NLP Written Report

Longfei ZHao, u5976992

Q2.

$count(am) = 3$
$count(am, Sam) = 2$
$|\{x : count(am, x) > 0\}| = 2$
Assume $N(w_i) = |\{x : count(x, w_i) > 0\}|$ and $d = 0.75$. Hence,
$N(\text{<s>}) = 0$
$N(\text{</s>}) = 2$
$N(I) = 2$
$N(am) = 1$
$N(Sam) = 3$
$N(do) = 1$
$N(not) = 1$
$N(like) = 1$
$N(green) = 1$
$N(apples) = 1$
$N(and) = 1$
$\Rightarrow \sum_{w_i} N(w_i) = 14$
$\Rightarrow \lambda(am) =$

$\Rightarrow P_{kn}(Sam) =$

$\Rightarrow P_{kn}(Sam|am) =$

Q3. Context-Free Grammars

$PRP\$ \rightarrow my \mid his \mid her \mid its$

~~$PNP \rightarrow nounEndWithS'$~~

$Nominal \rightarrow PNP$

~~$Det\ Nominal \rightarrow Det\ Noun$~~
$Nominal \rightarrow PRP\$\ Nominal$
$Nominal \rightarrow Nominal\ Noun$
$Nominal \rightarrow Noun$

Q4. Word Embeddings

We can consider an unseen word as it's subwords or character n-grams. We could train a ngram model which takes letters as tokens(Bojanowski, Grave, Joulin, & Mikolov, 2016). Therefore, we will get the frequence and word embeddings of all "syllables". Then, an unseen word can be splitted properly to a set of syllables. Hence, we use easily combine those syllables to get the word embedding.

Q5. Transition-based Dependency Parsing

Denote $\langle v_i|S, v_j|I, A\rangle$
The reason why $Left\text{-}Arc(LA)$ needs to remove the topmost element from the stack is that avoid creating a cycle in the graph. For example, if we keep v_i in S, there is a chance that adding an arc $v_i \rightarrow v_j$ to A in the later operation. $v_j \rightarrow v_i$ is alread in A. Therefore, there is a cycle (Nivre, 2003).
The reason for $Right\text{-}Arc(RA)$ is also to prevent to create a cycle. v_j should be reduced before v_i, otherwise arc linking these nodes might be added (Nivre, 2003).
The space complexity is also $O(n)$, the reason is as follow. For $Reduce(R)$ and $Shift(S)$, they won't increase the space. For LA and RA, the space will increase 1. It can be easily seen that $T_{RA} + T_S = n, T_{LA} + T_R \leq n$ (T_i means the time of i operation). Hence, $T_{RA} + T_{LA} \leq 2n$ and the initial data space complexity is $O(n)$. As a result, the space complexity is $O(n)$

Reference

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2016). Enriching word vectors with subword information. arXiv preprint arXiv:1607.04606.
Nivre, J. (2003). An efficient algorithm for projective dependency parsing. In Proceedings of

the 8th International Workshop on Parsing Technologies (IWPT.