正方形的圆


TAAC
腾讯广告算法大赛
Tencent Advertising Algorithm Competition

Contents

目录

1 团队介绍

2 赛题理解

2.1 赛题陈述

2.2 数据分析

3 特征工程

3.1 数据预处理

3.2 词向量特征

3.3 手工特征

3.4 数据增强

4 模型介绍

4.1 Text Matching Models

4.2 Transformers

4.3 Model Stacking

5 总结与思考

5.1 DEMO

5.2 经验与教训

团队介绍

团队介绍

黄泳锐，浙江大学计算机科学与技术学院研究生一年级，2019年，从华南师范大学获得软件工程专业学士学位，目前主要的研究兴趣是用户画像、机器学习及其应用。

方俊伟，浙江大学计算机科学与技术学院研究生一年级，2019年，从武汉大学获得信息管理与信息系统专业学士学位，目前主要的研究兴趣是普适计算、机器学习及其应用。

02

赛题理解

赛题陈述/数据分析

赛题理解—赛题陈述

传统推荐系统、计算广告任务：

特征集合(用户行为特征+人口统计学特征+…..) -->CTR预测,TopK推荐

特征集合(**with/without** 人口统计特征) -->推荐性能对比--> 判别人口统计特征是否有效

本次赛题：

用户在广告系统内的交互序列-->人口统计学特征(年龄/性别)

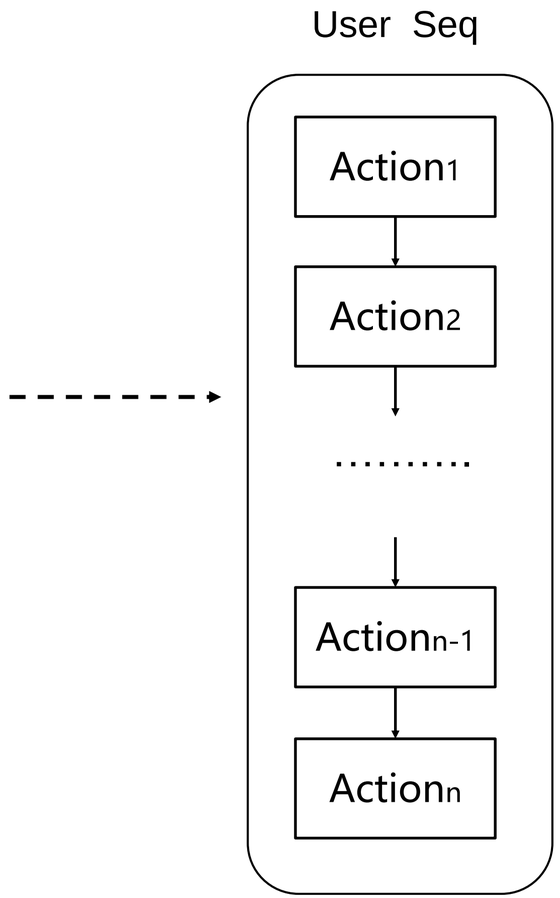
研究意义：

1. 缺失用户信息的填补
2. 在更广泛的但是拥有信息较少的用户群体上预测用户特征，进行基于人口统计特征的智能定向和受众保护。

用户数据：

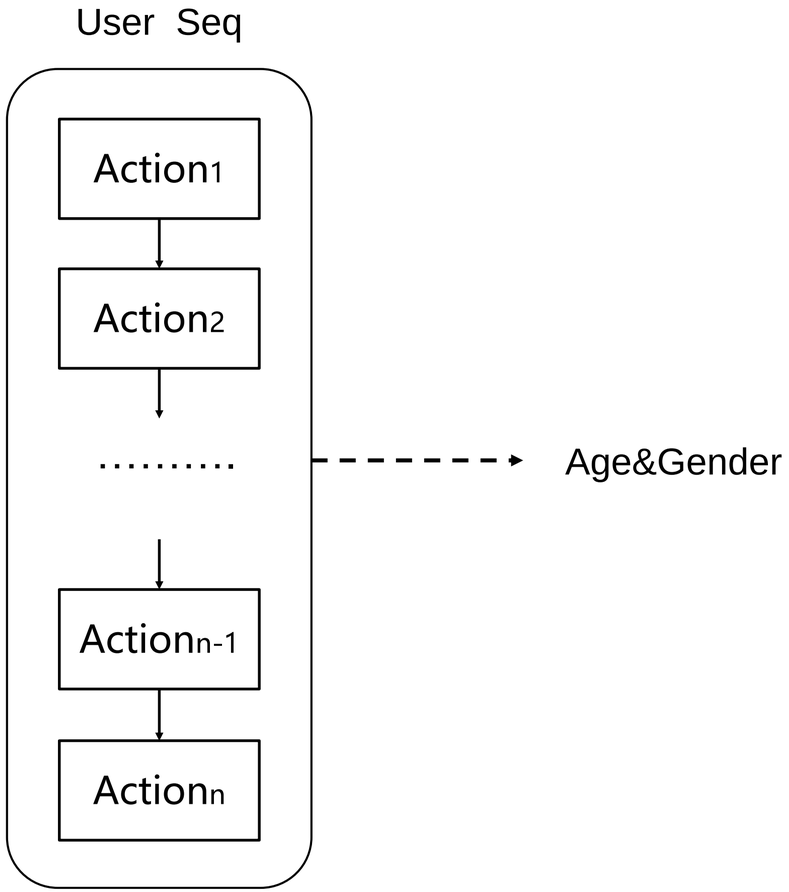
样本：针对每位用户拥有一组该用户在长度为 91 天（3 个月）的时间窗口内的广告点击历史记录，时间窗口内每个广告点击行为(Action)包括以下信息，根据时间的先后能够形成每个用户的广告点击行为序列：

Creative_id	广告素材id
Ad_id	广告id
Product_id	广告宣传产品id
Product_category	产品类别id
Advertiser_id	广告主id
Industry	广告主所属行业id
Click_times	该条广告点击次数
Time	行为时间(天)



预测标签：

样本：使用用户的广告点击行为序列预测用户的性别及年龄，性别的label为 [1,2] (对应男女)，年龄的label为 [1-10] (对应不同的年龄段)。



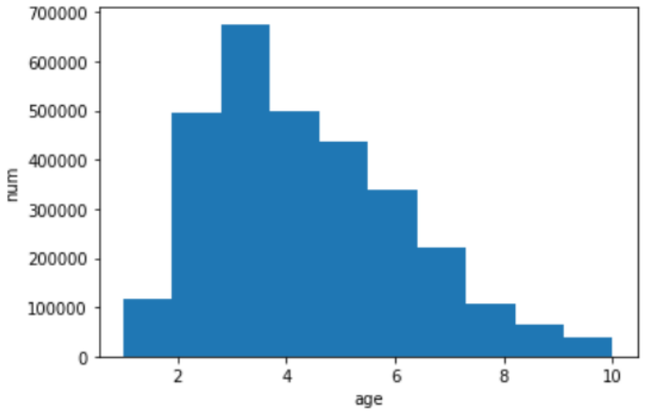


Fig 1 年龄分布

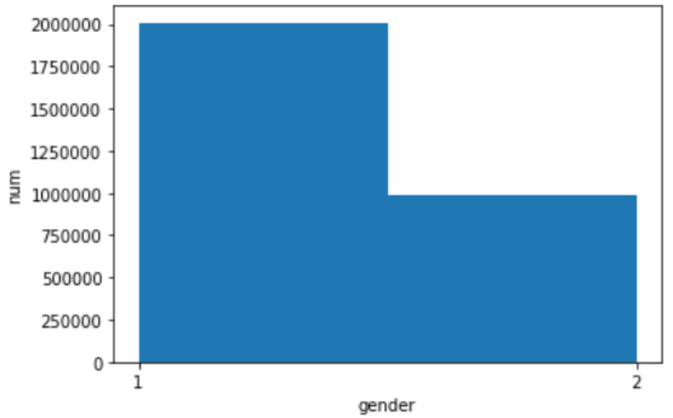


Fig 2 性别分布

① 性别分布上呈现性别比例2比1的态势。

② 年龄分布上，（据我们推测）20-35岁的用户应该对应占比最高的label2-4，其他年龄段的用户相对较少。

③ 存在轻微的样本不平衡现象。

用户数据：

样本：针对每位用户拥有一组该用户在长度为 91 天（3 个月）的时间窗口内的广告点击历史记录，时间窗口内每个广告点击行为(Action)包括以下信息，根据时间的先后能够形成每个用户的广告点击行为序列：

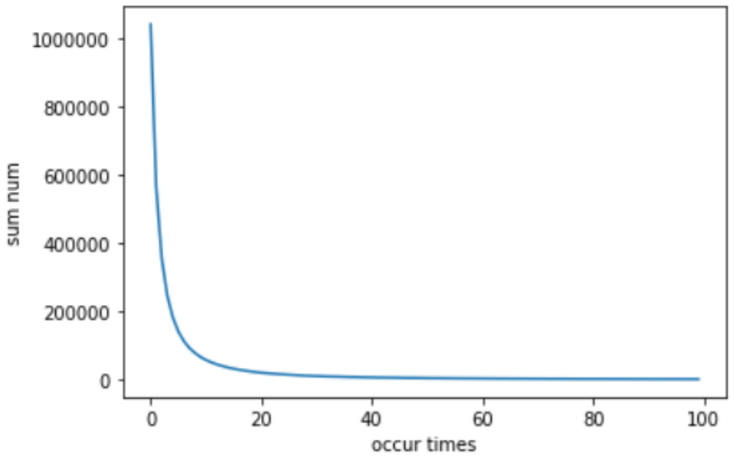


Fig 3 Ad_id 数量-出现次数

① ad_id的点击序列中的ID分布呈长尾状，即**少量的高曝光广告被点击多次，大部分广告鲜有用户点击。**

② Id的长尾分布与文本中的词语分布呈现相同的趋势，可以考虑使用**NLP**的数据处理及建模方式解决这一问题。

特征工程

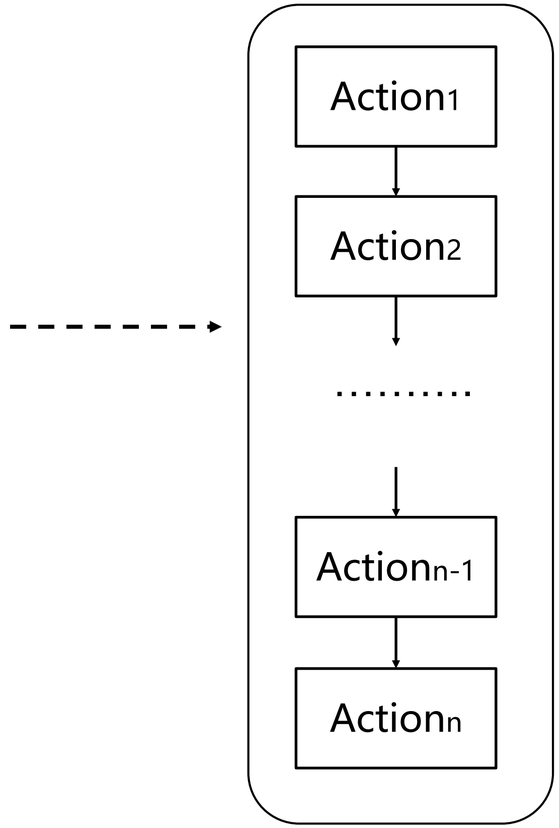
数据预处理/词向量特征/手工特征/数据增强

特征工程—数据预处理

User Data

Creative_id	广告素材id
Ad_id	广告id
Product_id	广告宣传产品id
Product_category	产品类别id
Advertiser_id	广告主id
Industry	广告主所属行业id
Click_times	该条广告点击次数
Time	行为时间(天)

User Seq



特征工程—词向量特征

序列ID(离散化，难以学习) --Pretrain--> 序列ID embedding(每个ID用100-300维的向量表示，有更强的语义信息，更方便学习)

词序列构建：

1. 使用6个ID分别构建序列，每个用户即拥有 Creative_id Seq, Ad_id Seq, Product_id Seq,Product_category Seq, Advertiser_id Seq, Industry Seq 6个序列，针对**6种ID的序列分别训练词向量。**
2. 各个ID的单词分布如下所示，可以看到最为重要的Creative_id和Ad_id的总ID数在**300万-400万**之间，与正常的词表的长度(**1.5万-10万)**差别较大，且大量ID的出现次数很低，在实际词向量构建时需要采取同文本词向量训练不完全一致的设置。
3. 部分ID存在**缺失值**，且比例不低，在词向量训练中需要考虑对这一问题进行处理。

	creative_id	ad_id	product_id	product_category	advertiser_id	industry
count	4445720	4445720	4445720	4445720	4445720	4445720
unique	4445720	3812202	44315	18	62965	336
top	4445720	681564	\N	2	14681	247
freq	1	37	1575509	1668098	32027	450335

Fig 4 各个id统计次数分布

特征工程—词向量特征

词向量构建：

Word2Vec(Skip-Gram):

① 对6个ID分别训练6套词向量。

② 考虑到大部分**ID出现次数少，ID总数多**，在NLP中常用的5-20的窗口长度，在高度低频化的ID序列中可能难以充分捕捉到不同ID之间的共现关系，改用了**175长度**的窗口训练词向量。

③ Product_id和Industry两项ID中有大量的‘\N’缺失值，将两个id相互交叉形成**Product_id_Industry**，同时与其密切结合的**Advertiser_id交叉形成Advertiser_id_Product_id和Advertiser_id_Industry两个新的序列**，额外训练词向量，解决原始ID的缺失问题，同时一定程度上捕捉了不同ID的相互关系。

Glove:

① 各种设置与Word2Vec基本一致。

② 相较于Word2Vec滑动窗口，对窗口内共现的单词使用SGD优化其词向量；Glove词向量则从统计角度出发，基于局部窗口来构建全局的词共现矩阵，将词向量的内积拟合全局共现矩阵，同时利用了**全局的先验信息**和**局部的窗口信息**，与Word2Vec词向量存在一定差异。

[1] Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[J]. arXiv preprint arXiv:1301.3781, 2013.

[2] Pennington J, Socher R, Manning C D. Glove: Global vectors for word representation[C]//Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014: 1532-1543.

特征工程—手工特征

手工特征分为两类：

(1) 序列特征：序列中每个item都有对应特征值（Target Encoding、图特征、TF-IDF Sequnece、TextRank Sequence）

(2) 用户特征：每个用户有对应的特征值（TF-IDF stack概率，统计特征）

TF-IDF Sequence

TF-IDF序列

Textrank

id序列中每个id生成TextRank值

Target Encode

交叉验证做每个id对于label的目标编码

图特征

根据序列关系建图，提取图特征，如PageRank, HITS，图出入度等。

序列特征

统计特征

出现Item个数、独立item出现次数等

TF-IDF stack

TF-IDF作为特征、弱分类器概率集成作为新特征

用户特征

特征工程—手工特征

序列特征：

- **TF-IDF特征：**将整个的训练集+测试集的用户行为序列看作语料库，每个用户的序列作为一个文档，每个点击行为作为一个词语，根据6个不同的id可以生成6个语料库，计算6个语料库内每篇文档每个词语的TF-IDF值,从而形成对应6个ID的TF-IDF Sequence。TF-IDF特征能够有效地捕捉那些非流行词但在本序列内出现次数较高的词语。

$$\text{词频(TF)}=\frac{\text{该词在文档内出现的次数}}{\text{文档内的总词数}} \quad \text{逆文档概率(IDF)}=\log\left(\frac{\text{语料库的文档总数}}{\text{出现该词的文档数}+1}\right) \quad \text{TF-IDF=TF*IDF}$$

- **TextRank特征：**同TF-IDF特征相同，将每个点击行为看成一个词语，为对应的6个ID序列计算TextRank值序列。从序列内部的id共现次数的角度出发，为频繁出现在其他词周围的词赋予高权值，提取出关键词，生成TextRank-Sequence。

$$S(V_i) = (1-d) + d\sum_{(j,i)\in\omega}\frac{S(V_j)}{Out_j}$$

特征工程—手工特征

序列特征：

- **Target Encoding特征：**

1. 对每个ID根据其所属用户群的age&label的均值进行编码,获取到标签的一个在ID层面上的细粒度分布。
2. 采用训练集内部5折交叉，测试集使用训练集5折的平均进行编码，避免标签泄露。
3. 引入**先验**(全局的target mean vaue)和**后验**(每个ID对应的target mean value)**结合**的方式编码，避免低频ID获取到的编码不够可靠。
4. 对于训练集中不存在的id，直接使用先验结果(全局的target mean vaue)为其编码。

- **图特征**

根据id的序列关系建图后，能够捕捉原本序列特征无法捕捉的特性，提取建图后图结构的特征，包括结点的出入度、PageRank和HITS特征。PageRank算法可以对图中更多结点指向的重要结点赋予一个更高的分数。而HITS算法可以挖掘出图中的高质量权威、枢纽结点。

PageRank: $$PR(A) = \left(\frac{PR(B)}{L(B)} + \frac{PR(C)}{L(C)} + \frac{PR(D)}{L(D)} + \cdots \right) d + \frac{1-d}{N}$$

HITS: $$auth(p) = \sum_{q \in p_{to}} hub(q)$$
$$hub(p) = \sum_{q \in p_{from}} auth(q)$$

特征工程—手工特征

用户特征：

- 以每个用户的整个点击序列为单位，计算聚合序列内各个ID的**统计特征**，包括：
  - 点击序列内6个ID的不同ID个数
  - 点击序列内6个ID分别出现次数最多的ID的出现次数
  - 点击序列内每条广告的平均/最大/最小点击次数及点击次数的标准差
  - 点击序列内每天的平均/最大/最小广告点击次数及每天广告点击次数的标准差
  - ……

	user_id	agg_click_times_mean	agg_click_times_max	agg_click_times_min	agg_click_times_std	agg_click_times_nunique	agg_click_times_freq	agg_click_ti
0	1	0.302252	0.221553	-0.001637	0.488323	0.395266	-0.308565	
1	2	-0.448291	0.221553	-0.001637	-0.185690	0.395266	0.194960	
2	3	-0.753200	-0.869045	-0.001637	-0.968952	-1.052705	-0.025332	
3	4	-0.753200	-0.869045	-0.001637	-0.968952	-1.052705	-0.041067	
4	5	-0.337416	0.221553	-0.001637	-0.054300	0.395266	0.006138	

- TF-IDF stack：

忽略TF-IDF序列特征，使用固定维的特征组织方式对每个用户进行表征，使用SGDClassifier, PassiveAggressiveClassifier, BernoulliNB, MultinomialNB, LogisticRegression5个弱分类器交叉验证得出概率。

特征工程—数据增强

序列shuffle：

由于时间粒度仅仅到天，且很多用户在一天内进行了多个广告点击行为，一天内的广告点击行为的先后顺序为止，故可以考虑对序列内同一天的点击行为排列组合生成更多中不同的序列，增加数据量，使得训练的模型泛化能力更强。

user_id	creative_id	time	click_time
1	[821396, 209778, 877468, 1683713, 122032, 7169...	[20, 20, 20, 39, 40, 43, 46, 52, 60, 64, 64, 7...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2]
2	[63441, 155822, 39714, 609050, 13069, 441462, ...	[10, 11, 14, 17, 28, 28, 28, 38, 38, 39, 41, 4...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
3	[661347, 808612, 710859, 825434, 593522, 72694...	[12, 13, 14, 14, 14, 17, 19, 22, 31, 36, 37, 4...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
...	...	...	...

Seq1.1
Seq1.2
Seq1.3 ……

Seq3.1
Seq3.2
Seq3.3 ……

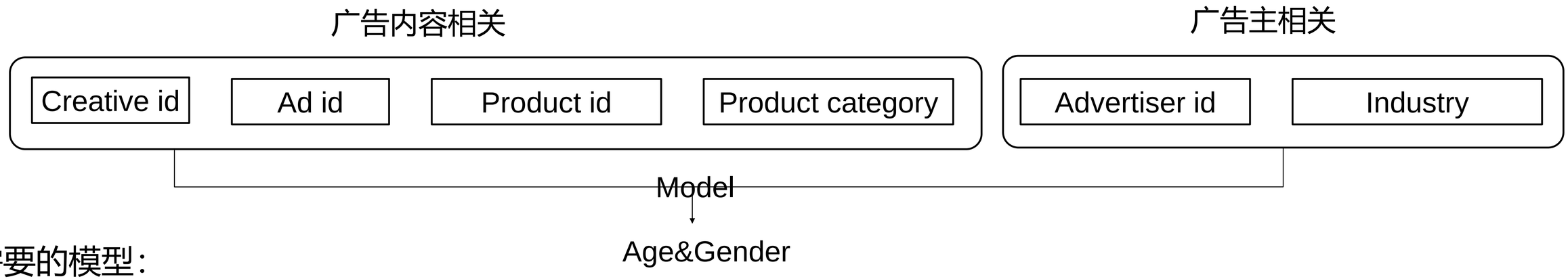
模型介绍

Text Matching Models/Transformer/Model Stacking

模型介绍—Text Matching Models: ESIM&RE2

模型输入：

- 6个ID可以划分为2类，将每个类的ID Seq的词向量拼接作为每个类的Sequence，模型的基本输入由对应2个类的2个Sequence组成。
- 不同的Sequence之间，Sequence内部都会有一定的相互关系，如广告使用的素材应该与宣传的商品类型相匹配；广告主所在的行业应该也与其宣传的商品有一定的对应关系。

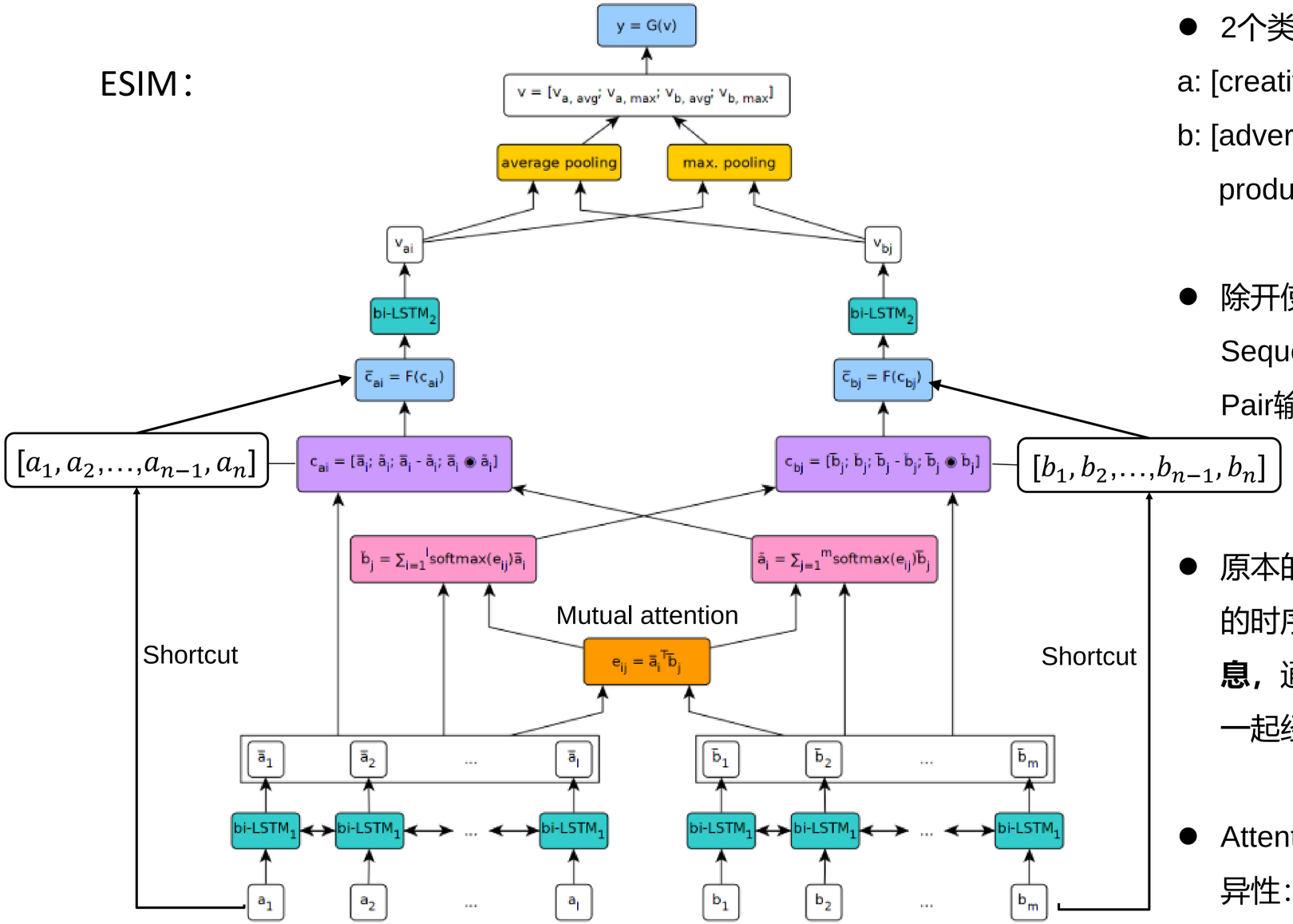


需要的模型：

- 处理多个序列输入
- 在模型内部实现多个序列的交互

模型介绍—Text Matching Models: ESIM&RE2

ESIM：



- 2个类的sequence&使用交叉序列：

a: [creative_id, ad_id, product_id] 拼接

b: [advertiser_id, product_id_advertiser_id,\
product_category_industry]拼接

- 除开使用pair sequence之外，拼接所有的id作为单独Sequence输入,原本的mutual Attention->Self-Attention。Pair输入和单输入的模型同时使用。

- 原本的模型基础上，除去使用第一层LSTM和Attention捕捉的时序信息外，**输入的词向量中也具备有预测性别/年龄的信息，**通过shortcut连接，和attention输出的sequence拼接，一起经过最后一层LSTM的信息提取。

- Attention机制的修改，引入SVD形式的注意力，增强模型差异性：

$$e_{ij} = \tilde{a}_i{}^T \tilde{b}_j \rightarrow S_{ij} = Relu\left(\tilde{a}_i{}^T W^T\right) DRelu(W\tilde{b}_j)$$

[3] Chen Q, Zhu X, Ling Z, et al. Enhanced lstm for natural language inference[J]. arXiv preprint arXiv:1609.06038, 2016.

模型介绍—Text Matching Models: ESIM&RE2

$$\bar{a}_i^1 = G_1([a_i; a'_i]),$$
$$\bar{a}_i^2 = G_2([a_i; a_i - a'_i]),$$
$$\bar{a}_i^3 = G_3([a_i; a_i \circ a'_i]),$$
$$\bar{a}_i = G([\bar{a}_i^1; \bar{a}_i^2; \bar{a}_i^3]),$$

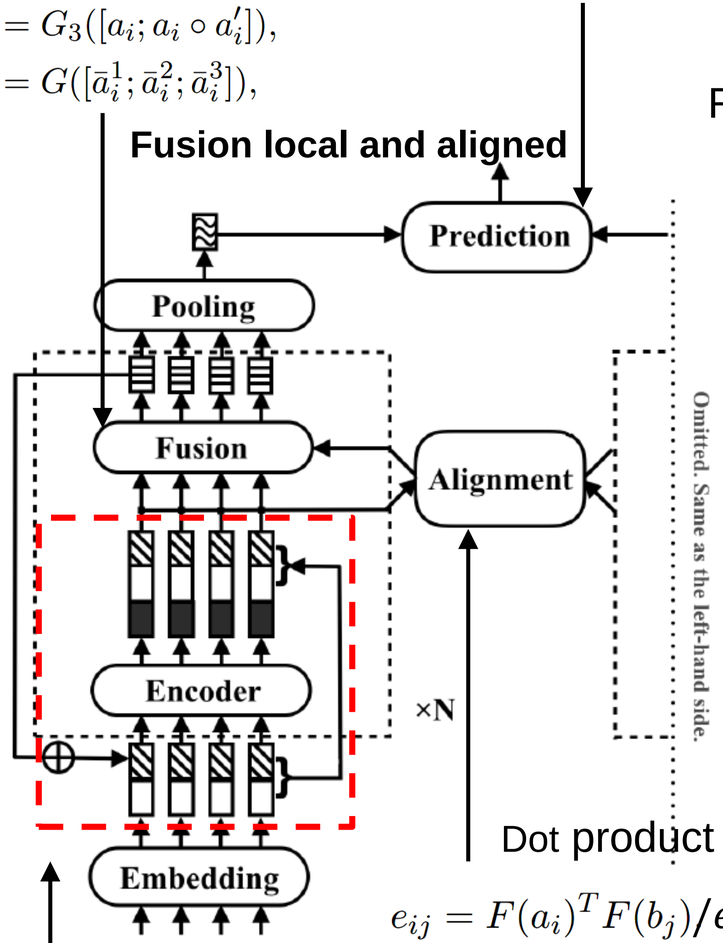
Pair输入

$$\hat{\mathbf{y}} = H([v_1; v_2; |v_1 - v_2|; v_1 \circ v_2])$$

单输入

$$\hat{\mathbf{y}} = H(\text{maxpool}(\text{sequence}); \text{sumpool}(\text{sequence}))$$

RE2



Dot product attention

$$e_{ij} = F(a_i)^T F(b_j) / e_{ij} = F(a_i)^T F(a_i)$$

Pair输入　　单输入

CNN Residual Encoder

$$x_i^{(n)} = [x_i^{(1)}; o_i^{(n-1)} + o_i^{(n-2)}]$$

Pair输入模型：将id特征分为两部分，对应两个Sequence

input1：广告特征， ad_id， creative_id等

input2：广告主特征， advertiser_id、 industry等

其他特征(target encoding、图特征，id)直接拼接在input1和input2后面。

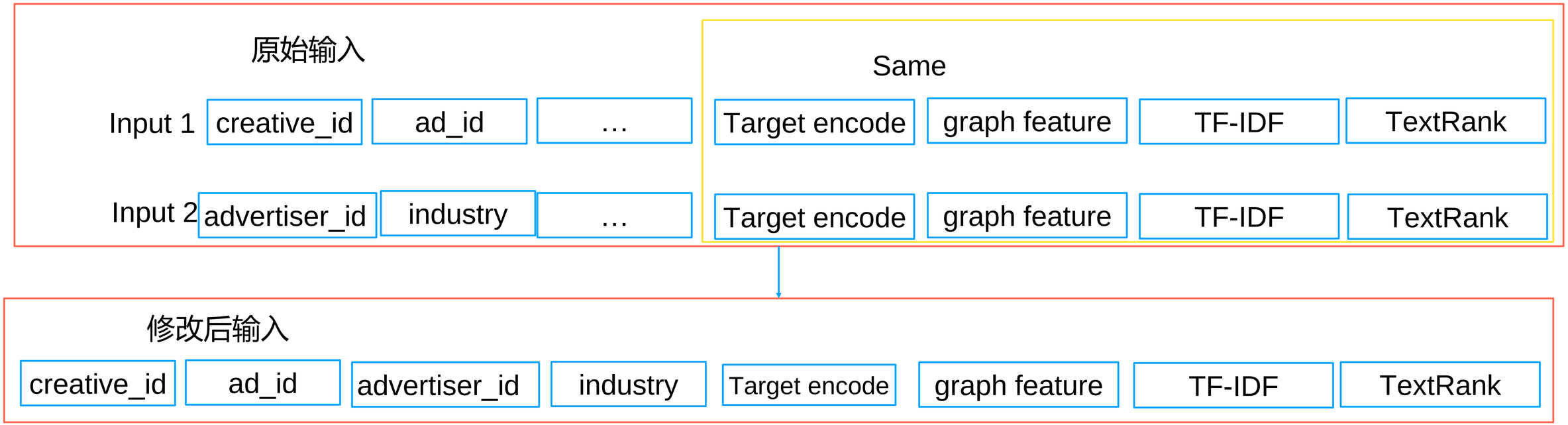
单输入模型：将所有id拼接在一起，修改模型结构适配

[4] Yang R, Zhang J, Gao X, et al. Simple and effective text matching with richer alignment features[J]. arXiv preprint arXiv:1908.00300, 2019.

单输入优化：

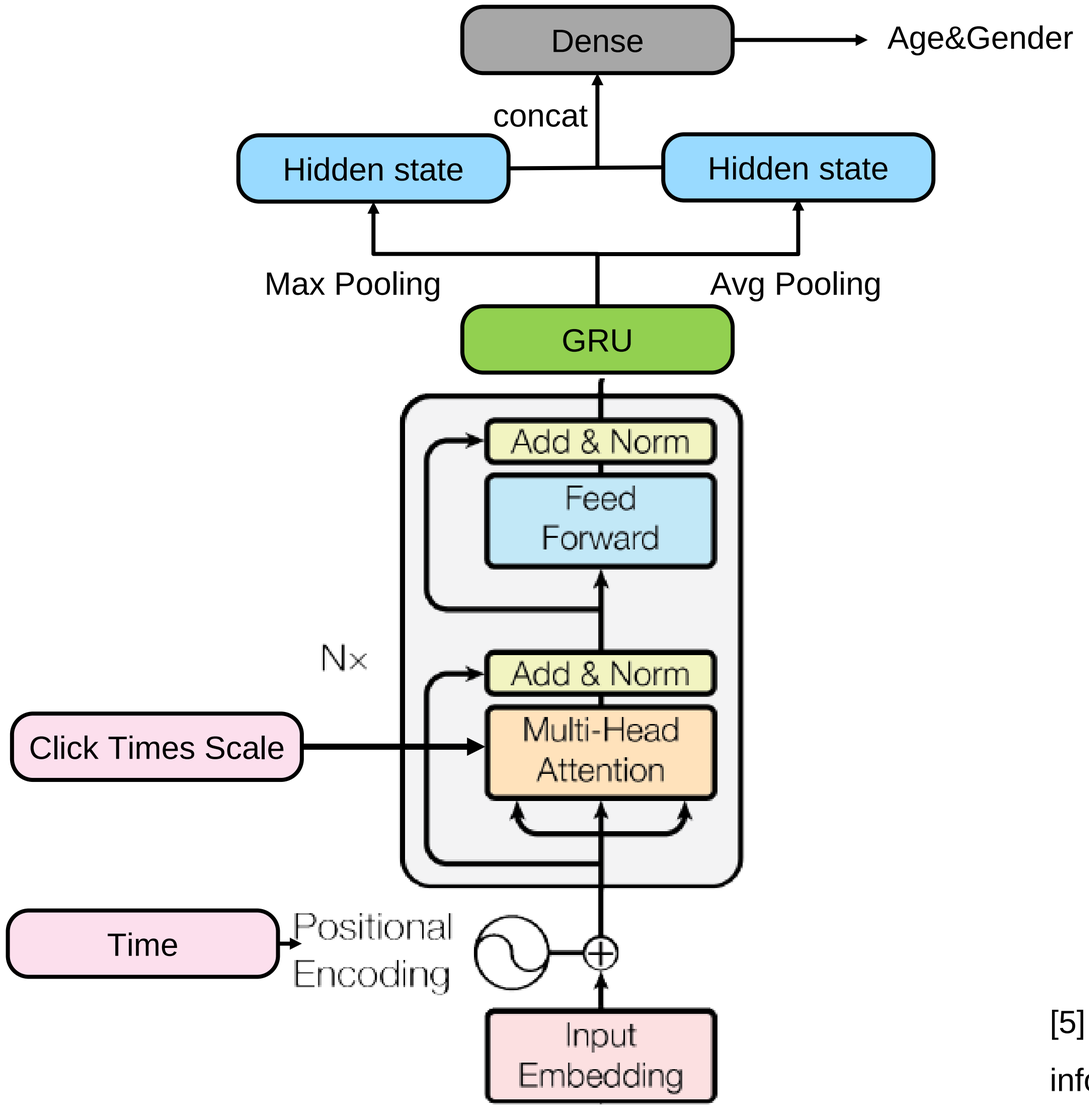
存在问题：

许多**冗余操作**，拼接其他特征导致两个输入序列部分hidden是一样的，使得模型的许多操作重复、失效，比如encode同样的数据，模型Predict 层2个输入的相减、拼接操作。



模型介绍—Transformer

Transformer Encoder：



- 将Text Matching Models中的id组合单输入作为模型输入。
- 使用Times Sequence作为Positional Encoding的index
- 使用Click Times对Self-Attention部分的weight进行缩放，增大点击次数多的行为的影响。
- 使用用户特征拼接在最后的Hidden state上。

[5] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]//Advances in neural information processing systems. 2017: 5998-6008.

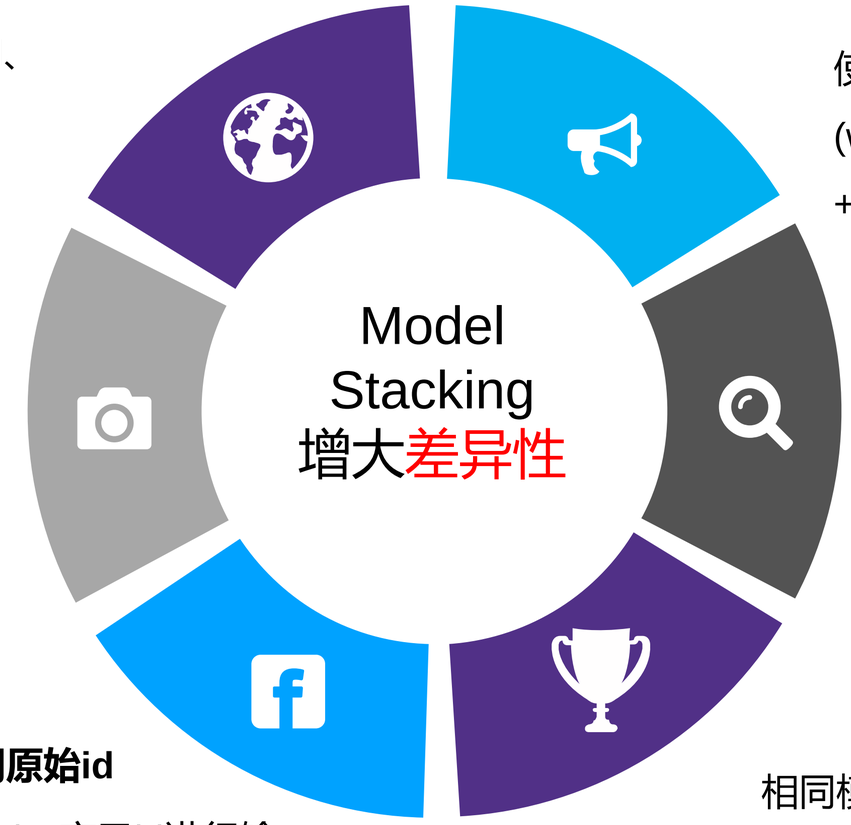
模型介绍-Model Stacking

1.模型

使用不同的模型 (ESIM、RE2、transformer)

2.结构

相同的模型不同的结构改造（单输入、双输入）



3.不同原始id

使用不同的id、交叉id进行输入

4.词向量组合

使用不同的词向量组合 (word2vec, glove, word2vec + glove)

5.特征删减

删去、增加一些特征（图特征、tf-idf stack）

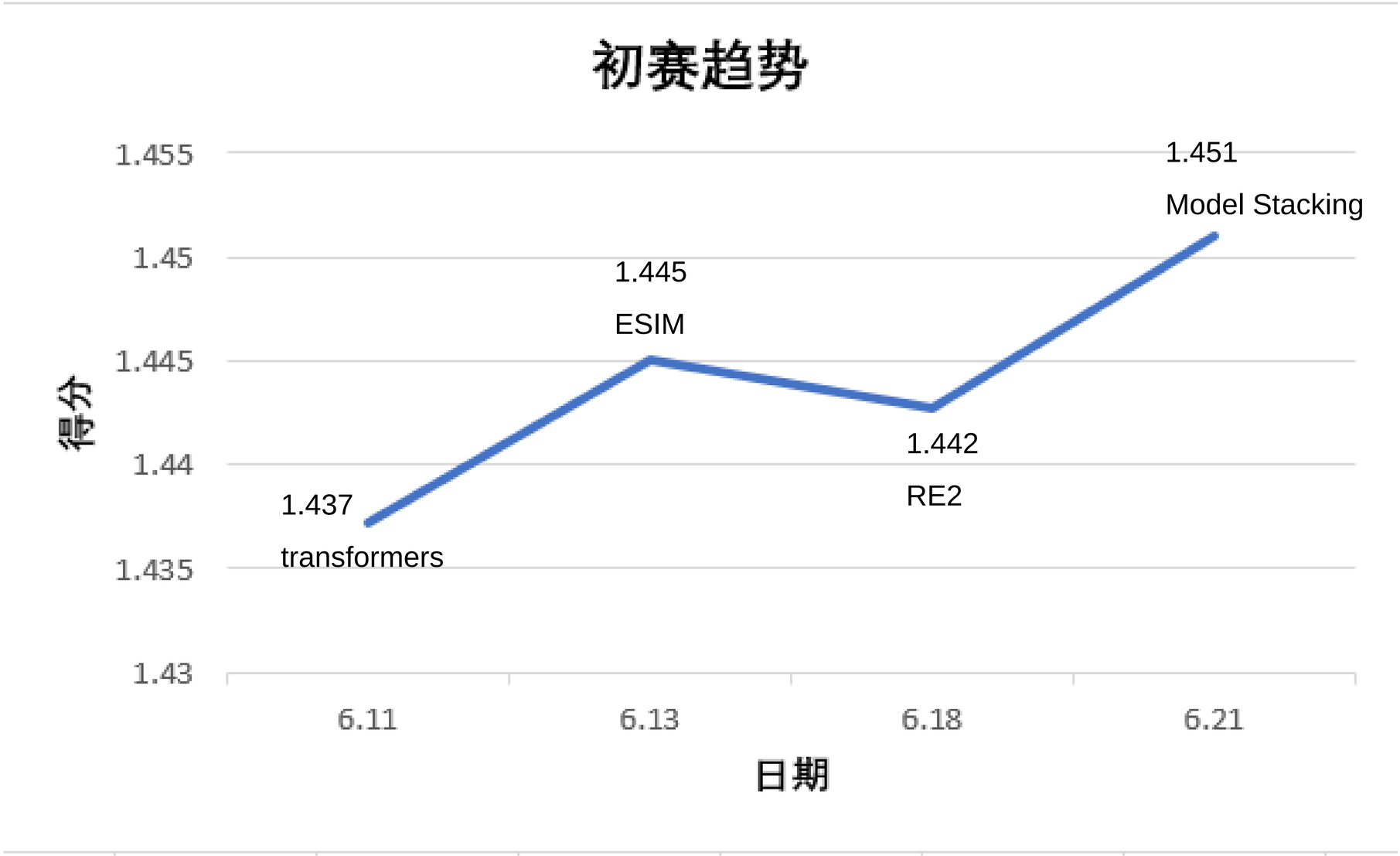
6.模型超参

相同模型不同的learning rate、weight decay/Label Smoothing。

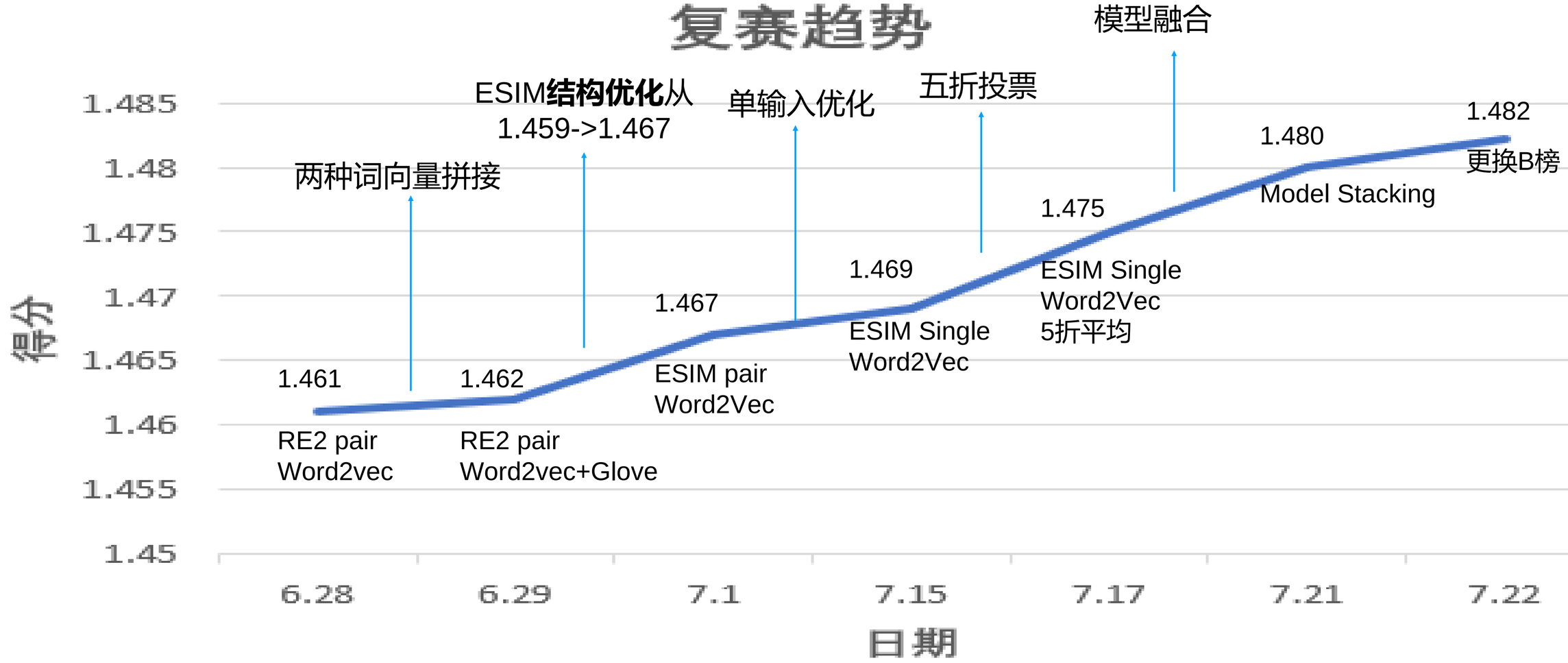
总结与思考

Demo/经验教训

DEMO


初赛趋势
1.455
1.45
1.445
1.44
1.435
1.43
得分
1.437
transformers
1.445
ESIM
1.442
RE2
1.451
Model Stacking
6.11
6.13
6.18
6.21
日期

总结与思考-DEMO


复赛趋势
模型融合
ESIM结构优化从
1.459->1.467
单输入优化
五折投票
两种词向量拼接
1.485
1.48
1.475
1.47
1.465
1.46
1.455
1.45
得分
1.461
RE2 pair
Word2vec
1.462
RE2 pair
Word2vec+Glove
1.467
ESIM pair
Word2Vec
1.469
ESIM Single
Word2Vec
1.475
ESIM Single
Word2Vec
5折平均
1.480
Model Stacking
1.482
更换B榜
6.28
6.29
7.1
7.15
7.17
7.21
7.22
日期

总结与思考-经验教训

- **完全地转化为NLP问题进行解题，缺乏了对用户广告系统行为地深入理解：** 在比赛地过程中，我们基本上完全将这次的用户人口统计特征预测转化为了NLP问题进行解答，没有足够仔细、深入地挖掘到用户在广告系统内行为的规律，设计的模型缺乏可解释性。

- **对推荐系统，计算广告学的常用方法掌握不足：**在设计特征和模型调参的过程中，由于此前缺乏设计推荐系统，解决推荐问题的经验，在推荐系统领域内常用的一些网络模型没有进行尝试。

- **网络调参过程浪费大量时间：**在进行模型训练时，由于缺乏足够的显卡和内存且没有足够的经验，在调节模型超参获取最佳结果这一过程上耗费了大量的时间。

THANKS