Pruning words in NLG using RL

Step 1:-

Read papers :-

1>Learn What Not to Learn: Action Elimination with Deep Reinforcement Learning

2>Accelerated Reinforcement Learning for Sentence Generationby Vocabulary Prediction

3>Deep Reinforcement Learning in Large Discrete Action Spaces<If time permits>

Direction 1:-

Hierachical Approach:-

First the words are binned together and a hierarchical RL agent is used to learn the task.

Step 2:-

Figure out if the RL component will work or not. We will train a hierarchical agent on a small fourroom gridworld.

1>In a grid world normally an agent has 4 actions. To fit our scenario we will increase the number of actions from 4 to 100.
2>We will represent each action by a randomly generated N dimensional vector.
3>Do K-Means or K-Means++ on top of it. To get M clusters.
4>Create M DQN agents with actions corresponding to a particular cluster.(Lower level agents) [intra policy]
5>Create M+1 th DQN agent which has M actions.(Higher level agent)[inter policy]
6>Choose-Action:-

Choose an expert epsilon greedily

Choosen expert picks an action epsilon greedily.

7>Have seperate replay memory for each intra policy and update using DQN normal updates.
8>Have a seperate replay memory for the inter policy and update using DQN normal updates.
9>Compare this with a DQN agent with all the 100 actions.

Step 3:-

Deploy the above idea in a NLG setting. Datasets to be decided

Step 4:-

Compare the above idea with a REINFORCE based NLG agent and an DL method without RL agent.
DL method II where we are going to bin words again but wouldn't be having any RL agent.

Direction II:-
Step 2:-
1>understand the code https://github.com/hassyGo/NLG-RL . Recently accepted in NAACL-2019. I still have to see how better we can do.