Answers

ai:  
Adam算法使用了动量（momentum）的技巧，通过保持梯度的滚动平均值m来减少更新的变化。m的引入可以使更新的方向更加稳定，减少了更新的方差。这种低方差的更新有助于学习过程，因为它可以帮助算法更快地收敛到最优解，避免陷入局部最优点，并且在参数空间中更加平滑地搜索。

Aii:

Adam算法通过引入自适应学习率的技巧，扩展了动量的概念，通过保持梯度幅值的滚动平均值v来调整更新的幅度。由于Adam算法将更新除以√v，那些具有较大梯度幅值的模型参数将获得更大的更新。这有助于学习过程，因为较大的更新可以使模型更快地近最优解，特别是对于那些具有较大梯度幅值的参数，它们可能对模型的性能影响更大。通过自适应地调整学习率，Adam算法可以更好地平衡不同参数的更新幅度，从而提高学习的效率和稳定性。

Bi:

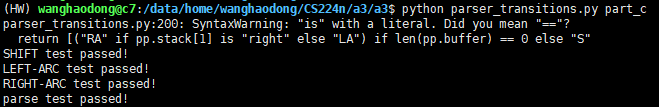
γ必须等于1/(1-pdrop)。这是因为希望dropout后的期望值仍然等于原来的值，即Epdrop [hdrop]i = hi。由于每个元素被置为0的概率为pdrop，被保留（即乘以1）的概率为1-pdrop，所以我们有：hi = γ\*(1-pdrop)hi + 0pdrop，解这个等式可以得到γ = 1/(1-pdrop)。

ii. Dropout应该在训练期间应用，因为它是一种正则化技术，可以防止模型过拟合。通过随机关闭一部分神经元，可以使模型更加健壮，不会过分依赖任何一个特征，从而提高模型的泛化能力。然而，在评估或测试阶段，希望模型是确定的，不应该引入随机性。此外，由于dropout在训练时已经对模型进行了正则化，所以在测试时没有必要再次应用dropout。

2.

A.

|  |  |  |  |
| --- | --- | --- | --- |
| Stack | Buffer | New dependency | Transition |
| (ROOT ,parsed,this) | sentence, correctly |  | SHIFT |
| (ROOT,parsed, this, sentence) | correctly |  | SHIFT |
| (ROOT, parsed, sentence) | correctly | Sentence->this | LEFT-ARC |
| (ROOT,parsed) | correctly | Parsed->sentence | RIGHT-ARC |
| (ROOT,parsed,correctly) |  |  | SHIFT |
| (ROOT,parsed) |  | Parsed->correctly | RIGHT-ARC |
| (ROOT) |  | ROOT -> parsed | RIGHT-ARC |

1. 一句话有n个词，入栈需要n次，出栈需要n次，所以一共需要2n次。
2. 
3. 
4. 