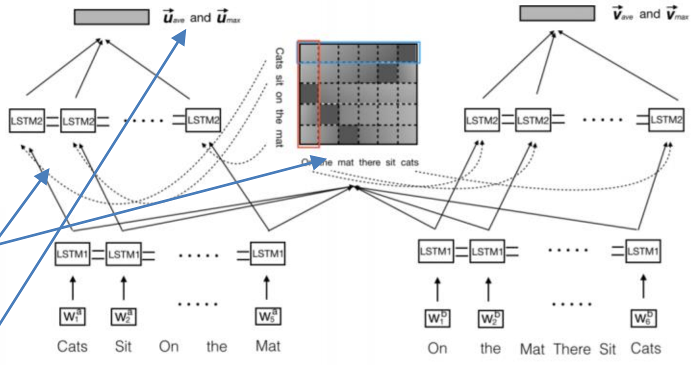
Model 2: ESIM

soft alignment attention:

1. $e_{ij} = \bar{a}_i^T \bar{b}_j$

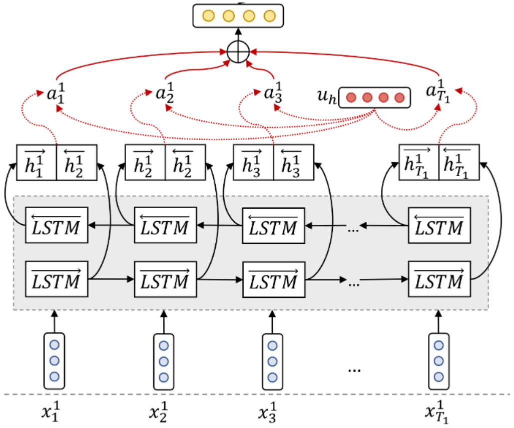
2.
$$\tilde{\mathbf{a}}_i = \sum_{j=1}^{\ell_b} \frac{\exp(e_{ij})}{\sum_{k=1}^{\ell_b} \exp(e_{ik})} \bar{\mathbf{b}}_j, \forall i \in [1, \ldots, \ell_a],$$
$$\tilde{\mathbf{b}}_j = \sum_{i=1}^{\ell_a} \frac{\exp(e_{ij})}{\sum_{k=1}^{\ell_a} \exp(e_{kj})} \bar{\mathbf{a}}_i, \forall j \in [1, \ldots, \ell_b],$$

3.
$$\mathbf{m}_a = [\bar{\mathbf{a}}; \tilde{\mathbf{a}}; \bar{\mathbf{a}} - \tilde{\mathbf{a}}; \bar{\mathbf{a}} \odot \tilde{\mathbf{a}}],$$
$$\mathbf{m}_b = [\bar{\mathbf{b}}; \tilde{\mathbf{b}}; \bar{\mathbf{b}} - \tilde{\mathbf{b}}; \bar{\mathbf{b}} \odot \tilde{\mathbf{b}}].$$


$\vec{u}_{ave}$ and $\vec{u}_{max}$
$\vec{v}_{ave}$ and $\vec{v}_{max}$
Cats sit on the mat
On the mat there sit cats
LSTM2
LSTM1
W_1^a
W_2^a
W_5^a
W_1^b
W_2^b
W_5^b
Cats Sit On the Mat
On the Mat There Sit Cats

Note: figure for encoder is from paper (Lan, et al.)

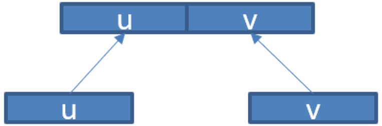
Model 3: Siamese LSTM with self-attention(self-attentive lstm)



a_1^1
a_2^1
a_3^1
u_h
$a_{T_1}^1$
$\overrightarrow{h_1^1}$ $\overleftarrow{h_1^1}$
$\overrightarrow{h_2^1}$ $\overleftarrow{h_2^1}$
$\overrightarrow{h_3^1}$ $\overleftarrow{h_3^1}$
$\overrightarrow{h_{T_1}^1}$ $\overleftarrow{h_{T_1}^1}$
$\overleftarrow{LSTM}$
$\overrightarrow{LSTM}$
...
x_1^1
x_2^1
x_3^1
$x_{T_1}^1$

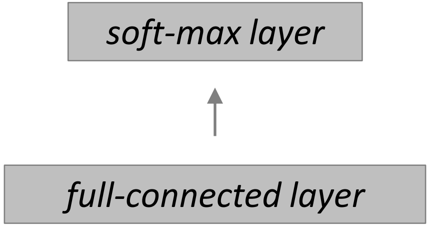
Encoder with self-attention

+



matching

+



classifier

Note: figure for encoder is from Baziotis, etc. (SemEval-2017). 2017.

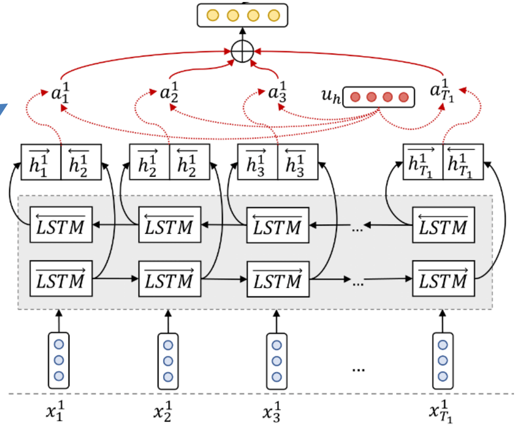
Model 3: Siamese LSTM with self-attention

Context attention layer (Yang et al., 2016):

$$e_i = tanh(W_h h_i + b_h), \quad e_i \in [-1, 1]$$

$$a_i = \frac{exp(e_i^\top u_h)}{\sum_{t=1}^{T} exp(e_t^\top u_h)}, \quad \sum_{i=1}^{T} a_i = 1$$

$$r = \sum_{i=1}^{T} a_i h_i, \quad r \in R^{2L}$$



a_1^1
a_2^1
a_3^1
u_h
$a_{T_1}^1$
$\overrightarrow{h_1^1}$ $\overleftarrow{h_1^1}$
$\overrightarrow{h_2^1}$ $\overleftarrow{h_2^1}$
$\overrightarrow{h_3^1}$ $\overleftarrow{h_3^1}$
$\overrightarrow{h_{T_1}^1}$ $\overleftarrow{h_{T_1}^1}$
$\overleftarrow{LSTM}$
$\overrightarrow{LSTM}$
...
x_1^1
x_2^1
x_3^1
$x_{T_1}^1$

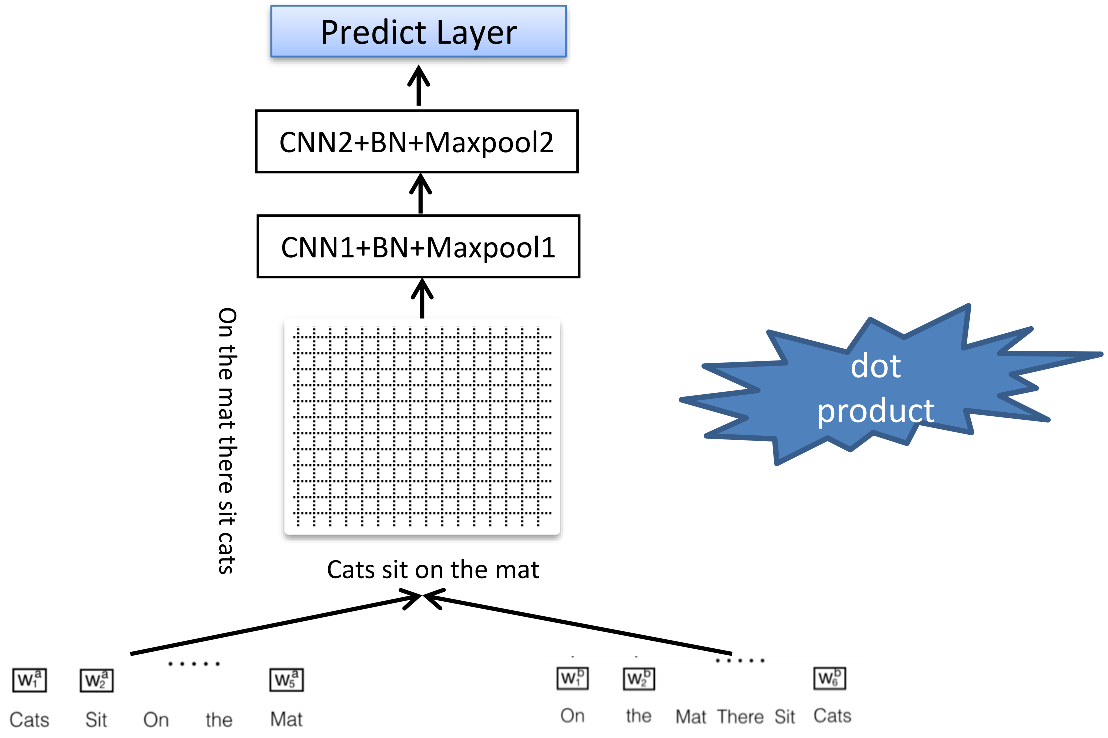
Encoder with self-attention (context-attention)

Interpretation: use fixed query u_h, that helps to identify the informative words in this sentence.

Note: figure for encoder is from Baziotis, etc. (SemEval-2017). 2017.

Model 4: MatchPyramid

Using Glove to embed words into vector space.



Predict Layer

CNN2+BN+Maxpool2

CNN1+BN+Maxpool1

On the mat there sit cats

Cats sit on the mat

dot product

w_1^a w_2^a ..... w_5^a

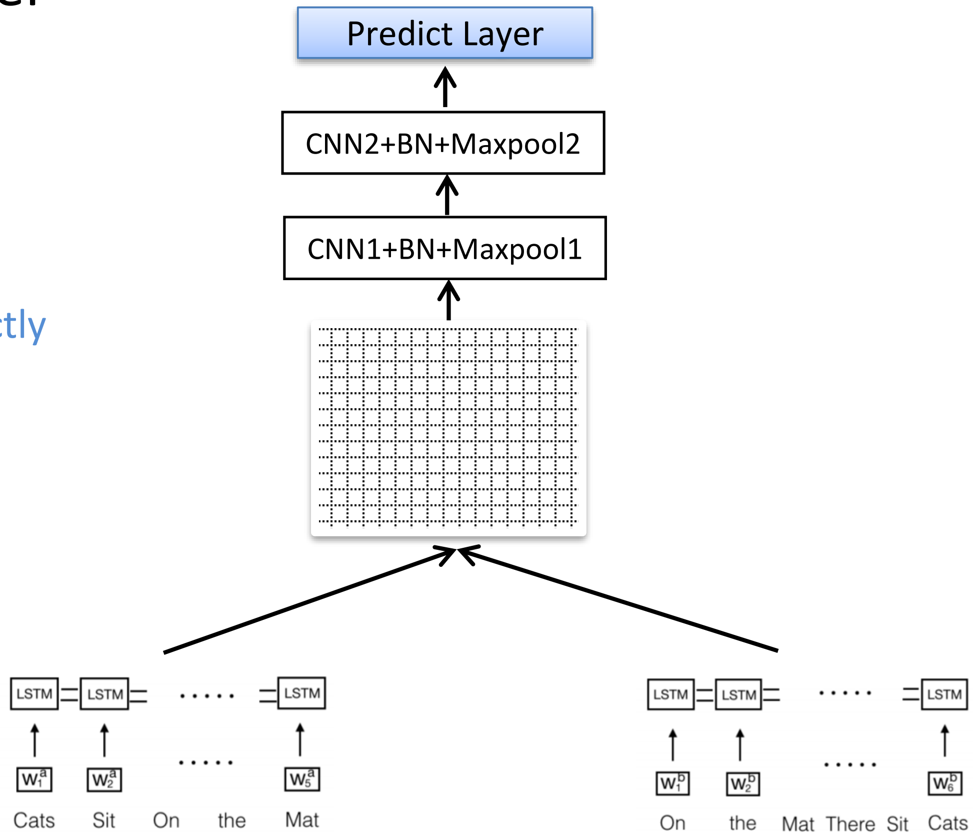
Cats Sit On the Mat

w_1^b w_2^b .... w_6^b

On the Mat There Sit Cats

Model 5: Match Encoder

Instead of computing interaction directly after embedding, use encoder to aggregate some context information first.



3. Experiment & Results

Dataset

- **Quora Question Pairs (Iyer et al., 2017):** Contains 400k question pairs collected from the Quora website, with positive and negative labels indicating whether they are paraphrase or not.

POS	*What should I do to avoid sleeping in class ?*	*How do I not sleep in a boring class ?*
NEG	*Do women support each other more than men do ?*	*Do women need more compliments than men ?*

- **Twitter-URL (Lan et al., 2017):** includes 50k sentence pairs collected from tweets that share the same URL of news articles. This dataset contains both formal and informal language.

NEG	*Are you open to 2020 ? I am open to doing everything I can right now.*	*From one Senator to the next .*
POS	*Letter warned Wells Fargo of " widespread " fraud in 2007.*	*Letters suggest Wells Fargo scandal started earlier.*

3. Experiment & Results

Implementation details:

- Using Tensorflow and Keras
- Using Glove for embedding
- Batch size: 64
- Epoch: 30
- Optimizer: Adam

3. Experiment & Results

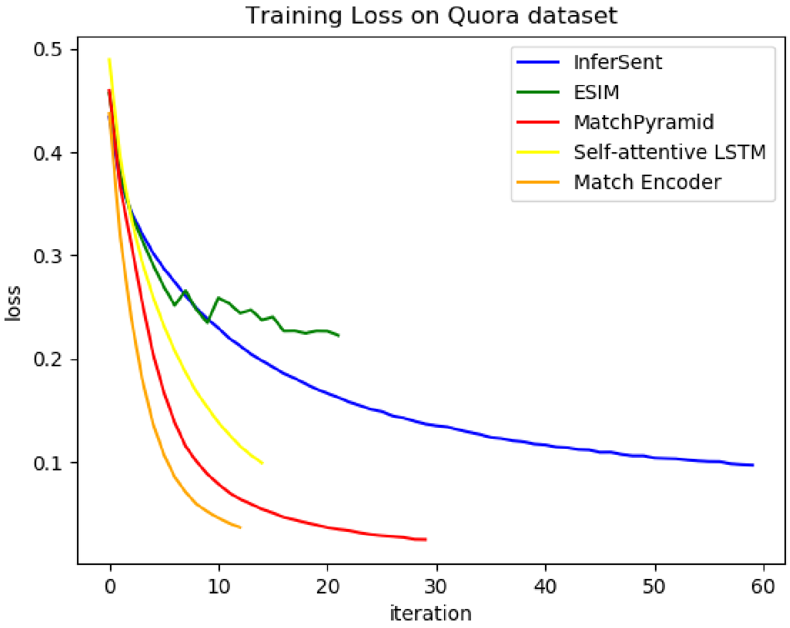
train:384,348
validation: 10,000

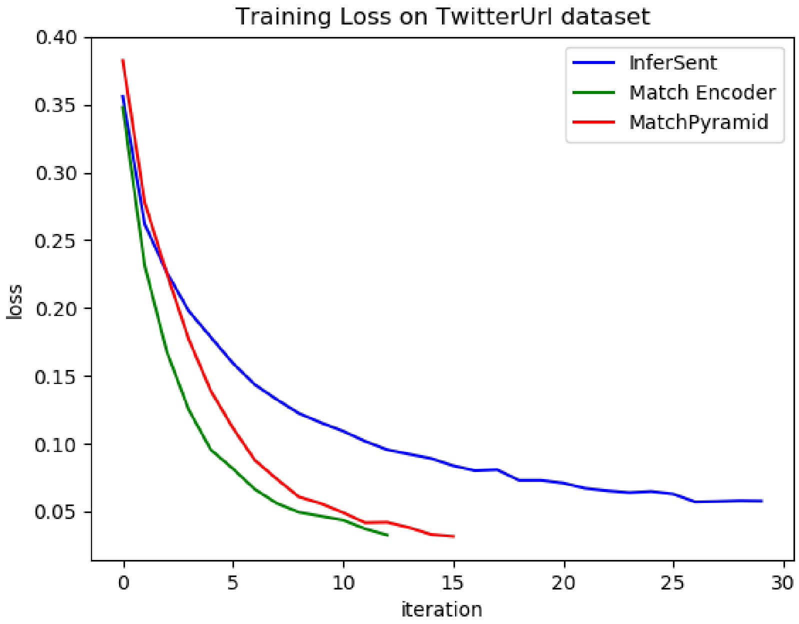
train:42,200
validation: 9,334

Model	Quora (acc)	Twitter URL (f1)
InferSent	86.93%	68.90%
ESIM	87.06%	71.23%
MatchPyramid	83.60%	69.87%
Match Encoder	85.88%	68.89%
self-attentive lstm	82.89%	--

Note: results of self-attentive lstm is using Glove.6B.100D, which gives the lower acc than Glove.840B.300D.

3. Experiment & Results





4. Summery & Conclusion

- Paraphrase identification tasks ;
- Quora Question Pairs dataset & TwitterURL dataset;
- Different sentence-encoder models & sentence-interaction models.

4. Summery & Conclusion

Some Conclusion:

- **MatchPyramid** does not work well on Quora dataset. We think this comes from high frequency of overlapping words (> 40%) between sentence pairs. However, **MatchEncoder** performs better because of aggregating of context information.

- Glove.840B.300D(840B tokens, 2.2M vocab, 300d vectors) brings better results than Glove.6B.100D(6B tokens, 400K vocab, 100d) for embedding. We think word vectors trained on large corpus will definitely be more generic to be transferred on other tasks.

- Self attention does not help a lot on sentence pair modelling task, which fails to capture the relationship between the sentence pairs in our consideration. On the contrary, interacted attention give better performance.

5. Future Work

- Combinations of two different types of models;
- Some other attention mechanism;
- Other sentence pair modelling tasks, like NLI.
- ……

Thank you !