Next Word Predictor

A Research Project By Carter & Willow

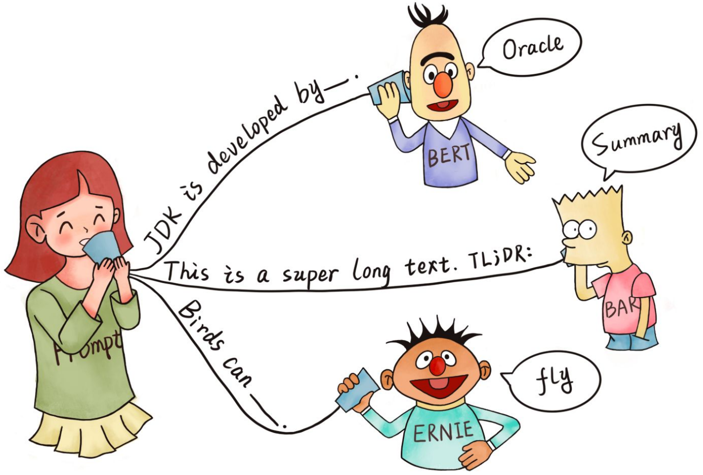
Foundation of Our Project

We started with W9 language model assignment

Uses Pytorch to build NN

Uses an LSTM RNN

Uses default W2V word embeddings



What Had to Change

Output words looking like this “I like to) when: “ **Added a Regex function**

Forced to use pre programmed prompts **Added user interface to to enter prompts**

One value to manage embedding dimension **Created Global Variable**

Model predicts a group of words based on prompt **Created funcion to predict one word at a time then predict again using added word**

Errors when entering a word that is not in word vocab **Added check if words entered are in vocab**

Language Model Architecture Modifications

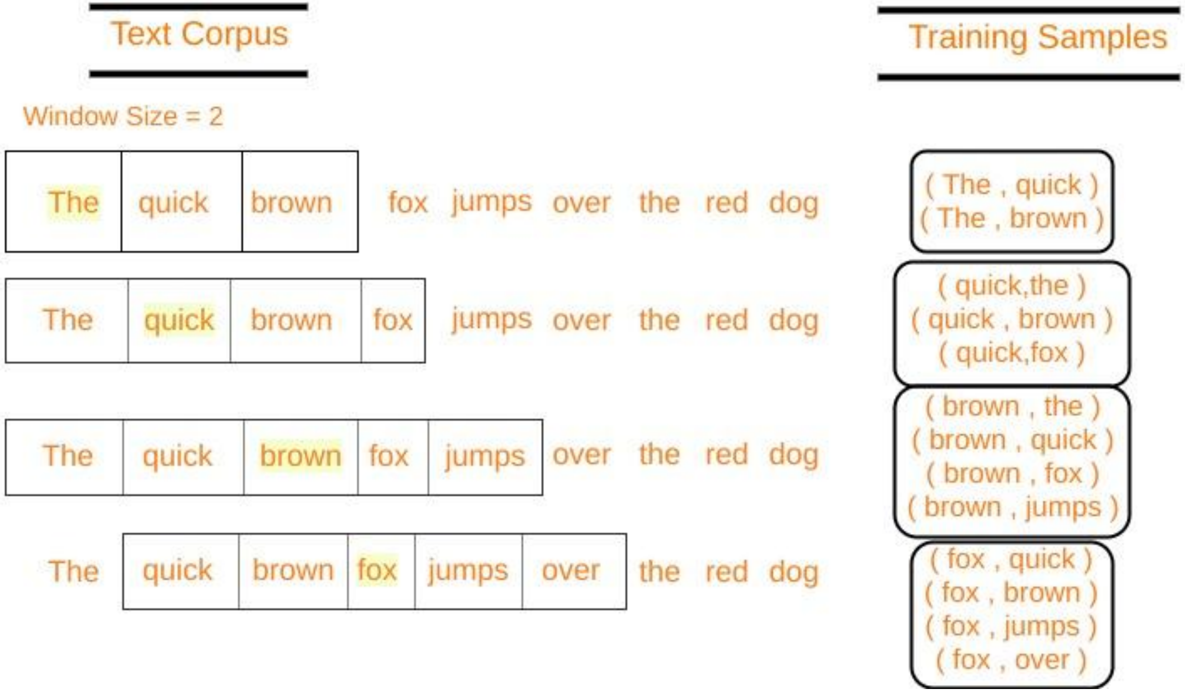
Tested word embedding models: W2V, Fasttext, Glove, W2V+N-grams, Bert

Tested NN Models: Plain RNN and LSTM

Why Does This Suck

The neural network is bottlenecked by a few factors

- Size of the Corpus
- Lack of positional data
- Quality of word embedding model
- Compute power



With all these drawbacks why do this?

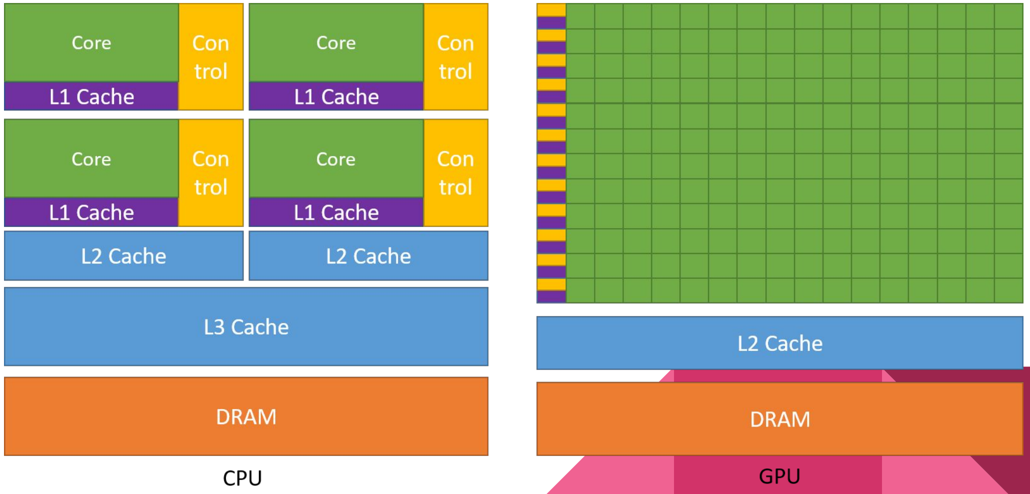
What might give us a chance

Cuda!

Why use the your CPU

when you can use a GPU

This will increase our hyperparameter tuning turnaround time



To Push The Limits and Learn from it

Even though this model is lacking a lot of features that would help its performance that doesn't mean we can’t get meaningful results.

We wanted to a crack at it since after talking with Amy we knew there was a chance to get meaningful results

As well we wanted to contribute to NLP research

How do we know its good or bad?

First We Try to Use Loss to Determine Quality

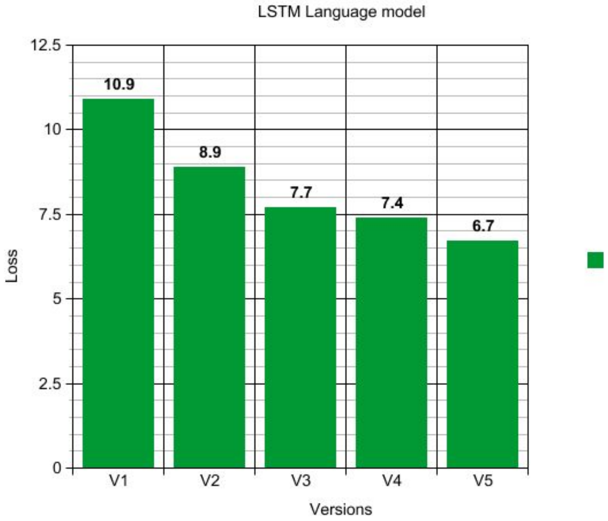
- Through hyperparameter tuning we have been able to reduce loss during our models training.
- However looking at loss by itself can be misleading

and does not give us the full picture

- So how do we know if our output is any good?

$$L\left(y,\hat{y}\right) = -\frac{1}{N}\sum_{i}^{N}\sum_{j}^{C} y_{ij} \log\left(\hat{y}_{ij}\right)$$

Cross-entropy Loss formula



We Made our own benchmarks!

In order to verify if our models responses are meaningful we need to give it a portion of a sentence that is fairly open ended as well as only having a few ways that it makes sense

- “I Like to”
- “Humans are”
- “War is”
- “What are”
- “The economy of the Netherlands” (one of Amy’s examples)

Funny Results

War is barrels destroyed drama women 1989 pardon Me

Humans are semi-submersible very Northern killers 2 Rowling obtain

humans are currencies

I like to battle virginia

Most meaningful results from our testing

there is no measurement possible among annexation tower typhus lee

war is 60% effective

the economy of the netherlands imports a

War is barrels destroyed drama women 1989 pardon Me

I like deep-water rigs

Humans are kind

what are terawatt-hours

what are different non-consecutive subdivisions

Did we successfully get meaningfully results?

YES and no…

Yes though or testing we were able to receive meaningful

However these responses were few compared to the number of confusing responses

In order to get more consistent meaningful responses we would have needed two things. I much larger dataset and even more compute power