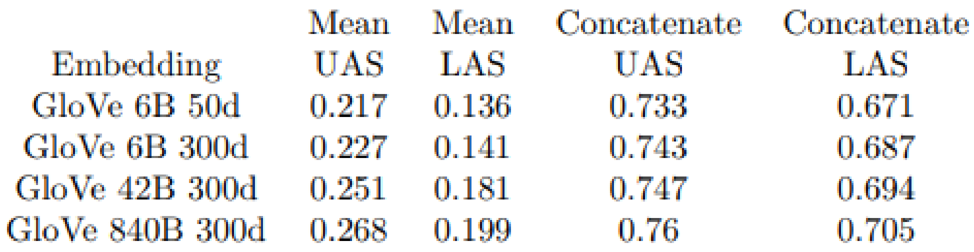
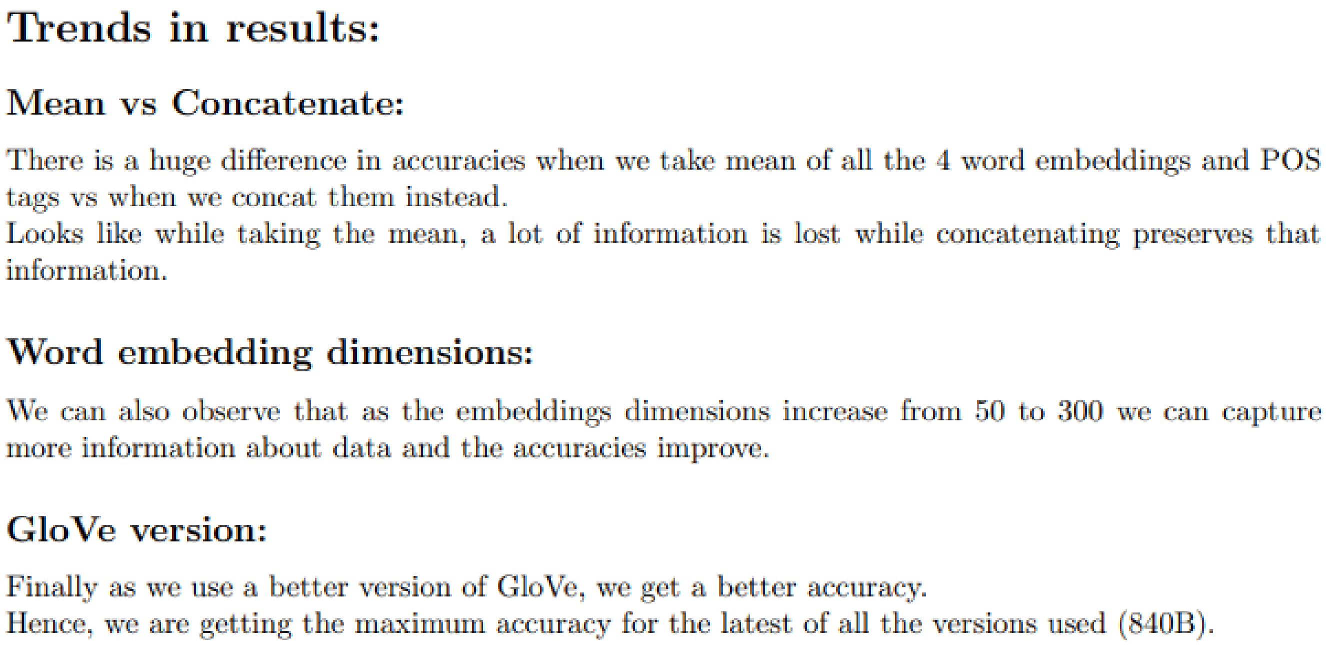
Q2) Results Table:

The following results are on learning rate 0.0001.



Embedding	Mean UAS	Mean LAS	Concatenate UAS	Concatenate LAS
GloVe 6B 50d	0.217	0.136	0.733	0.671
GloVe 6B 300d	0.227	0.141	0.743	0.687
GloVe 42B 300d	0.251	0.181	0.747	0.694
GloVe 840B 300d	0.268	0.199	0.76	0.705

Q3)



Trends in results:

Mean vs Concatenate:

There is a huge difference in accuracies when we take mean of all the 4 word embeddings and POS tags vs when we concat them instead.
Looks like while taking the mean, a lot of information is lost while concatenating preserves that information.

Word embedding dimensions:

We can also observe that as the embeddings dimensions increase from 50 to 300 we can capture more information about data and the accuracies improve.

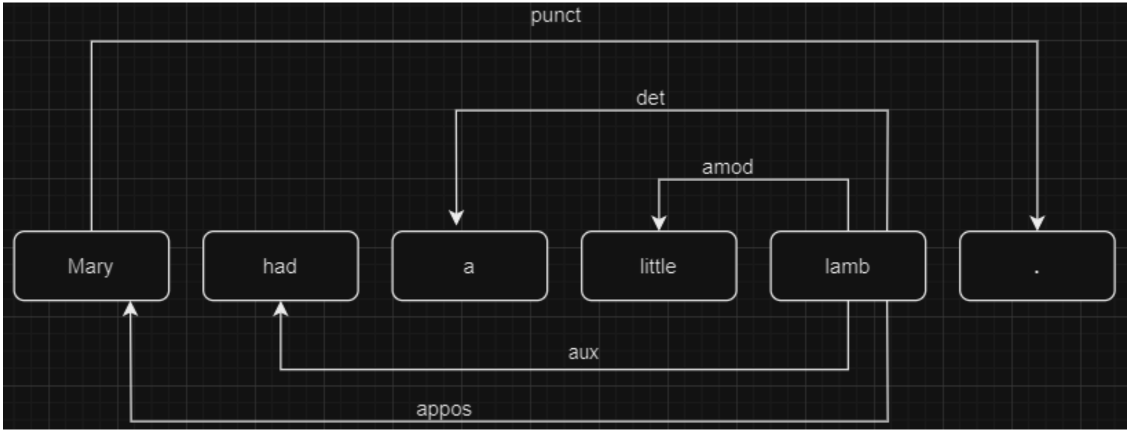
GloVe version:

Finally as we use a better version of GloVe, we get a better accuracy.
Hence, we are getting the maximum accuracy for the latest of all the versions used (840B).

Q4)

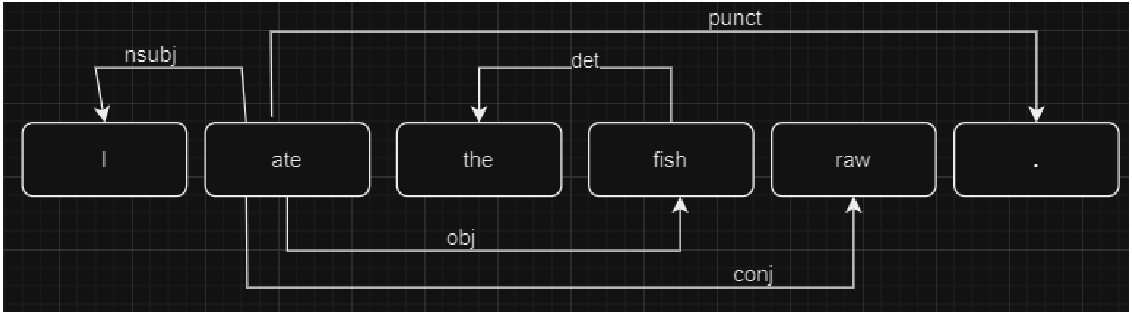
a) Mary had a little lamb . ||| PROPN AUX DET ADJ NOUN PUNCT
Predicted Actions:
SHIFT SHIFT SHIFT SHIFT SHIFT REDUCE_L_amod REDUCE_L_det REDUCE_L_aux REDUCE_R_appos SHIFT REDUCE_R_punct

Dependency Tree:



b) I ate the fish raw . ||| PRON VERB DET NOUN ADJ PUNCT
Predicted Actions:
SHIFT SHIFT REDUCE_L_nsubj SHIFT SHIFT REDUCE_L_det REDUCE_R_obj SHIFT REDUCE_R_conj SHIFT REDUCE_R_punct

Dependency Tree:



c) With neural networks , I love solving problems . ||| ADP ADJ NOUN PUNCT PRON VERB VERB NOUN PUNCT
Predicted Actions:
SHIFT SHIFT SHIFT REDUCE_L_amod REDUCE_L_case SHIFT REDUCE_R_punct SHIFT SHIFT REDUCE_L_nsubj REDUCE_L_nsubj SHIFT SHIFT REDUCE_R_obj REDUCE_R_advcl SHIFT REDUCE_R_punct

Dependency Tree:

Q5)

Parse State differences in models

Piyush Hinduja

October 2023

Even if the main idea of both the methods is same, that is, making a parse state using the embeddings of words and respective POS tags, we have implemented a way simpler version of the paper.

The main differences lies in the parse state representations which are as follows:

1) Size of word (or token) list taken:

For each parse state, we have used 4 tokens (c = 2) which comprises of 2 stack tops and 2 leftmost elements from buffer but the paper uses 18 tokens which has 3 stack tops, 3 left buffer elements and remaining are the children of stack elements which we have taken.

2) POS Embeddings list:

Since we have a reduced word list, our corresponding POS Tags are also 4 and the paper has 18 for each word.

3) Labels:

We use just the words and corresponding POS tags to train our model and predict the actions but the original paper also uses a set of labels corresponding to the children of stack words mentioned above.
Since we are not including the children, we don't use the labels either.

4) Other differences (outside Parse state) which I thought are crucial:

i. Where we use ReLU function in our model, the paper uses a cube activation function.
ii. We have used Glove static embeddings, the model uses Collobert et al., 2011 embeddings for English.
iii. The original paper also has some other features such as Activation function which captures higher order interaction features, pre-computation trick, etc. which we haven't implemented.