Wide and deep 模型的核心思想是结合线性模型的记忆能力（memorization）和DNN模型的泛化能力（generalization）。wide端对应的是线性模型，输入特征可以是连续特征，也可以是稀疏的离散特征，离散特征之间进行交叉后可以构成更高维的离散特征。线性模型训练中通过L1正则化，能够很快收敛到有效的特征组合中。

Memorization of feature interactions through a wide set of cross-product feature transformations are effective and interpretable, while generalization requires more feature and interpretable

线性模型通过特征的交叉乘积后能够有很好的记忆功能，但如果要实现泛化能力则在特征工程中需要更多的特征。当特征不足时，

Deep neural networks can generalize better to unseen feature comnbinations through low-dimensional dense embeddings learned for the sparse features. However, deep neural networks with embedding can over-generalize and recommend less relevant items when the user-item interactions are sparse and high-rank.

将一些稀疏的特征转化成低维更密集的相量，作为神经网络的输入，通过多层网络学习到一些更高维的特征，问题就出在它会过度泛化，推荐一下不太相关的商品。

推荐系统面临的是：不仅要推荐相关性的商品，还有注重新颖性和多样性

作为比较的话，由于它数据集的问题，它只比较了wide-only model 和 deep-only model

Memorization can be loosely defined as learning the frequent co-occurrence of items or features and exploiting the correlation available in the historical data.

记忆可以松散地定义为学习项目或特征的频繁共现并利用历史数据中可用的相关性。

Generalization, on the other hand, is based on transitivity of correlation and explores new feature combinations that have never or rarely occurred in the past.

另一方面，泛化是基于相关的传递性，并探索过去从未或很少发生的新特征组合。

为什么认为Deep-model 会出现过度泛化的情况，因为针对一些具有特殊偏好的用户或者具有狭窄吸引力的小众项目，与大多数商品是没有交互的，因此很难学到有效的低维特征，而它最终又不得不有一个，很多时候我们并不是真的学到了一个十分行之有效的代表相量，可能只是一个最有可能的而已！

联合训练和集成训练的区别，联合训练各部分是起到互相补充的作用，而集成训练每一部分其实都是作为单独的完整的个体

**AppJoy: Personalized Mobile Application Discovery**

用户下载一个应用其实是弱指向性的，因为它并不能说明用户是否喜欢这个app,特别是这个应用是免费的，用户仅仅是想尝试一下而已。

除了协同过滤，还提到基于内容的推荐和基于位置的推荐，如一个人在火车站，那就很可能推荐给他一个火车列表的app