题一

(1) 从地到相机 $\vec{X} = \begin{pmatrix} 4 \\ 6 \\ 8 \end{pmatrix}$

$$R\vec{X} + T = \vec{x}$$

$$= \begin{pmatrix} -5 \\ -4 \\ 10 \end{pmatrix}$$

$$\vec{x}^{*\prime} = \begin{pmatrix} u' \\ v' \\ w' \end{pmatrix} = K\vec{x} = \begin{pmatrix} 20 \\ 20 \\ 10 \end{pmatrix} \Rightarrow u,v = (2,2)$$

(2) 不妨设 $P_w = \begin{pmatrix} a \\ b \\ c \end{pmatrix}$

$$\therefore P_1 = R_1 P_w + T_1 = \begin{pmatrix} -b+1 \\ -a \\ c+2 \end{pmatrix}$$

$$P_2 = \begin{pmatrix} b+1 \\ -c+8 \\ -a+6 \end{pmatrix}$$

$$K_1 P_1 = \begin{pmatrix} -8b+6c+20 \\ -5a+4c+8 \\ c+2 \end{pmatrix} = \begin{pmatrix} u_1 w_1 \\ v_1 w_1 \\ w_1 \end{pmatrix}$$

$$K_2 P_2 = \begin{pmatrix} 4b-4a+28 \\ -4c-4a+56 \\ -a+6 \end{pmatrix} = \begin{pmatrix} u_2 w_2 \\ v_2 w_2 \\ w_2 \end{pmatrix}$$

代入 u,v，化简:

$$\begin{aligned} -8b+6c+20 &= 2c+4 \\ -5a+4c+8 &= 2c+4 \\ -4a+4b+28 &= -18a+108 \\ -4a-4c+56 &= -4a+24 \end{aligned}$$

$$\Rightarrow c=8 \quad b=6 \quad a=4$$

$$P_w = (4,6,8)^T$$

题二

(认为题中的 x, y, z 轴均指地系

$$R = \begin{bmatrix} \cos\beta & \sin\beta & \\ -\sin\beta & \cos\beta & \\ & & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\gamma & & \sin\gamma \\ & 1 & \\ -\sin\gamma & & \cos\gamma \end{bmatrix} \cdot \begin{bmatrix} 1 & & \\ & \cos\alpha & -\sin\alpha \\ & \sin\alpha & \cos\alpha \end{bmatrix}$$

直接展开即可

$$= \begin{pmatrix} \cos\beta\cos\gamma & \cos\beta\sin\gamma\sin\alpha - \sin\beta\cos\alpha & \cos\beta\sin\gamma\cos\alpha + \sin\beta\sin\alpha \\ \sin b\cos\gamma & \sin\beta\sin\gamma\sin\alpha + \cos\beta\cos\alpha & \sin\beta\sin\gamma\cos\alpha - \cos\beta\sin\alpha \\ -\sin\gamma & \cos\gamma\sin\alpha & \cos\gamma\cos\alpha \end{pmatrix}$$

题三

为了方便，将三角形表倒着列

i	saw	the	boy	with	a	tele-
NP	V	D	N	P	D	N
ϕ	ϕ	NP	ϕ	ϕ	NP	
ϕ	VP	ϕ	ϕ	PP		
S	ϕ	ϕ	ϕ			
ϕ	ϕ	NP				
ϕ	VP					
S						

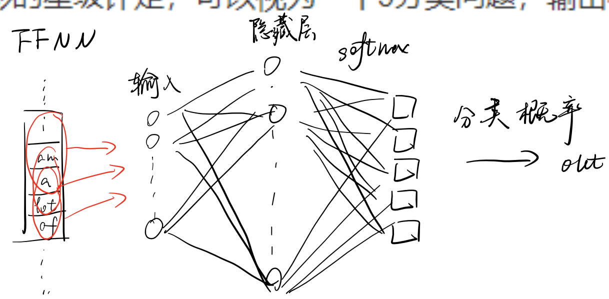
在最终层中有S，说明句子可以被解释为S的展开，符合语法

#题四 （在电脑上写了粘贴到这里）

题目4

基于FFNN的方案

- 输入和输出表示：
  - 输入：将影评文本的每个词表示为数值向量，可以使用独热模型，也可以使用已有的word2vec等词嵌入模型，从而映射为一个向量表示，方便输入网络。
  - 输出：电影的星级评定，可以视为一个5分类问题，输出概率最大的类别。
- 网络结构：
  - [图1]

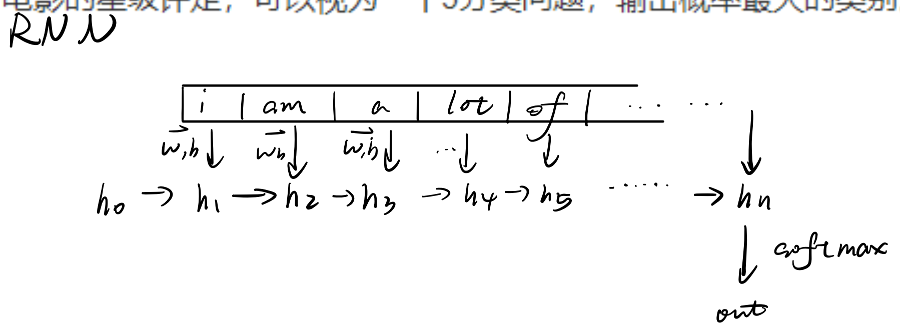


  - 输入层：由于FFNN是传统的神经网络结构，只能处理固定长度的输入向量，因此我们不能一下处理所有词。可以借助n-gram构造很多个定长的输入，比如使用滑动窗口来读取所有连续的n个词作为输入，拼接每个词的向量得到输入向量，同时保留了上下文信息。
  - 隐藏层：可以包含多个隐藏层，每个隐藏层使用激活函数（如ReLU）引入非线性变换。
  - 输出层：对于每次的输入，使用 softmax 函数输出分类的概率结果，并对一个文本内的所有输入的分类概率求积并归一化，从而得到文本的分类结果。
- 训练过程：
  - 数据集建立：可以在前一阶段要求部分用户写完影评文本后手动给出评分星级，构成数据集，再划分来构建训练集和验证集。
  - 数据预处理：对影评文本进行分词，并使用选定的模型来生成输入向量。
  - 训练：使用训练集对 FFNN 进行训练，采用与正确标签的交叉熵损失函数作为评价指标，对于从输入层到隐藏层的权重等参数，使用梯度下降算法或其变种来最小化损失函数。最后比较验证集上的准确率，调整超参数，直到满足一定的正确率要求。
- 推理过程：
  - 对于一个新的影评文本，首先进行分词，然后使用训练好的模型来预测其星级评定。FFNN的逻辑可以简单的理解为，筛选出一些关键词或词组（n-gram）所形成的特征维度，比如“好看”、“不好看”、“喜欢”、“不喜欢”等，然后根据这些维度上测试文本的向量大小即关键词的出现情况来判断影评的星级。

基于RNN的方案

> 题意是指RNN/Transformer里选一个？个人感觉FFNN和RNN是一个层面的对立的两种方案，而Transformer是另一个层面的，不太好比较，因此这里选用RNN。

- 输入和输出表示：
  - 输入：同上，将影评文本的每个词表示为数值向量，可以使用独热模型，也可以使用已有的word2vec等词嵌入模型，从而映射为一个向量表示，方便输入网络。
  - 输出：电影的星级评定，可以视为一个5分类问题，输出概率最大的类别。
- 网络结构
  - [图2]



  - 输入层：RNN的特点是可以处理变长的输入，因此可以直接将影评文本的每个词表示为数值向量，作为输入层的输入。
  - 隐藏状态：RNN的隐藏状态是一个向量，可以理解为网络的记忆；隐藏状态从头开始，每次输入一个词的向量后，隐藏状态会根据参数线性组合当前输入和前一个隐藏状态，并经过激活函数，从而得到当前的隐藏状态；因此，RNN可以保留上下文的“连续”信息。
  - 遍历输入的文本后，将得到的最后一个隐藏状态作为隐藏层输出，输出给输出层，使用softmax函数输出分类的概率结果，这个概率即文本的分类。
- 训练过程：
  - 数据集建立与预处理：同上。
  - 训练：整个训练过程与FFNN是类似的，只是由于RNN的网络结构与FFNN不同，其参数的位置是影响输入的每个词与前一个隐藏状态的结合的权重参数，因此其反向传播的结构也不同，但原理上仍是通过反向传播实现的梯度下降，达到降低loss函数（如交叉熵损失）的训练目的。并通过验证集来调整超参数，直到满足一定的正确率要求。
- 推理过程：
  - 使用训练好的模型，将输入的文本放入RNN，遍历完所有词后，得到最后一个隐藏状态，作为输出层的输入，使用softmax函数输出分类的概率结果，这个概率即文本的分类。
  - 逻辑上，RNN关注了连续词之间的联系，同时继承式的有序的使用一个状态记忆了所有词的输入，很适合评价性的分类问题。