



Upgrade

Open in app



Published in Towards Data Science · Following ▾



Edward Ma · Follow

Oct 13, 2018 · 5 min read · Listen



Learning sentence embeddings by Natural Language Inference

Unsupervised learning approach seems like a normal way to build word, sentence or document embeddings because it is more generalized such that pre-trained embeddings result can be transfer to other NLP downstream problems. For example, skip-gram in word embeddings and skip-thought in sentence embeddings and distributed bag-of-words in paragraph embeddings.



“closeup photo of person carrying professional video recorder” by [Laura Lee Moreau](#) on [Unsplash](#)

Conneau et al. noted that supervised learning in ImageNet (Image Classification) doing good job in transferring result to downstream problems. Some features can be transferred to downstream somehow. Therefore, Conneau et al. used textual entailment data to train a sentence embeddings layer which calls InferSent.

After reading this article, you will understand:

- InferSent Design
- Architecture
- Implementation
- Take Away

InferSent Design





Upgrade

Open in app



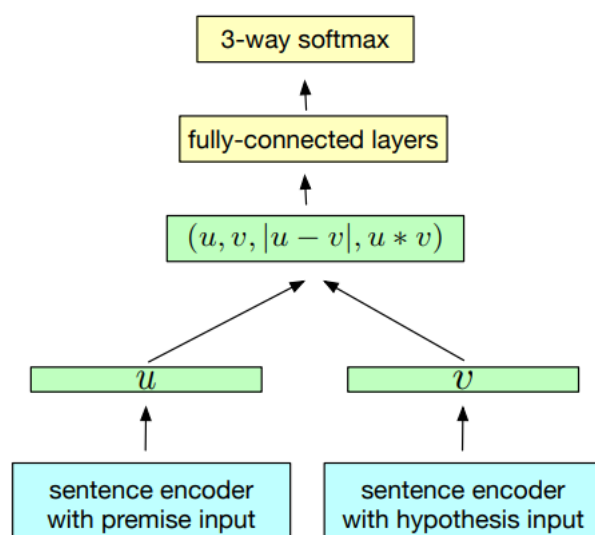
"two apples and walnuts on white towel" by [Alex Kotomanov](#) on [Unsplash](#)

1. I eat fruit.
2. I eat apple.

Intuitively, the relationship is entailment. Authors believe that NLI is a suitable task to understand semantic relationships within sentences such that it helps to build a good embeddings for sentence embeddings for downstream NLP problems.

Architecture

The overall idea is that two sentences (premise input and hypothesis input) will be transformed by sentence encoder (same weights). After that leveraging 3 matching methods to recognize relations between premise input and hypothesis input.





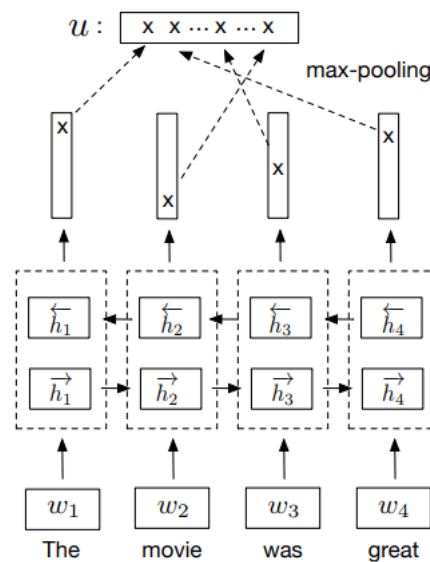
2. Element-wise product two vectors

3. Absolute element-wise difference of two vectors

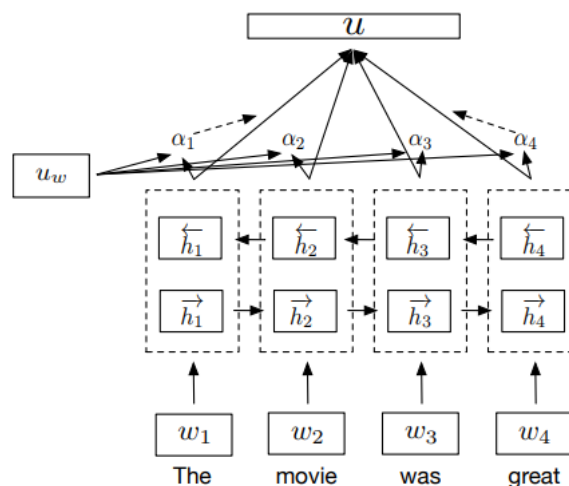
After the overview, may jump into the architecture of sentence encoders. Conneau et al. evaluated 7 different architectures:

1. Standard LSTM
2. Standard GRU
3. Concatenation of last hidden states of forward and backward GRU
4. Bi-directional LSTM with mean polling
5. Bi-directional LSTM with max polling
6. Self-attentive Network (Attention with BiLSTM)
7. Hierarchical convolutional networks

Before conclude the best approach first, we may believe that Attention with BiLSTM should be the best approach as attention mechanism helps to identify important weight. Actually, it may harm when using it in transfer learning. On the other hand, BiLSTM with mean polling perform not very good may due to unable to locate the important part.



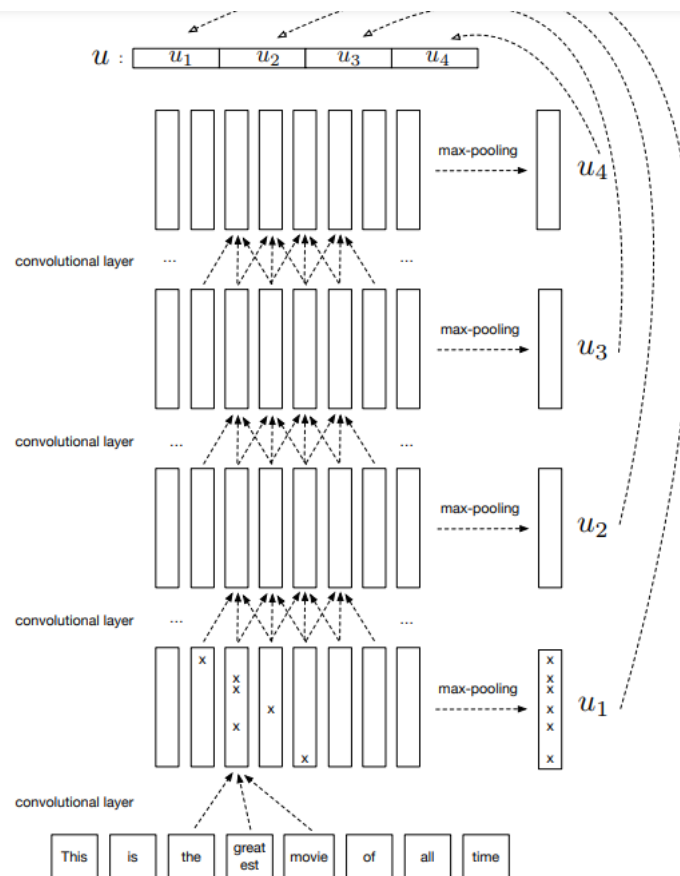
#5 Bi-directional LSTM with max polling (Conneau et al., 2017)





Upgrade

Open in app



Hierarchical convolutional networks (Conneau et al., 2017)

From the experiment result, the best approach is Bi-directional LSTM with max polling.

Model	dim	NLI		Transfer	
		dev	test	micro	macro
LSTM	2048	81.9	80.7	79.5	78.6
GRU	4096	82.4	81.8	81.7	80.9
BiGRU-last	4096	81.3	80.9	82.9	81.7
BiLSTM-Mean	4096	79.0	78.2	83.1	81.7
Inner-attention	4096	82.3	82.5	82.1	81.0
HConvNet	4096	83.7	83.4	82.0	80.9
BiLSTM-Max	4096	85.0	84.5	85.2	83.7

Conneau et al. (2017)

Implementation

There are 2 ways to use InferSent. First of is using a pre-trained embeddings layer in your NLP problems. Another one is building InferSent by your self.

Load pre-trained Embeddings

Facebook research team provide 2 pre-trained models which are version 1 (based on GloVe) and version 2 (based on fastText).

Loading both InferSent pre-trained model and GloVe (or fastText) model than you can encode sentence to vectors.

```
# Init InferSent Model
infer_sent_model = InferSent()
infer_sent_model.load_state_dict(torch.load(dest_dir + dest_file))
```





Upgrade

Open in app

```
# Encode sentence to vectors
model.encode(sentences, tokenize=True)
```

Train embeddings

Another approach is training the embeddings by your self. You may either using your own data or using original data set. Since this InferSent uses supervised learning approach to generate sentence embeddings, you need to have a annotated (labeled) data first.

Here is the step of going first approach. Clone InferSent original [repo](#) to local. Then Execute “get_data.bash” in console such that SNLI (Stanford Natural Language Inference) and MultiNLI (MultiGenre NLI) corpus will be downloaded and processed. Make sure that you have to execute the following shell script in current folder but not other relative path

```
./get_data.bash
```

After that, downloading GloVe (and/or fastText)

```
mkdir dataset/GloVe
curl -Lo dataset/GloVe/glove.840B.300d.zip http://nlp.stanford.edu/data/glove.840B.300d.zip
unzip dataset/GloVe/glove.840B.300d.zip -d dataset/GloVe/
mkdir dataset/fastText
curl -Lo dataset/fastText/crawl-300d-2M.vec.zip https://s3-us-west-1.amazonaws.com/fasttext-vectors/crawl-300d-2M.vec.zip
unzip dataset/fastText/crawl-300d-2M.vec.zip -d dataset/fastText/
```

Downloading InferSent pre-trained model. Version 1 is trained by using GloVe while Version 2 leveraged fastText.

```
curl -Lo encoder/infersent1.pkl https://s3.amazonaws.com/senteval/infersent/infersent1.pkl
curl -Lo encoder/infersent2.pkl https://s3.amazonaws.com/senteval/infersent/infersent2.pkl
```

Finally, you can execute the following command to train the embeddings layers.

```
python train_nli.py --word_emb_path ./glove.42B.300d.txt
```

For my single GPU VM, it takes about 1 day to finish the training.

Take Away

To access all code, you can visit my [github](#) repo.

- Compare to other embedding approaches, **InferSent uses a supervised learning** to compute word vectors.
- **InferSent leverages word embeddings** (GloVe/ fastText) to build sentence embeddings.
- Pretrained model supports both GloVe (version 1) and fasttext (version 2)

About Me

I am Data Scientist in Bay Area. Focusing on state-of-the-art in Data Science, Artificial Intelligence , especially in NLP and platform related. You can reach me from [Medium Blog](#), [LinkedIn](#) or [Github](#).

Reference

Conneau, D. Kiela, H. Schwenk, L. Barrault, A. Bordes, *[Supervised Learning of Universal Sentence Representations from Natural Language Inference Data](#)*





Upgrade

Open in app

Sign up for The Variable

By Towards Data Science

Every Thursday, the Variable delivers the very best of Towards Data Science: from hands-on tutorials and cutting-edge research to original features you don't want to miss. [Take a look.](#)

Get this newsletter

Emails will be sent to ammaarahmad1999@gmail.com.
[Not you?](#)

