10-10-1

1. The application is taking the previous 50 words and first turning them into numbers. Then, using the numbers matrix, it is then trying to predict the next word
2. The patent is taken and first filtered to remove stuff like $, #, %, etc. Then each word or part of the word (sometimes if it is a long word or with a prefix, it is broken up), and converted into a list of floats. The list of floats is then concacted into an array (embedding) to be used to train.
3. The features used is the previous 50 tokens (wordsish) as one large matrix (embedding)
4. Classes are the words in the model's vocabulary. The next word is the predicted value
5. It updates the weight of each node differently. Instead of applying .01 or .001, it changes it based off of how the gradient has been changed previously. It also uses the mean and variance to change the gradients accordingly. It also also has a moving average of past gradients (to help "roll" through local minima)