MMGCN多模态论文中用到的数据集可能没办法找到，作者只提供了三个数据集中小部分数据

Tiktok：论文中链接已失效

Kwai ： 2018年比赛用数据集，数据下载已经封闭。

MovieLens：从MoiveLens-10M数据集中收集了电影的标题和描述，并从Youtube5中抓取了相应的预告片。我们使用预先训练的ResNet50模型从从微视频中提取的关键帧中提取**视觉特征**。在声学模态方面，我们使用FFmpeg6分离音轨，并采用VGGish学习**声学深度学习特征**。对于文本情态，我们使用Sentence2Vector从微视频的描述中导出**文本特征**。

TKDE2022最新深度学习推荐系统综述总结了推荐系统中**信息增强**的方法

https://arxiv.org/pdf/2104.13030.pdf

基于内容增强的方法是指在利用用户对物品的交互矩阵的基础上对用户侧或者物品侧的信息进行建模的方法。其中，用户或物品侧的信息包括文本信息（比如物品标签、物品文字描述以及用户评论等），多媒体信息（比如图像、视频、音频等信息）以及用户侧的社交网络以及物品侧的知识图谱等。本文根据可获得的内容信息将相关工作分为了五类：用户和物品的一般特征建模、文本内容信息建模、多媒体信息建模、社交网络和知识图的建模。
1.建模一般交互特征的一类方法主要是对特征进行**二阶的特征提取**（比如FM、FFM等）以及基于MLP的多阶特征提取以及基于树结构的提取等。

2.建模文本特征的一类方法主要是利用自然语言处理技术对特征进行处理，比如基于**自编码器**的方法、基于**词嵌入**的方法、基于**注意力机制**的方法以及基于**文本可解释性**的推荐方法等。

3 .建模多媒体特征的一类方法主要是根据输入的数据是图片数据还是视频数据还是音频数据以及它们的结合进行了分类

4. 建模社交网络特征的一类方法主要是分为了社交正则化的方法和GNN的方法

5 建模知识图谱特征的一类方法主要是分为了基于路径的方法、基于正则化的方法和GNN的方法

Disentangled Self-Attentive Neural Networks for Click-Through Rate Prediction

https://arxiv.org/pdf/2101.03654v3.pdf

ABSTRACT

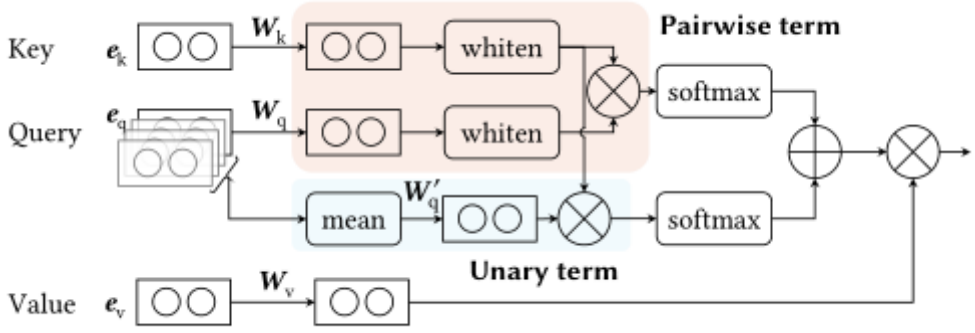
鉴于CTR预估数据具有稀疏和高维的特点，对高阶特征交叉建模是进行有效预估的关键，通过自注意神经网络(self-attention)对特征向量进行点积计算，是一种有效方式，但点积是在两个特征之间进行，**忽视了单个特征域(field)的影响。**

针对上述问题，论文提出 DESTINE结构，将一元(unary)特征重要性计算，从二阶特征交叉(pairwise interaction)解耦出来：一元项学习单个特征相对其他特征的重要度，二阶交叉项单纯地学习每个特征对的影响。

1 INTRODUCTION

CTR任务中使用的数据通常涉及许多分类特征，例如广告类别、用户设备模型等，因此是高维且极稀疏的，与连续的数字特征（如图像）不同。有了这样高维和稀疏的特征作为输入，一个复杂的模型将不可避免地容易过度拟合。因此，从这些高维数据中提取有用信息的成功解决方案是基于嵌入查找技术（也称为**交叉特征**）对特征域之间的组合交互进行建模。例如，对于电影CTR预测，一个基于**三阶特征交互的信息特征可以是{年龄、性别、流派}**，考虑到年轻男性倾向于喜欢动作片。然而，由于指数复杂性，不可能枚举所有组合特征交互。

DESTINE由图1所示的两个独立的计算块组成：whitened pairwise term，用于建模两个特征之间的特定交互，另一个unary term 用于描述一个特征对所有其他特征的一般影响。



对于CTR预测问题，我们首先将输入特征嵌入到低维空间中，然后通过堆叠多个解耦自注意层来计算高阶特征交互。最后，使用最后一个交互层产生的嵌入来预估点击行为。

2 THE PROPOSED DESTINE APPROACH

2.1 Problem Definition

假设训练数据集 D = $\{x_i, y_i\}_{i=1}^N$ 包含N个样本，其中每个样本 x_i 由M个用户和物品的特征字段组成，其关联的标签y_i∈{0,1}表示该用户的行为(例如是否点击某项)。点击率预测的问题是在给定一个特征向量 x_i 的情况下，预测$\hat{y}_i$，以准确估计用户是否会与物品进行交互。

2.2 Learning Decoupled Feature Interaction

DESTINE由三个关键组件组成：（a）嵌入层、（b）交互层和（c）输出层。首先，输入特征被送入嵌入层，嵌入层将输入特征转换为密集的低维嵌入向量。然后，这些特征嵌入被送到几个堆叠的交互层中，这些交互层对高阶交互进行建模。之后，我们将最后一个交互层的嵌入信息输入到输出层，以预测点击行为。

对于每个稀疏输入特征 x_i, 我们通过嵌入查找将其转换为密集嵌入 $e_i \in R^d$ 。一旦我们获得了每个特征的紧凑表示，我们就使用缩放的点积注意方案来建模特征域之间的高阶特征交互。具体来说，我们将每个特征交互定义为（键、值）对，并通过乘以每个特征嵌入来学习每个特征交互的重要性，这样重要的键-值对获得更高的关注分数。形式上，我们首先将每个特征嵌入转换为新的嵌入空间 如 $\mathbb{R}^{d'}$ 下所示

$$\boldsymbol{q}_m = \boldsymbol{W}_{\mathrm{q}} \boldsymbol{e}_m, \tag{1}$$

$$\boldsymbol{k}_n = \boldsymbol{W}_{\mathrm{k}} \boldsymbol{e}_n, \tag{2}$$

其中 query 和 key 变换分别由两个线性变换矩阵 Wq、Wk参数化。

然后，我们计算特征 m 和 e 的相关性α (em, en) 。先前在视觉表示学习中的工作证明了特征m /特征n的重要性得分可以分解为两个项:：一个用于建模特定交互的成对 term，一个用于对所有特征域的一般影响进行建模的一元 term。我们将这两项相加：

$$\alpha(\boldsymbol{e}_m, \boldsymbol{e}_n) = \alpha_{\mathrm{p}}(\boldsymbol{e}_m, \boldsymbol{e}_n) + \alpha_{\mathrm{u}}(\boldsymbol{e}_m, \boldsymbol{e}_n). \tag{3}$$

对于成对项，我们对key和query 向量执行whitening，以对特征之间的交互进行建模，这使得两个交互特征之间的相关性降低：

$$\alpha_{\mathrm{p}}(\boldsymbol{e}_m, \boldsymbol{e}_n) = \sigma\left(\left(\boldsymbol{q}_m - \boldsymbol{\mu}_{\mathrm{q}}\right)^{\top}\left(\boldsymbol{k}_n - \boldsymbol{\mu}_{\mathrm{k}}\right)\right), \tag{4}$$

$\sigma(.)$是softmax函数，$\boldsymbol{\mu}_{\mathrm{q}} = \frac{1}{M}\sum_{i=1}^{M} W_q e_i$和$\boldsymbol{\mu}_{\mathrm{k}} = \frac{1}{M}\sum_{j=1}^{M} W_k e_j$分别取key和query vector的平均值。

对于一元项，我们引入另一个查询变换矩阵W ' q∈Rd ' ×d用于建模显著特征:

$$\alpha_{\mathrm{u}}(\boldsymbol{e}_m, \boldsymbol{e}_n) = \sigma\left(\left(\boldsymbol{\mu}'_{\mathrm{q}}\right)^{\top}\boldsymbol{k}_n\right), \tag{5}$$

u'_q 是另一个变换的平均向量，用于模拟对 key 向量的一般影响，即 $\boldsymbol{\mu}_{\mathrm{q}'} = \frac{1}{M}\sum_{i=1}^{M} W'_q e_i$。

在计算每个特征交互(m, n)的注意力得分之后, 我们通过线性变换矩阵W v∈Rd ' ×d参数化的值变换将每个候选特征变换到新的嵌入空间,

$$\boldsymbol{v}_k = \boldsymbol{W}_{\mathrm{v}}\boldsymbol{e}_k. \tag{6}$$

最后，我们通过线性组合所有特征,更新了特征域的最终表示m 。

扩展到多头自注意。 为了让模型学习不同子空间中的不同特征交互，我们使用了多个注意头。具体来说，我们计算了注意头 h 下的特征域m的表示

$$\boldsymbol{z}_m^{(h)} = \sum_{k=1}^{M} \alpha^{(h)}(\boldsymbol{e}_m, \boldsymbol{e}_k) \cdot \boldsymbol{v}_k^{(h)}, \tag{7}$$

其中$\alpha(h)$是通过式(3)计算的注意得分。注意，每个注意头h保持其不同的权重参数$W_k^{(h)}$,$W_q^{(h)}$,$(W'_q)^{(h)}$,$W_v^{(h)}$,

然后，我们得到 z_m, 特征的整体隐藏表示m, 通过将所有注意力头的表示连接为

$$\boldsymbol{z}_m = \left[\boldsymbol{z}_m^{(1)}; \boldsymbol{z}_m^{(2)}; \ldots; \boldsymbol{z}_m^{(H)}\right], \tag{8}$$

H 是注意力头的数量。此外，根据先前的工作，我们通过**残差**连接合并了原始的单个特征（一阶特征），

$$\hat{\boldsymbol{z}}_m = \varphi(\boldsymbol{z}_m + \boldsymbol{W}_{\mathrm{r}}\boldsymbol{e}_m), \tag{9}$$

其中，Wr∈Rd 'H×d为线性投影矩阵，以避免维度不匹配，$\varphi(\cdot)$= max(0，·)为ReLU激活函数。

使用从最后一个交互层接收到的特征 $\hat{z}m$，我们将所有M个特征连接起来，并在它们上面使用一个简单的逻辑回归模型来预测用户行为，公式为

$$\hat{y} = \sigma\left(\boldsymbol{w}^{\top}\left[\hat{\boldsymbol{z}}_1; \hat{\boldsymbol{z}}_2; \ldots; \hat{\boldsymbol{z}}_M\right] + b\right), \tag{10}$$

其中 W 是线性投影向量，b是偏置项，c（x）=1/（1+e^{-x}）是sigmoid函数。

最后，我们使用在CTR预测模型中广泛使用的二元交叉熵函数作为损失函数，

$$\mathcal{L} = -\frac{1}{N}\sum_{(y,\hat{y})\in\mathbb{D}}\left(y\log\hat{y} + (1-y)\log(1-\hat{y})\right), \tag{11}$$