知乎

Search Result without Intents

阿里2024最新研究：利用属性建模用户多意图搜索

SmartMindAI
专注搜索、广告、推荐、大模型和人工智能最新技术，欢迎关注我

已关注

Introduction

搜索引擎+是当前最重要的技术之一，它基于查询和文档构建倒排索引+以完成信息检索。随着搜索内容的丰富，现代系统开始存储查询和项目的属性信息，例如品牌、实体和地区等，这些信息可以帮助分类查询或项目是否匹配。

搜索平台上的文本匹配方法很多，其中深度学习方法日益受到青睐，因为它们通常将查询和项目表示为低维向量+。然而，目前大多数方法只是简单地将属性信息整合到文本表示中，忽视了属性的有效性。

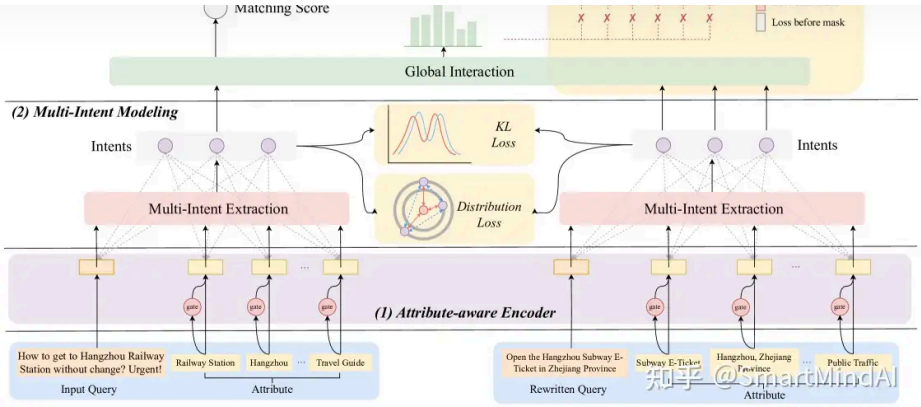
本文提出了"意图"概念，这是一个更具抽象性和代表性的概念，反映了客户的需求。一个合适的重新写的查询或一项合适的产品总是能满足客户的意图。

首先，全面地利用查询和项文本文本属性进行多意图理解；其次，提出了一种意图感知编码器、多意图建模模块和意图感知匹配模块；最后，实现了这种多意图理解并应用于实际的A/B测试中。

Model

Problem Formulation

知乎



• 我们首先引入了一些符号和关键概念。

•为了简化，我们使用一个用户查询与预定义的查询匹配的场景作为例子。

•对于输入查询文本X，其中包含单词x_i，假设存在与查询相关的属性A。

•属性表示为

$$A = \{a_1, a_2, \cdots, a_{n_A}\}$$

其中a_i包含l_i^a个单词

$$\{a_i^1, a_i^2, \cdots, a_i^{l_i^a}\}$$

• 我们的目的是从输入查询识别出重述查询Y，其中包含单词y_j。

•重述查询Y还附加了属性B。

•在多意图文本匹配模型中，首先从文本和属性中提取多个意图，并将这些信息汇总到输入中生成最终匹配分数。

Attribute-aware Encoder

首先，我们需要学习输入文本和属性的语义含义。考虑到注意力机制⁺在文本匹配任务中的优越性能，我们将输入文本与属性连接起来。 然而，我们注意到属性的重要性不仅来自于与其他输入单元的关系，还受到其自身内容的影响。如果属性不传达与匹配相关有意义的信息，它在编码过程⁺中应该扮演较小的角色。

因此，我们提出了一种缩放自注意力模块来模拟匹配对中文本和属性之间的时序交互。 具体来说，我们在连接后的文本前附加一个特殊的标记"$[CLS]$"，并在每个属性和文本前附加一个特殊标记"$[SEP]$"。表示"$[SEP]$"附加到第k个属性的表示为h_k^{Attr}。然后，我们根据h_k^{Attr}获得第k个重要门值g_k：$g_k = \sigma(h_k^{Attr})$，其中$k \in [1, n_A + n_B]$ $\sigma(.)$是一个全连接层，使用sigmoid函数⁺。

接下来，我们将门值融入注意力过程。注意，第k个属性可能包含多个单词，所有单词对应相同的门值g_k。为了简洁起见，在下面的部分中，我们将门值索引从属性⁺索引更改为单词级别索引。 随后，所有输入都通过全连接层⁺转换为查询Q，密钥K和值V。然后，加权模块根据

$$\alpha'_{i,j} = Q_i \times K_j$$

将每个Q_i映射到K_j。软Max注意力机制将门分数g_j加入到门觉察操作中，得到注意力分布$\alpha_{i,j}$，通过公式

$$\alpha_{i,j} = \frac{\exp(g_j \cdot \alpha'_{i,j})}{\sum_{n=1}^{L} \exp(g_j \cdot \alpha'_{i,n})}$$

$$h_i' = \sum_{j=1}^{L} \alpha_{i,j} \times V_j$$

这里，连接的序列长度⁺被计算为

$$L = m + \sum_{k=1}^{n_A} l_k^a + n + \sum_{k=1}^{n_B} l_k^b$$

为了防止梯度消失或爆炸，还应用了层归一化操作在前馈层的输出上。

Multi-Intent Modeling

1. 数据预处理⁺：对数据进行清理、格式转换和数据规范化等操作，以方便后续的学习任务。
2. 特征提取⁺：从原始数据中提取有用的特征信息，例如词性标注、命名实体识别⁺等。
3. 训练模型：选择合适的机器学习算法并训练模型，使其能够根据输入的特征信息预测⁺输出结果。常见的机器学习算法有决策树⁺、支持向量机⁺、神经网络等。
4. 评估模型：使用交叉验证⁺等方法对训练好的模型进行评估，以确保其性能足够好，能够适应实际应用需求。
5. 应用模型

$$w_{c,j} = \mathbf{softmax}([h^X; h_j^A]W^A + b^A),$$
$$I_c^X = w_{c,j}^{\mathrm{T}} \times h_j^A,$$

BERT是利用Transformer架构，通过自编码的方式实现预训练的过程。模型由两个部分组成：编码器和解码器。编码器通过多头注意力机制从文本序列中抽取上下文信息，而解码器则用于生成新的文本序列。 在BERT模型中，我们可以通过添加一个额外的全连接层，将每个单词的嵌入向量传送给这个全连接层，从而获取句子级别的嵌入向量。然后，我们可以使用一个单层的softmax分类器对句子进行分类，以获取句子的语义信息。

除了文本分类⁺外，BERT还可以用于其他类型的文本生成任务。例如，我们可以使用BERT来生成对话。在这种情况下，我们需要首先训练一个BERT模型⁺，然后将这个模型与一个生成器结合起来。生成器⁺可以接收用户输入，并将其转化为语义相似的句子。然后，BERT模型会根据生成器的输出进行微调，以使它更好地理解和生成语义相似的句子。

然而，BERT模型也有其局限性，比如它需要大量的计算资源进行训练，而且在处理某些特定类型的任务时可能存在一些限制。总的来说，BERT模型是一种非常灵活和通用的模型，可以用于各种文本处理任务。

$$\mathcal{L}_{dis} = \mathbb{E}[-\log \frac{e^{cos(I_i, h^*)/\tau}}{\sum_{I_i, I_j \in I^*} e^{cos(I_i, I_j)/\tau}}],$$

对于配对的查询，它们在潜在空间中的意图也应该彼此对齐。因此，我们在训练数据集中提出了一个KL散度损失，该损失的目标是促使正例的意图表示具有相似的分布，同时对于负例对，我们的目标是最大化KL散度。我们选择了针对意图分布而不是直接推动意图表示更近，因为我们强调多个表示之间的"匹配"。

$$\mathcal{L}_{KL} = \mathcal{D}_{KL}\left(q(I^X \mid h^X, h_i^A) \| q(I^Y \mid h^Y, h_i^B)\right),$$

其中q代表生成意图的统计分布。

Intent-aware Matching

获得多个意图表示后，可以利用它们来预测两个输入的匹配程度。具体方法是通过使用注意力机制将意图信息融入文本表示中。需要注意的是，在 § 中，我们为每个输入前面添加了一个CLS标记。在此过程中，我们使用来自属性感知编码器的优化表示h^{cls}作为查询，并将每个输入的意图I^X和I^Y相加以计算意图重要性权重β_j。

$$\beta_j = \frac{\exp\left(h^{cls} \times I_j\right)}{\sum_{t=1}^{2c} \exp(h^{cls} \times I_t)},$$

其中σ表示sigmoid函数。模型训练的目标是最大化每个正目标项相对于其余负项目对该概率分数的影响。

$$\mathcal{L}_{match} = -(s\log(P) + (1-s)\log(1-P)),$$

其中s表示输入匹配（1）或不匹配（0）。一种被称为"意图掩码自监督任务"的方法被提出，用于识别有助于匹配的意图以及在匹配过程中的作用。这种任务通常会逐个遮盖意图，并计算每种情况下的匹配损失。如果损失显著增加，则意味着遮盖的意图对匹配性能有很大影响。因此，模型将会学习给予意图权重较高的结果，从而提高模型在匹配方面的性能。

$$\mathcal{L}_{mask} = \sum_{j=1}^{2c} \|\exp(\mathcal{L}_{new} - \mathcal{L}_{match}),\ \beta_j\|_2,$$

其中$\|\cdot\|_2$表示向量的L2范数。为了优化端到端框架的参数，我们将使用全局损失$\mathcal{L}$。

$$\mathcal{L} = \mathcal{L}_{match} + \mathcal{L}_{dis} + \mathcal{L}_{KL} + \mathcal{L}_{mask}.$$

Experiments

Amazon	number of query-item pairs	1,818,825
	training dataset size	27,757
	valid dataset size	139,306
	test dataset size	425,762
Alipay Query Rewriting	number of rewritten query pairs	32,604
	training dataset size	1,253,757
	valid dataset size	2,347
	test dataset size	2,500
Alipay Query-Item Matching	number of query-item pairs	730,027
	training dataset size	710,027
	valid dataset size	10,000
	test dataset size	10,000

Baselines.

我们对比了我们的模型与三个文本匹配基准方法：BERT、SimCSE和RankCSE。BERT是最常用的预训练语言模型，用于文本匹配；SimCSE是一种对抗学习框架，用于提升句子嵌入；RankCSE则是一种无监督句子表示学习方法，它结合排名一致性、排名离散化和对抗学习。

Implementation Details

- 对于Amazon数据集，我们将截断长度设置为512；
- 对于Alipay数据集，我们将截断长度设置为128；
- 我们将默认的意图数设置为3；
- 模型基于BERT-base进行微调，包含12层和768个隐藏维度，其他参数全部初始化为零均值高斯分布的标准差为0.01；
- 实验将使用批大小为256进行操作；
- 我们将使用Adam优化器作为优化算法，并在训练过程中设置范围为$[-1, 1]$的梯度裁剪；
- 我们将选择性能最佳的5个检查点，并报告测试集的平均结果。

Evaluation Metrics

离线评估时，我们会使用准确性、AUC和F1指标来评估候选匹配的效果。准确性是正确的正例数量占总样本的比例，AUC）是ROC曲线下的面积，F1分数是准确率和召回率的调和平均数，它们都被广泛应用在候选匹配的评估中。

知乎

任务没有明显作用的部分。比如，尽管RankCSE模型在嵌入学习任务上表现出色，但是在匹配任务中，它仍不如一些专门为匹配任务设计的模型效果好。这说明了在匹配任务中，信息融合的重要性是显著的，并且其有效性在匹配任务中至关重要。

此外，具有属性的模型在三个数据集上都表现出了较好的性能，表明属性信息在文本匹配任务中是非常重要的。与BERT-concat、Machop和DCMatch相比，这些模型都具有候选匹配能力，这可能是因为它们能够更好地利用属性。AGREE和MADRAL通过使用属性作为附加监督信号实现了强性能。

最后，我们发现多角度建模在各个方向上都是有效的。MVR是一种具有max-pooling操作的多视图表示方法，而Machop具有多属性结构感知池化。它们两者都比其他基准线实现的好。最后，MIM在所有数据集和不同度量标准下都取得了最好的性能。

	Amazon			Alipay Query Rewriting			Alipay Query-Item Matching		
	Accuracy	AUC	F1	Accuracy	AUC	F1	Accuracy	AUC	F1
BERT [9]	0.7315	0.7767	0.8145	0.8560	0.9092	0.6944	0.8051	0.8722	0.8127
SimCSE [8]	0.7423	0.7882	0.8213	0.8577	0.9104	0.7003	0.8147	0.8736	0.8143
RankCSE [21]	0.7495	0.8071	0.8247	0.8649	0.9192	0.7218	0.8154	0.8785	0.8166
MVR [38]	0.7492	0.8036	0.8275	0.8622	0.9137	0.7151	0.8137	0.8773	0.8182
MADRAL [11]	0.7624	0.8177	0.8229	0.8670	0.9188	0.7332	0.8142	0.8758	0.8184
BERT-concat	0.7434	0.7941	0.8238	0.8578	0.9197	0.7102	0.8168	0.8764	0.8195
AGREE [31]	0.7681	0.8193	0.8282	0.8624	0.9129	0.7257	0.8169	0.8797	0.8213
Machop [35]	0.7705	0.8314	0.8303	0.8695	0.9136	0.7384	0.8213	0.8903	0.8340
DCMatch [40]	0.7732	0.8356	0.8343	0.8700	0.9208	0.7399	0.8202	0.8847	0.8366
MIM	**0.7813**	**0.8449**	**0.8413**	**0.8792**	**0.9324**	**0.7506**	**0.8343**	**0.9012**	**0.8431**

具体而言，MIM在Amazon上的AUC得分超过DCMatch0.0091，Alibaba查询重写数据集上的AUC得分超过0.0116，Alibaba查询项匹配任务上的AUC得分超过0.0165。所有系统之间的所有对等比较都使用t检验（p值为0.01）进行统计学显著性测试。上述观察表明，我们的意图提取方法✦更有效地使用了属性，而我们设计的意图对齐机制提高了文本匹配性能。

Online Experiments

除了离线实验，我们还在Alipay搜索系统中部署了MIM进行在线A/B测试。 在对照组中，使用了与当前系统类似（即采用交叉编码结构）的匹配策略。 在测实验组中，提出了MIM来处理候选查询的重写匹配和查询项的相关性匹配。

使用相同的召回率和排名策略进行公平比较，并将平均AB测试结果报告在表中。

Method	Alipay Query Rewriting			Alipay Query-Item Matching		
	pvCTR	GR	BR	pvCTR	GR	BR
online	+0.0	+0.0	-0.0	+0.0	+0.0	-0.0
MIM	**+0.11%**	**+11.95%**	**-13.43%**	**+0.84%**	**+2.32%**	**-1.15%**

结果显示，MIM在所有在线评分指标上都提供更优的表现，具体而言，在好案例率和坏案例率方面，均远超现有系统，这提升了用户搜索体验。 在两种情况下，MIM分别实现了相对提升0.11%和0.84%的pvCTR。 因此，可以说MIM为Alipay的数百万用户提供更好的搜索结果。

原文《Multi-Intent Attribute-Aware Text Matching in Searching》

编辑于 2024-03-11 21:12 · IP 属地北京

搜索引擎　意图识别　检索

理性发言，友善互动