No	Experiment	UAS	LAS	Loss
1	Vanilla	67.8639978064	63.6887105217	0.351419464946
2	HL=2	64.0052845427	58.6110626418	0.533974532485
3	HL=3	15.0185706808	0.767754318618	4.51095648289
4	Sigmoid	42.8546501483	30.6503477329	1.18303002357
5	Tanh	59.6629857666	53.4536480794	0.554764813185
6	Relu	54.1715482215	47.9572251165	0.566298021376
7.	Parallel	49.0814367974	37.6349178652	0.788040463924
8.	Gradient Clipping	60.1436049555	56.0356208091	0.545713037252
9	Fixing Embeddings	35.833319474	20.5682382920	1.278347190232
10	HL=3(tanh activation)	69.5540543909	64.1274272752	0.444753924906

Q3

1 Hidden Layers

1. Hidden Layers = 2
   2 Hidden layers with cube activation function does not increase the accuracy but slightly decreases it. It may be because of the of using the cube activation both ways.
2. Hidden Layers = 3
   3 Hidden layers with cube activation function gives very high loss. Using other activation function like tanh it gives very high accuracy. This is because more hidden layers are better and will be more learnable.

2 Capturing Interactions

1. Sigmoid, tanh, ReLU
   Losses of all sigmoid, tanh, ReLU
   Tanh is better among all because it also has activation values that are negative. Although it squashes values between -1 and 1 and doesn't keep them open as the cubic would do.
   Sigmoid which gives output between 0 & 1 and ReLU gives only output either 0 or 1 perform around the same as there is not much variance in the output.

2. Parallel connections
   Using parallel hidden layers the loss is more comparatively.

It is because when trained together the POS, words and labels are able to learn better.

3. Fixing Embeddings
   By Fixing embeddings the loss is higher because the configuration of the embeddings remains the same and they are not learnt and therefore if we start with poor embeddings they will affect our final result.
4. Best Configuration
   Best Configuration is with 3 Hidden layers and tanh activation function UAS is 69.5540543909

5. Gradient Clipping
   Gradient Clipping:
   In Neural Networks when the gradient is propagated back if they are multiplied by numbers greater than one they keep increasing and lead to exponentially larger number in order to contain the magnitude of the gradient it is clipped to give output between two numbers.

   If Gradient Clipping is not used then the gradient explodes and at some results in nan. So Loss value without Gradient Clipping cannot be trusted.