Identifying Personal Medication Intake from Twitter

Giray Eryılmaz
Mehmet Akif Yücel

Recall : The task

Mission: Monitoring drug consumption.

Data: Tweets mentioning some drug names.

Problem: Not all tweets actually indicate consumption, there are 3 types;

1. Actual intake: *Still don't feel good n I took some Tylenol but that ain't help*
2. Ambiguous: *I am going to need a Xanax after this*
3. False alarm: *Spongebob on steroids <inappropriate slang>*

Because of twitter, we could not get all the data, some were missing. We had 8336 for training and 6126 for testing. In the baseline they had 9663 to 7513.

The approach to the problem:

This is a ternary classification. We have talked about 3 methods in the proposal.

RNN based methods: BiLSTM with word vectors.

Transfer learning: Fine tuning BERT.

HMMs: Further investigation revealed that they are not favorable to RNNs in our context. But instead:

NER; Subject - Object - Verb analysis.

Word embeddings for twitter was optional, did it as well.

NER ; Subject, Object, Verb Extraction Analysis

I could extract these from 5462 of the training data, 2874 did not give results which makes about 65% yield.

I made experiments within these extracted ones trying to see if I can make use of these to boost performance. But the recall on the 1st category was quite low, about 20%, which is arguably the most important category. So I did not dig much deeper on this having seen the results of other methods.

More results can be found on the submitted notebook on this.

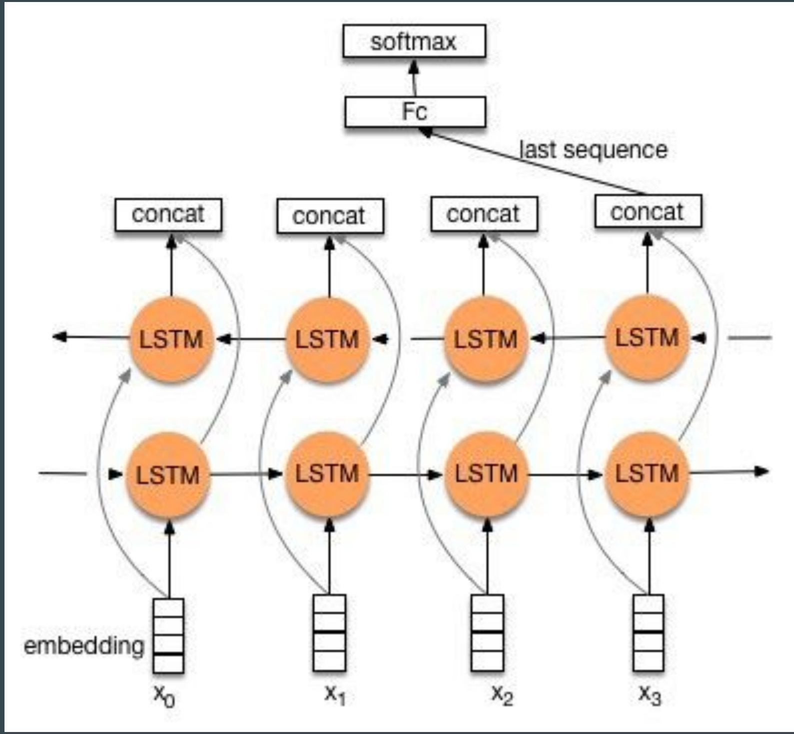
BiLSTM

I have tried different word embeddings from gensim:
glove-twitter-200,
fasttext-wiki-news-subwords-300
Glove-wiki-gigaword-300,
Word2vec-google-news-300

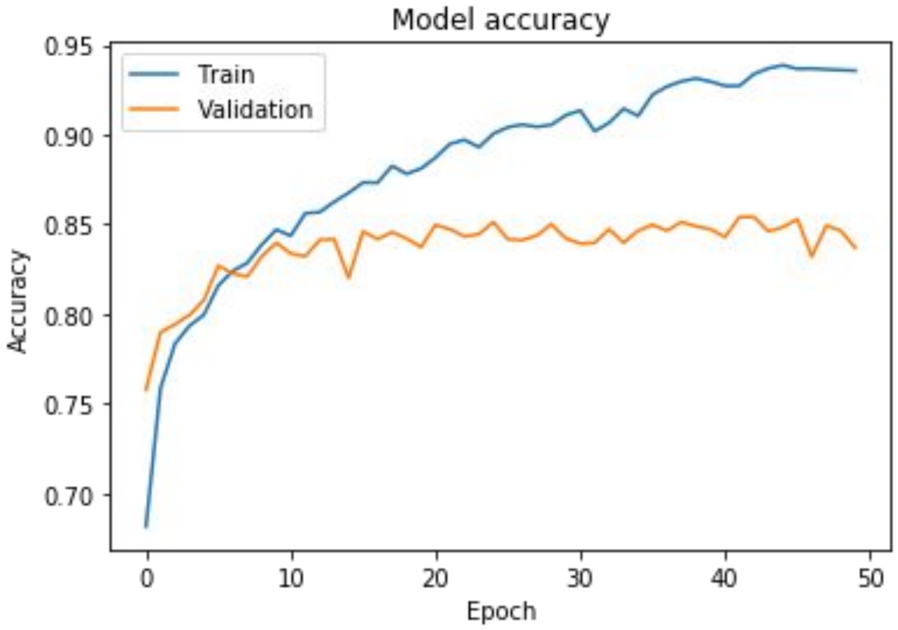
I have observed that the ones trained on twitter performed better despite smaller feature space.

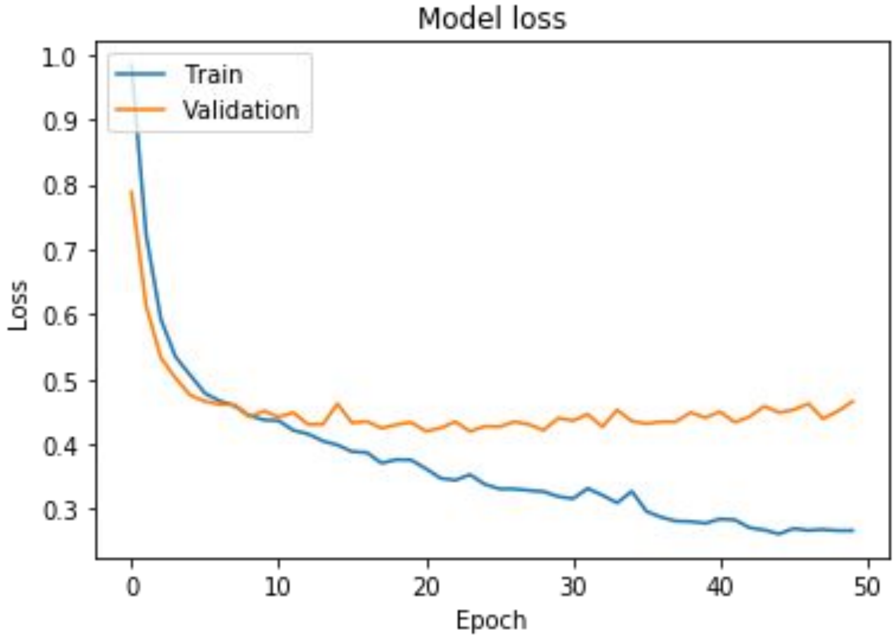
Bidirectional part did not have much over regular lstm.

Achieved 70% accuracy.



BiLSTM


Model accuracy
Train
Validation
Accuracy
Epoch
0.95
0.90
0.85
0.80
0.75
0.70
0
10
20
30
40
50


Model loss
Train
Validation
Loss
Epoch
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0
10
20
30
40
50

BERT Fine Tuning

Bert is a pre-trained (language) model that can be fined to do all kinds of nlp tasks.

The fine tuned Bert out performed the baseline paper in all metrics for all categories (except 1). Achieved 77% accuracy.

	Recall			Prescision			F1		
	1	2	3	1	2	3	1	2	3
BERT	0.7906	0.6608	0.8710	0.6963	0.7713	0.8297	0.7404	0.7118	0.8499
BiLSTM	0.6430	0.5914	0.8312	0.6458	0.6827	0.7423	0.6444	0.6338	0.7842
Baseline	0.6900	0.6480	0.8530	0.7120	0.7330	0.7610	0.7010	0.6880	0.8040

You can find more detailed results in the notebook submitted.

Final remarks & feature work

I believe, it would be very educational to investigate what ***exactly*** BERT has learnt.

The sentence analysis did not provide very informative results, but note that the parser was not trained on twitter data. I still believe that a fine tuned parser may provide good results. But there is possibility that BERT has already done that implicitly.

Fine-pre-tuning bert language model on twitter can increase performance, judging from the fact that twitter word vectors provided better performance for lstm.

More details and results, data analysis and preprocessing details can be found in the submitted ipynbs.

References

- Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).
- Ceshine Lee, 2017, Understanding Bidirectional RNN (only the image)
- Textacy, https://chartbeat-labs.github.io/textacy/