a4_written_answer

1. Attention Exploration (14 points)

(a): i. 当 $query$ 向量 q 与某个key向量 k_j 的相似度很高,即 $k_j^T j$ 远大于其他 key 的点积时，$softmax$后的注意力分布α会将几乎所有分布压在α_j上

ii. 输出向量 c 几乎就是第 j 个 $value$ 向量 v_j

(b): $q = k_a + k_b$

(c): i. 在噪声很小，key 方向几乎与 μ_i 一样时，选 $q = k_a + k_b$ 仍然能使注意力集中在a和 b 之间, 所以 $c \approx \frac{1}{2}(v_a + v_b)$

ii. 在这种协方差设定下，虽然 key 的方向大致没变，但 k_a 的长度波动很大，所以用同一个 q 得到的打分 e_a 有很大随机性。$softmax$ 会把这种长度差放大成注意力权重的差异：有时几乎全看 v_a，有时更平均，有时更偏向别的 $value$。结果就是，输出向量 c 在不同样本之间非常不稳定，方差明显高于 (i) 的情况，再也不能稳定地近似 $c \approx \frac{1}{2}(v_a + v_b)$

(d): (i) 选择：$q_1 = \mu_a + \mu_b, q_2 = \mu_a - \mu_b$ 这两个$query$ 会关注到相同的两个 key，得到的输出平均后接近：$c \approx \frac{1}{2}(v_a + v_b)$

(ii)由于 ka 的范数波动很大：$c1$和 $c2$ 的输出各自方差都很高（不稳定）,但因为 $q1$ 与 $q2$对 vb 的偏好方向相反,它们的误差在平均时会部分抵消。所以：最终输出$c \approx \frac{1}{2}(v_a + v_b)$的方差会明显小于单头注意力，结果更稳定。

2. Position Embeddings

(a). i. 由于对输入乘以置换矩阵P会使Q, K, V 都变成各自左乘 P，因此 $Z_{perm} = softmax(Q_{perm}K_{perm}^T)V_{perm} = Psoftmax(QK^T)V = PZ$

ii. 这一性质意味着 Transformer（无位置编码时）对词序完全不敏感： 如果打乱输入 token 顺序，它只会在输出中以同样方式打乱，而不会改变内容。这对 NLP 是非常糟糕的，因为语言语义高度依赖词序（如主谓宾关系、因果顺序等）。因此 Transformer 必须加入位置编码才能理解自然语言。

(b). i. 是的，位置编码会帮助解决前面的问题。 因为位置编码为每个 token 引入唯一的位置信息，使 Transformer 不再对输入的置换保持等变性，从而能够利用词序信息来理解语言。

ii. 不会。不同位置 t 具有不同的正弦和余弦值，因此它们的向量 Φ(t) 不可能完全一致。位置编码专门设计为在实际序列长度范围内保持唯一性

4. Considerations in pretrained knowledge

(a). 非预训练模型只看到有限的 name ↔ birthplace 监督样本，不足以覆盖世界上大多数人名，基本是在瞎猜常见城市，所以准确率接近或略高于 baseline。

预训练模型在大规模语料（维基）中已经多次见到这些人名及其出生地附近的上下文，微调时只是把已有知识对齐到“问答格式”，因此能显著高于 10%。