Q：你在模型中使用了topic model来构造feature， 为何选用这个算法？这个算法的结果如何解读？

A: 原始数据中的product和pending\_code两列都是由多个code组合而成，对于他们最简单的处理方法是OneHot展开。但由于code种类很多，这样只会得到一个非常稀疏的高维矩阵，因此需要对这个矩阵进行降维，从中抽出有效信息。

PCA当然是一个可行的选择，但对于这两个feature，使用topic model是一个更好的选择。每一个code可以看成一个word，那么每一条记录（sample）可以看成一个或多个word构成的document。使用topic model对原始的onehot稀疏矩阵(sample \* word)进行分解，找出其中隐藏的”topic（主题）”，使得每一篇document都有一个topic分布，每一个词也有一个topic分布，进而将 sample \* word稀疏矩阵 分解为 sample \* topic矩阵 和 topic \* word矩阵两部分。这时候sample \* topic就可以看作降维后的矩阵，而topic \* word 矩阵就可以用来研究每一个topic的具体倾向。

以case study中的feature importance为例，product抽取出的topic3和topic4的重要性都靠前。考察product列使用主题模型分解出的 topic \* word 矩阵，找出这两个topic中权重最高的词语。可以看到 ‘life’,’cancer’等word最重要，可以用来描述这两个topic。

Q: 你使用AUC指标来评价你的模型，能介绍一下这个指标和选择他的动机吗？

A: AUC是ROC曲线下面积，常见的2分类模型评价指标。其意义大致可以理解为该模型把预测的阳性样本排在阴性样本之前的概率。

选择AUC是因为case study中我构造出来的这个数据集恰好是正负样本平衡的，此时这个指标能够很好地描述模型性能。但如果样本正负严重不平衡，我通常不会首先选择这个指标，而是更倾向于考察precision、recall等更直接描述阳性样本的指标。

Q:你会如何向不懂技术的其他同事，如销售等解释你这个模型？

A: 如果我要解释模型的话，我更倾向于单独训练一个logistic regression模型，虽然它的效力不如lightgbm等树模型，但是模型的各个参数都有odds ratio这样明确的可解释含义。树模型天生就不容易解释，即使单独画出某一棵树，也不容易说明白。

Q: 介绍一下你做feature engeering的整体思路

A: 对于这个case，如果是数值型feature，由于我使用了树模型，通常不优先做过多处理；对于类别型变量，如果类别在5个以下的话，个人经验直接one-hot即可；但如果类别很多，通常要考虑各种降维/embedding方法，如这里使用的topic model。

Q: 你是如何处理原始数据中的duplicates的？、

A: 首先我会尽量向数据源头收集者了解重复的来源, 如输入错误/确实存在重复现象/etc，再做下一步决定；但由于这次没有这个条件，我只能将数据保留第一条，移除剩下的。