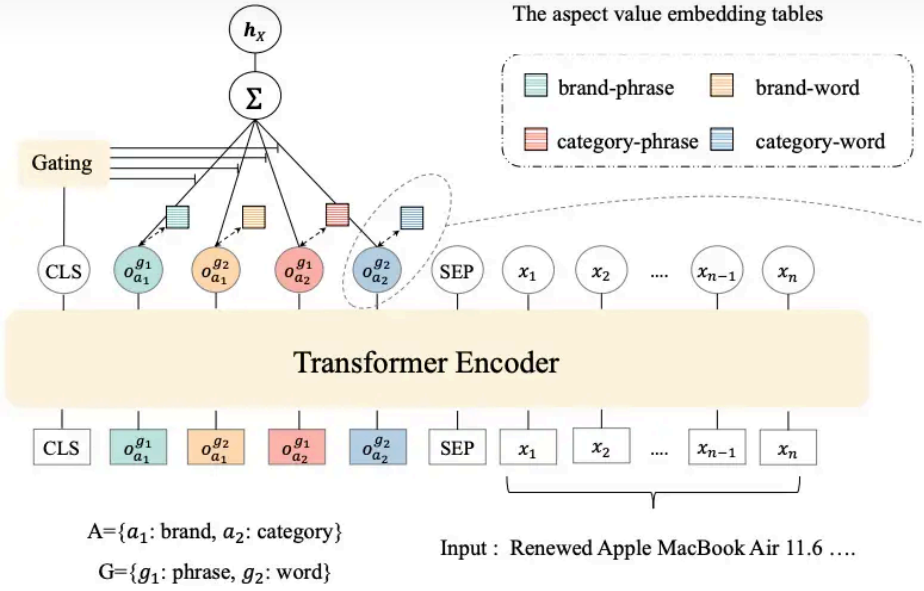
知乎



阿里2023最新研究：揭秘推荐系统中多维度的向量召回技术

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

9 人赞同了该文章

Introduction

近年来，向量检索方法在信息检索（IR）和自然语言处理（NLP）社区中得到了广泛的研究。然而，这些方法大多针对非结构化数据，并没有研究如何有效地利用结构化数据中的属性信息，例如产品类别和人与机构的关联。例如，在图中，"运动手套"查询的目标是用于体育使用的手套，所以厨房手套应该避免。显然，四个内容的类别可以帮助区分各种类型的手套并提高检索性能。

Query: "sports gloves"			Relevance
Items	Title	Aspect: Category (Phrase-Level)	
i1	ATERCEL Weight Lifting Gloves Full Palm Protection, Cycling, Exercise,…	Exercise & Fitness;	😀
i2	Hestra Army Leather Heli Ski Glove - Classic 3-Finger Snow Glove for Skiing, …	Sport Specific Clothing;	😀
i3	HEAD Leather Racquetball Glove - Web Extra Grip Breathable Glove…	Tennis & Racquet Sports;	😀
i4	HSL 2 Pairs Reusable Kitchen Dishwashing Gloves, Waterproof, Non-Slip, Gardening,…	Household Supplies;	😟

Figure 1: An example of a query and its candidate items.

最近，@madr提出了一种有效的多维度的向量检索方法，即MTBERT和MADRAL。这些方法遵循一种典型的模式，学习维度嵌入并通过预测其相关值的辅助目标进行。一个具体的例子是图中"类别"的维度嵌入i4将通过预测其值"家用用品"来学习。虽然有效，但他们考虑了某个维度的值作为孤立的类，并忽视了不同值之间的潜在相关性，这可能会导致性能不佳。以图中的内容为例，尽管它们属于四个不同的类别，但前三者与用户查询"运动手套"有关，而最后一个是无关的。辅助目标预测他们的分类ID将每个类别视为平等对待，可能无法捕捉到它们的精细关系。注意到该问题，我们提议使用词和标记级别代替短语级别作为粒度。图中，分解类短语为小块，会使得第一、二、三的连接更为清晰，因为它们都与运动相关。

我们的模型名为BABY，意为多粒度感知维度的学习模型。Transformer层前、后分别加入单独的维度嵌入作为输入，如图所示。顶层维度嵌入通过各个粒度级别监督价值预测指导，即短语、单词和标记。与其他方法相比，Babay具有多个优势：

1）与混合在"CLS"中的内容维度信息不同，Babay分别表示两种类型的信息，通过门机制增强交互；

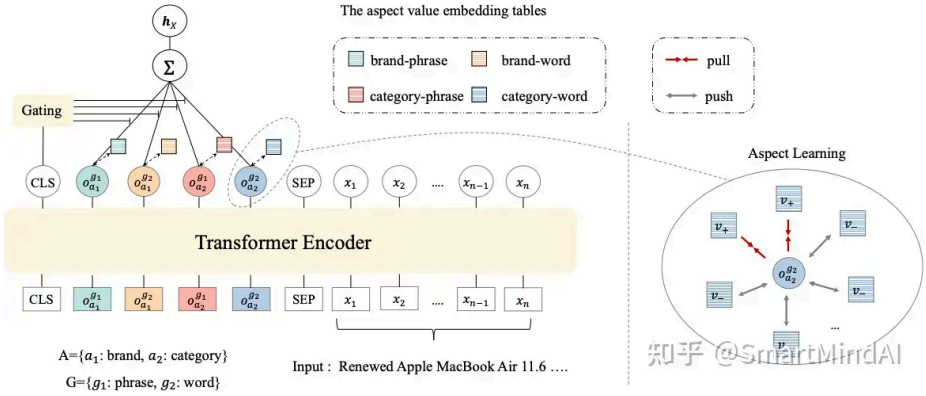
知乎

3）Babay整合所有粒度级别维度信息，捕捉不同粒度层次维度的语义关系，有利于检索性能提升。我们在两个真实世界搜索数据集上进行了大量实验。

结果显示，我们的方法在两个数据集上均优于竞争性基准。值得一提的是，即使没有维度注释监督，我们的模型也能够取得令人印象深刻的结果，这意味着即使不使用维度信息，Babay也可以学习到有用的隐含表示。针对不同粒度的ablation研究揭示，每种粒度都对多维度检索性能有所贡献，同时融合所有粒度则能得到最佳效果。

METHODOLOGY

我们提出一个多角度稠密检索的MGL模型，并介绍了其核心组成部分。如同所示，MGL基于BERT构建。由于MGL以相同的方式编码项和查询，我们仅用项进行说明。



Aspect Representations

合理地在预训练模型+中表示视角对于指导有效训练至关重要。为了充分利用Transformer编码器的能力，如图所示，我们在内容词之前引入了几种tokens来从各种角度看视图维度。这与从content获取 CLS 的方法相吻合，并且这些tokens可以与内容词充分交互。

在预训练期间，这些插入的嵌入可以作为一种不同的context视图，在预测内容中的掩码+token时起作用。通过这种方式，这些嵌入也可以学习到掩码语言模型目标，从而捕捉来自不同隐含观点的内容语义，这可能会带来更多的好处，特别是在没有维度值注释的情况下。

Aspect Learning

为了简单起见，我们使用一个领域为a的粒度为g的示例来说明。有两部分重要的组件：值表示和目标学习目标。针对粒度为g的领域知识表示，我们可以有两种选择：粒度扩展和低级集成。粒度扩展通过增加更多细节信息来提高粒度表示的准确性，而低级集成则通过融合不同粒度的信息来增强表示的泛化能力。当我们获得表示的粒度为g的维度为H的特性a，用$\mathbf{h}_a^g(X)$表示时，我们采用广为使用的组内对比损失来预训练编码器。 它旨在使源表示更接近目标组中的实例，并使其与其他组的表示相距甚远。

$$\mathcal{L}_a^g(X) = -\frac{1}{|\mathcal{A}_a^g|}\sum_{v^+\in\mathcal{A}_a^g} log\frac{exp(sim(\mathbf{h}_a^g(X),\mathbf{e}_{v^+}))}{\sum_{v\in V_a^g} exp(sim(\mathbf{h}_a^g(X),\mathbf{e}_v))}$$

其中$\mathbf{e}_v/\mathbf{e}_{v^+}$是$E_a^g$中特性的值嵌入$sim(\cdot)$是点积函数$\mathcal{A}_a^g$是粒度为$g$的特性值注释集，用于特性$a$。

Multi-Granularity-Aspect Grouping

假设存在$|A|$个维度和$|G|$个粒度，目标是针对每个粒度的每个维度设置$|A| \times |G|$个学习目标。为避免信息过度压缩限制学习效果，我们提出三种分组方案：基于单目标的分组、基于粒度的分组和基于维度的分组。在仅有少量维度和粒度时，基于单目标的分组会在输入序列X中引入$|A| * |G$个tokens以从不同视角捕捉内容语义，每个token对应一个学习目标。具体序列形式为：

知乎

并利用最后一层隐藏向量$\mathbf{h}(o_k)$表示维度a_i在特定粒度下的内容含义。

$$\mathcal{L}_{\mathcal{A}}(X)=\frac{1}{|A|*|G|}\sum_{i=1}^{|A|}\sum_{j=1}^{|G|}\mathcal{L}_{a_i}^{g_j}(X).$$

当$|A|*|G$很大时，添加大量tokens会影响原始输入的语义表示。因此，需要进一步按照不同粒度和多个维度对目标进行分组。 细粒度分组。同一种粒度表示相同级别的语义信息，以同一粒度分组目标是合理的。在这种情况下，G个token会被插入到输入序列中，得到

$$X=(x_0,o_1,\ldots,o_{|G|},x_1,x_2,\ldots,x_n)$$

这些token的编码表示为$\mathbf{h}(o_j)$（$j=1,\ldots,|G$），代表所有维度信息的角度。一个维度嵌入可以表示粒度为g_j的所有维度损失$\mathcal{L}_{a_i}^{g_j}(X)$即所有$g_j$粒度的所有维度$h_{a_i}^{g_j}$（$i=1,\ldots,|A$）都相同。维度学习的目标是：通过调整和优化这些嵌入参数来最小化

$\sum_{i=1}^{|A|}\sum_{j=1}^{|G|}\mathcal{L}_{a_i}^{g_j}(X)$

$\mathcal{L}_{\mathcal{A}}(X)=\frac{1}{|G|}\sum_{j=1}^{|G|}\mathcal{L}_{g_j}(X),\text{where }\mathcal{L}_{g_j}(X)=\frac{1}{|A|}\sum_{i=1}^{|A|}\mathcal{L}_{a_i}^{g_j}(X)$

跨多粒度-维度的目标进行分组的一种方式是在维度的基础上按维度进行分组，这样不同的维度信息就不会混杂在一起，并且各种粒度层次可以从彼此中受益。 在这个模型中，我们引入A个引导令牌在内容令牌之前：

$$X=(x_0,o_1,\ldots,o_{|A|},x_1,x_2,..,x_n)$$

每个维度都有自己的隐藏向量$\mathbf{h}(o_i)$（$i=1,\ldots,|A$）。这些隐藏向量捕获了对应于维度a_i的输入项的各个粒度的信息表示。 特别的是，在使用方程计算损失$\mathcal{L}_{a_i}^{g_j}(X)$时，维度$a_i$的表示$\mathbf{h}_{a_i}^{g_j}$在整个粒度层次上保持一致。在这种聚合方法下，损失$\mathcal{L}_{\mathcal{A}}$可以重新表述为以下形式：

$\sum_{i=1}^{|A|}\sum_{j=1}^{|G|}\mathcal{L}_{a_i}^{g_j}(X)$

$\mathcal{L}_{\mathcal{A}}(X)=\frac{1}{|A|}\sum_{i=1}^{|A|}\mathcal{L}_{a_i}(X),\text{where }\mathcal{L}_{a_i}(X)=\frac{1}{|G|}\sum_{j=1}^{|G|}\mathcal{L}_{a_i}^{g_j}(X)$

为了适应许多维度和粒度的情况，我们可以通过粒度或维度划分来减少指导令牌的数量。这样做可以使模型的架构保持不变，只是在同一粒度或维度上将学习目标放在共享的令牌上处理。

Aspect Embedding Fusion

考虑到效率，我们需要将多个嵌入结合在一起，以便最大限度地减少存储和计算成本。受到启示，在 CLS -Gating深度融合机制。为了说明融合过程，我们举了一个使用单个目标划分方法的例子，该方法已经在第一节中讨论过了。 具体来说，我们将 CLS 嵌入经过线性层和softmax函数计算得到权重分数，用于

$$\mathbf{h}(o_1),\ldots,\mathbf{h}(o_K)$$

其中$K=|A|*|G$。

$\mathbf{w}=Softmax(U\mathbf{h}(x_0)+b)\in\mathbb{R}^K$

在这个过程中，

$U\in\mathbb{R}^{K\times H}$

和$b\in\mathbb{R}^K$是可以调整的参数。然后，我们会利用这些学到的权重来融合多个嵌入，从而获得输入X的最终编码表示。

$$\mathbf{h}_X=\sum_{k=1}^{K}w_k\cdot\mathbf{h}(o_k).$$

根据上述论述，在之前的研究中，掩码语言模型（MLM）任务可以帮助建立优秀的文本表示，因此，我们也采用了MLM作为除维度学习之外的预训练目标。

$$\mathcal{L}_{MLM}(X) = -\sum_{w \in masked(X)} logP(w|X_{\backslash masked(X)}),$$

其中X代表输入句子$masked(X)$和$X_{\backslash masked(X)}$分别表示被遮蔽的标记和剩余的标记。接着，我们将维度学习损失和MLM损失联合起来对Transformer编码器进行预训练，如下所示：

$$\small

$$\mathcal{L}_{total}(X) = \mathcal{L}_{MLM}(X) + \lambda \mathcal{L}_{\mathcal{A}}(X), $$

其中λ是一个超参数[+]。接下来是关于微调的部分。我们在微调过程中使用batch内softmax交叉熵[+]损失$\mathcal{L}_{SCE}$作为学习目标。需要注意的是，尽管维度学习损失也可以添加到微调过程中，但我们的实验结果显示所有多维度的检索器都没有显著的改进。因此，在本文中省略了这个目标。

$$\mathcal{L}_{SCE} = -log\frac{exp(sim(\mathbf{h}_Q,\mathbf{h}_{I^+}))}{exp(sim(\mathbf{h}_Q,\mathbf{h}_{I^+}))+\sum_{I^-} exp(sim(\mathbf{h}_Q,\mathbf{h}_{I^-}))}$$

EXPERIMENTAL SETTINGS

Datasets

我们使用了两个来自于真实世界的大型搜索数据集进行了实验。这些数据集的统计信息已经在表格中列出了。

	MA-Amazon	*Alipay*
aspect	*item*	*item / query*
brand	94% (5k,6k,5k)	0.6%/44% (9k,11k,3k)
color	67% (2k,1k,1k)	–
category	87% (8k,5k,5k)	90%/91% (457,650,548)

第一个数据集是多角度亚马逊ESCI数据集（MA-Amazon）。在这个数据集中，我们加入了亚马逊ESCI数据集中的产品类别信息。在MA-Amazon 中，只有产品有品牌、颜色和类别注释。产品语料库[+]包含了482K个不同的产品，用于预训练。检索数据集包括17k、3.5k和8.9k个查询，用于微调、验证和测试，且没有查询重叠。对于每个查询，检索数据集提供了平均20.1个商品及其与给定查询相关的ESCI相关性判断（*准确*，*替换*，*补充*，*不相关*），这表明每个商品对给定查询的相关性。我们将*准确*处理为正例，其余所有为负例进行微调和需要二进制标签的度量。 第二个数据集是Alipay搜索数据集。这个数据集与亚马逊ESCI数据集有所不同，它是一个中文迷你程序（类似服务）搜索数据集。在这个数据集中，查询和商品都带有两个维度。

Baselines

1. BIBERT是一种标准双编码器基准，其主要作用是对文本类型进行编码。BIBERT采用BERT的方法来进行预先训练，并在此基础上进行微调，其中损失函数为$\mathcal{L}_{SCE}$。
2. Condenser是一种专门针对无结构文本向量检索的预训练方法。它通过引入一种中间层token来优化文本向量检索，同时使用相应的头层token来实现短路，从而使得该方法在预训练过程中可以更好地封装信息。
3. BIBERT-CONCAT将维度值视为文本，并将它们与查询/内容的内容进行concatenate操作。在进行微调时，由于concatenate查询可能会改变查询语义，所以只有内容的维度特性被用于相关性匹配。
4. MTBERT是一种基于BIBERT的多任务学习模型，除了在预训练阶段进行MLM操作之外，还会对$|\mathcal{A}$个维度预测任务进行任务处理，而在这个过程中，使用的是 CLS 编码器。
5. MADRAL则引入了维度提取注意力网络来提取$\mathcal{A}$个维度代表用于查询和内容。这些嵌入是从维度预测任务中学习得到的，并在预训练阶段被融合起来，从而获得了最终的表示来进行微调。

知乎

我们自行实现了一个名为Baby的基准，以确保在实施细节和比较过程中的公平性。

Pre-training

为了实现知识共享，我们将使用预先训练好的编码器，并对其进行微调。在训练过程中，我们会使用包括内容语料库或查询和内容语料库在内的多个级别值集。这些值集包含了每个维度在不同级别的词汇表+，其中包括单词和令牌级别词汇表。

Fine-tuning

我们使用Tevatron工具包对所有模型进行了20次迭代的微调。在微调过程中，我们不仅考虑了批次内的负样本，还额外增加了每个查询的硬负样本。我们采用学习率5e-6和批大小为64进行训练。此外，我们的最大令牌长度设置为32查询和156内容。所有模型均采用相关性损失$\mathcal{L}_{SCE}$进行训练。

Evaluation Metrics

我们报告了R\@100、R\@500和NDCG\@50的结果。然后，我们针对每种类型的查询将各自的加权分数设为1.0、0.1、0.01和0.0。这发生在MA-Amazon平台上。我们进行了两次独立的t检验（p\<0.05），以检测是否存在显著差异。

EXPERIMENT RESULTS

Studies on Model Variants

为了保证结果的再现性，所有实验都基于公开可用的MA-亚马逊数据集进行。

Method	MA-Amazon			Alipay		
	R@100	R@500	NDCG@50	R@100	R@500	NDCG@50
BIBERT	0.6075	0.7795	0.3929	0.4464	0.6284	0.2033
Condenser	0.6091	0.7801	0.3960	0.4520	0.6423	0.2072
BIBERT-CONCAT	0.6137	0.7814	0.4005	0.4517	0.6291	0.2103
MTBERT	0.6137	0.7852	0.3969	0.4498	0.6280	0.2064
MADRAL	0.6088	0.7815	0.3950	0.4506	0.6383	0.2057
MUR	$0.6282^{\dagger\ddagger*}$	$0.7943^{\dagger\ddagger*}$	$0.4151^{\dagger\ddagger*}$	0.4556^{*}	$0.6458^{\ddagger*}$	0.2046
MURAL	$\mathbf{0.6371}^{\dagger\ddagger*}$	$\mathbf{0.8023}^{\dagger\ddagger*}$	$\mathbf{0.4228}^{\dagger\ddagger*}$	$\mathbf{0.4630}^{\dagger\ddagger*}$	$\mathbf{0.6519}^{\dagger\ddagger*}$	$\mathbf{0.2177}^{\dagger\ddagger*}$
MURAL-CONCAT	$0.6389^{\dagger\ddagger*}$	$0.8005^{\dagger\ddagger*}$	$0.4281^{\dagger\ddagger*}$	$0.4669^{\dagger\ddagger*}$	$0.6474^{\ddagger*}$	$0.2124^{\dagger*}$

我们在进行角度学习时，研究了一种使用相同数量的令牌在输入序列的开始部分进行角度学习的方法，名为$\mathbf{baby}^{first_k}$。结果表明$\mathbf{baby}^{first_k}$与表中的最佳基准性能相近或更好，但低于$\mathbf{baby}$的最佳变体。这说明多粒度敏感的角度学习是有益的，但需要使用单独的指导令牌进行学习。 为了探究$\mathbf{CLS}$-Gating对于角度嵌入融合的影响，在

$\mathbf{baby}^{no\_clsgating}$

中，我们移除了$\mathbf{CLS}$-Gating并使用$\mathbf{CLS}$表示作为最终表示。相比表格中的最佳基准$\mathbf{CLS}$-Gating会使最终表示的表现更好，但略逊于最佳变体。这表明融合应该以适当的比例对最后的角度嵌入进行融合。

论文原文《A Multi-Granularity-Aware Aspect Learning Model for Multi-Aspect Dense Retrieval》

编辑于 2024-03-08 10:09 · IP 属地北京

搜索　搜索引擎　推荐系统算法实践（书籍）