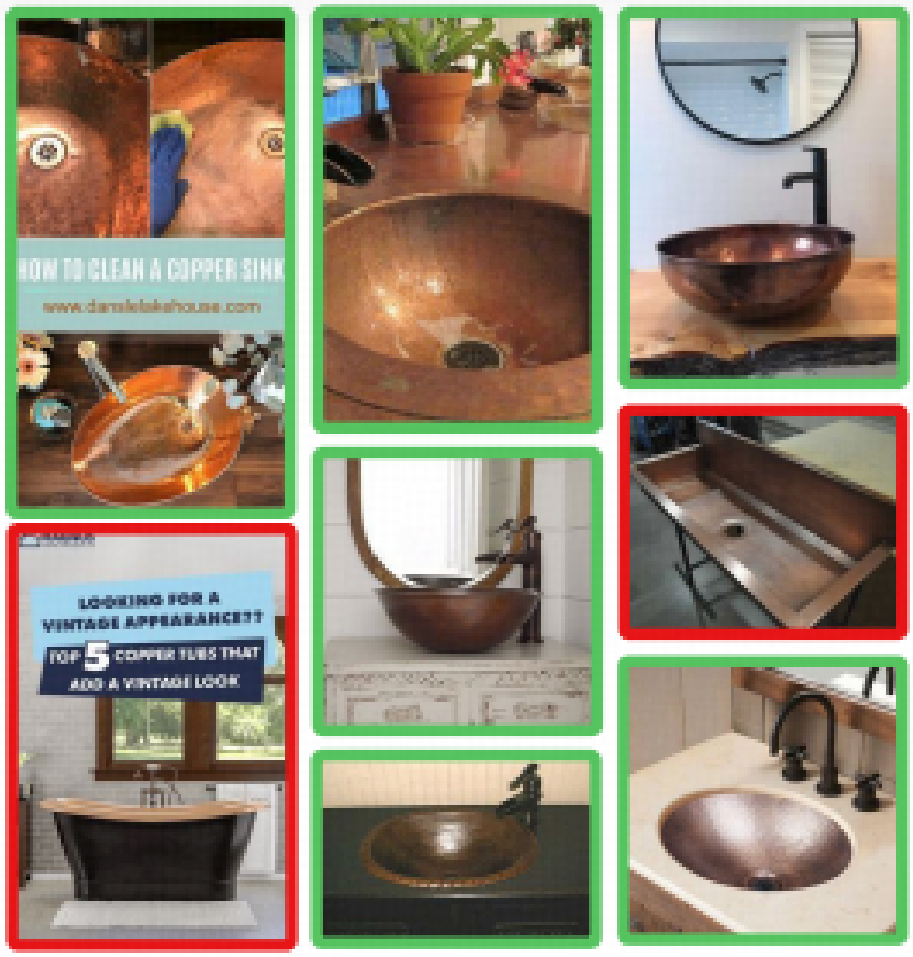
知乎



2024-Pinterest：多任务多实体embedding提升搜索效果

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

18 人赞同了该文章

Introduction

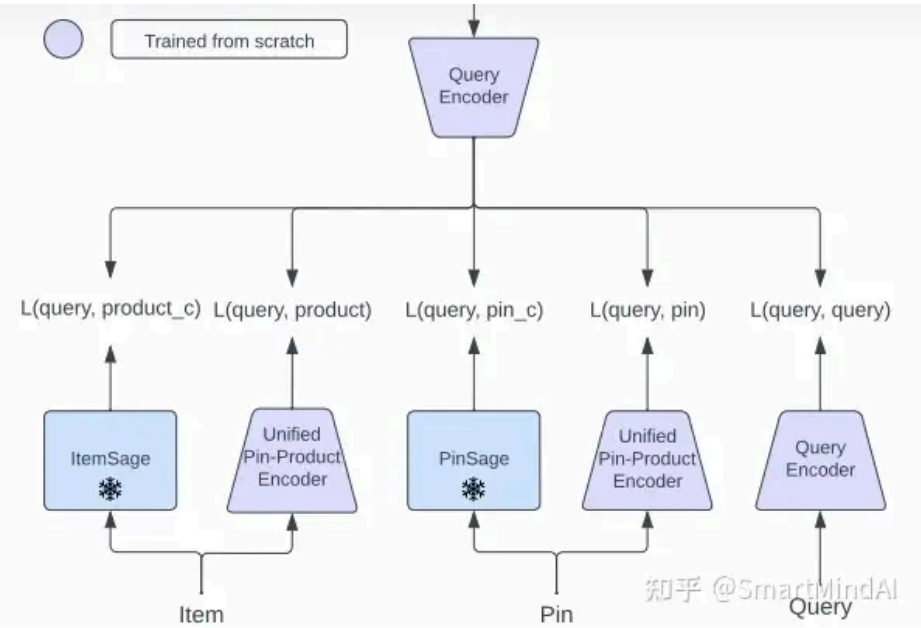
Pinterest的使命是为每个人带来创造他们所爱生活的灵感。搜索是Pinterest上用户获取灵感的关键界面，这些灵感覆盖了从家居装饰、婚礼策划到美容和时尚的最新趋势等广泛兴趣领域。为了提升搜索体验，现代搜索系统旨在并入各种类型的内容，包括网页文档、新闻、购物项目、视频等。同样，Pinterest的搜索流包含了多样的内容，包括书签、购物项目、视频书签以及相关查询。

在推荐系统[+]中，通过构建嵌入（embeddings）来理解内容与查询，借助近邻搜索（ANN）实现高效检索。然而，为不同应用单独构建嵌入可能导致显著的模型管理成本和潜在推荐质量的下降。通过严谨的离线实验，我们在构建Pinterest大规模搜索中嵌入的关键决策的影响上展示了显著的成效：

- 通过集成生成型语言模型（LLM）生成的图像描述、历史参与行为以及用户自定义的版块，可以极大地丰富书签和产品的表现形式，引入多样化的文本描述，从而提升用户体验和个性化推荐的精度。
- 单一查询嵌入能够在检索查询、产品和书签时，几乎达到与专用嵌入同等的效果。
- 单一查询嵌入不仅能适用于检索查询、产品和书签，还能兼容多种现有嵌入，且在任务比较中展现出良好性能。

Method

知乎



Problem Formulation

现代搜索系统致力于整合多样化的内容，如网页文档、新闻、商品、视频等，以提升用户体验，Pinterest的搜索流程包含书签、商品、视频书签及查询等多个内容类型。传统的策略是为每个内容类别及其特征定制查询嵌入模型，这往往耗时且效率不高。为此，我们提出了一种名为OmniSearchSage 的统一查询嵌入模型，该模型能够在联合训练下处理查询间、查询与书签、查询与商品的检索与排序。

在实际操作中，考虑到与已有嵌入系统的兼容性需求，我们还训练了查询嵌入方式，使其能够与当前实体的预设嵌入配套使用，这种设计不仅有助于系统高效运行，简化迁移流程，同时借助余弦相似性+法则的三角不等性质，还能增强与某些嵌入的兼容性，进一步优化搜索效果。

Enriching Entity Representations

Synthetic GenAI Captions

在平台上，约有30%的书签缺少或具有不相关、杂乱的标题或描述。为解决这一问题，我们采用了现成的图像描述生成模型+BLIP来为这些图片生成合成描述。为了评估这些生成描述的质量，我们进行了人类评估，结果显示87.84%的描述既相关且质量高。在收集的样本集中，这些评估标准涵盖了10k张图片，这些图片均匀分布在各种广泛的书签类别中。这些合成描述作为额外特征融入模型中，显著丰富了与每个实体关联的数据多样性。虽然这些描述对用户不是直接可见的，但它们对深入了解书签内容起到了关键作用。

Board Titles

在Pinterest平台上，用户探索并保存书签到他们的个人收藏中，通常被称为板。每一板都配有一个相关的标题，反映收藏的主题或主题。大多数情况下，这些由用户精心制作的板都是有条理的，每个专注于一个特定的主题或目的。例如，用户可能会创建专门用于"社交媒体营销"和"图形设计"的单独板。因此，这些板的标题为相应板内的书签提供了有价值、用户生成的描述符。

我们通过收集每个保存在书签（或产品）中的所有板的标题来利用这些由用户整理的信息。我们限制选择，为每个书签（或产品）挑选最多10个唯一板标题，系统性地去除任何可能的噪音或冗余标题，如下所述。首先，为每个标题分配一个分数，该分数受到两个因素的影响：其出现频率以及构成其的词的普遍性。接着，根据它们的分数（从低到高排序）、词数（从多到少排序）和字符长度（从多到少排序），对这些标题进行排名。接下来，从最顶部得出的10个板标题被随后用作我们模型的一个特征。这一过程消除了任何可能的噪音或冗余标题于特征中。

Engaged Queries

表，该列表包含与之交互的查询以及这些交互的类型和数量。接着，这些列表根据每种交互类型计数进行排序，我们从排序后的每种计数中选取排名前20的查询，作为模型的特征。

在实验过程中，我们观察到使用较大的时间窗口可以提高性能，最终选择了两年作为计算特征的时间窗口。然而，频繁重算的计算成本成为一个问题。为了解决此问题，我们实施了一种渐进更新策略:每隔n天，我们更新新查询日志，并为每个书签生成一个包含最新查询的列表。然后，将这个新列表与当前最新的前20个查询列表合并，以实时更新特征值✦，确保它们始终反映最新的用户互动情况。

Entity Features

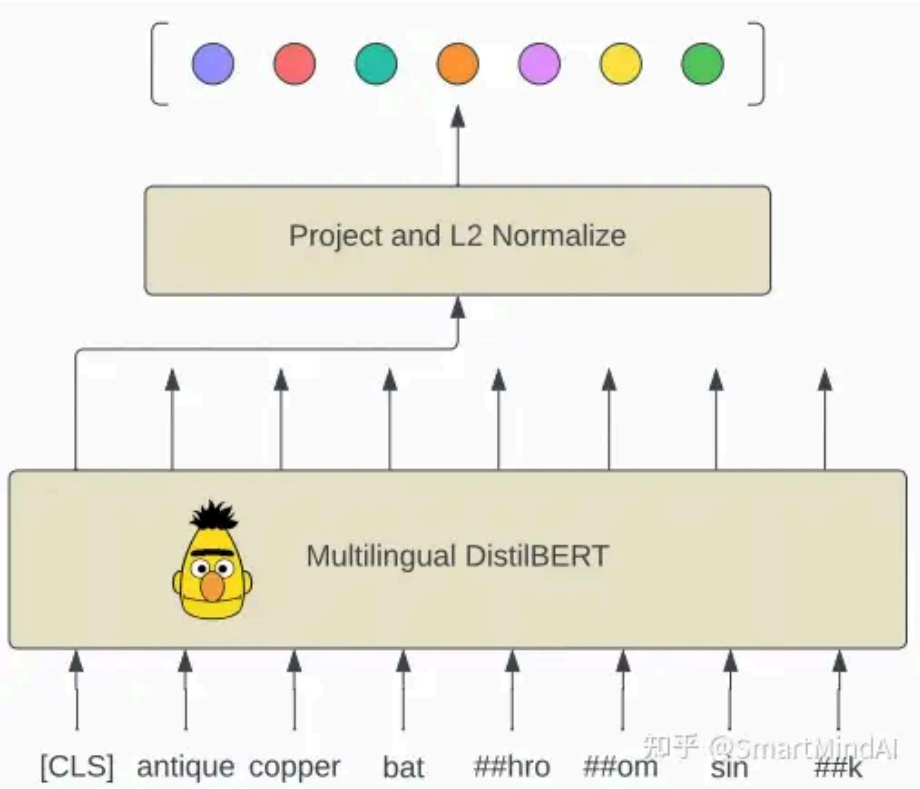
我们整合了一系列特性，旨在全面捕捉书签的核心信息。这包括PinSage 与统一图像嵌入，能够有效提取图片的关键特征。对于包含商品内容的书签，我们引入了ItemSage ，它特别适用于处理具有商品相关的书签。书签标题、描述等基于文本的特性也已成为我们特征集中的关键元素。更进一步，通过引入前文中所述的合成标题、版块标题以及根据互动查询生成的描述，我们增强了与每张书签关联的文字信息。通过综合运用以上所有特性，我们实现了对书签的全面且多层次表示，以此促进对表示的学习优化。

Encoders

在研究中，我们处理了三种核心实体:书签、商品和查询。构建了一个模型体系，包含:

1. 查询编码器:专门用于编码用户的查询。
2. 联合学习编码器:负责同时编码书签和商品，通过共享知识实现两者的相互增强。
3. 专用兼容性编码器:独立针对书签和商品，确保各自特征的有效提取与整合。这样的架构设计旨在全面提炼实体间的相互关系和特性。

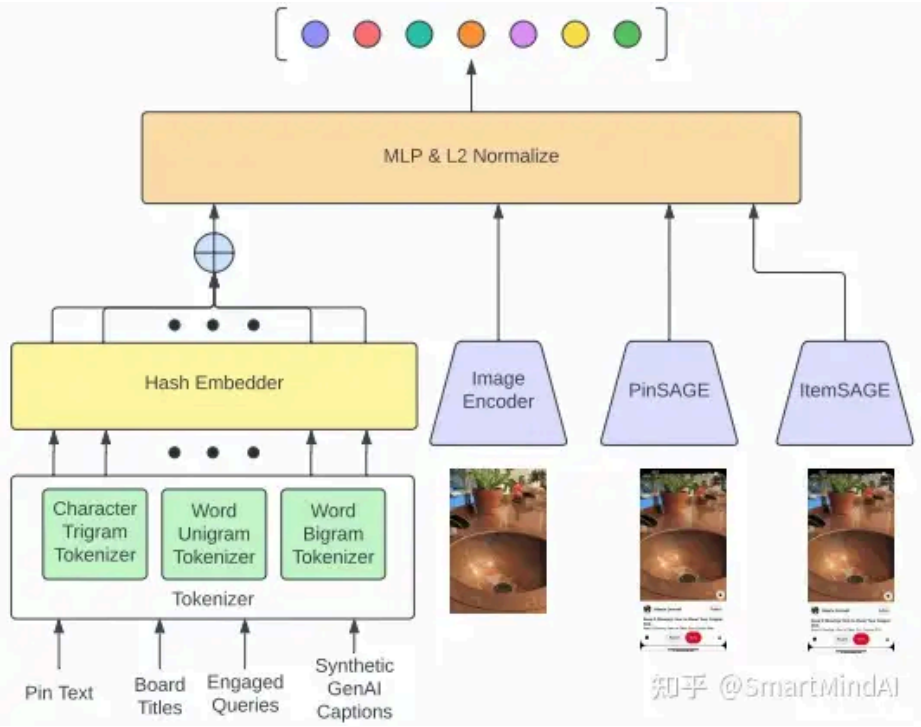
Query Encoder



在我们的模型中，查询编码器（如图 所示）基于DistilBERT的多语言版本------*distilbert-base-multilingual-cased*（底部注释1）。此库能在huggingface.co/distilbe...处获取。这个编码器设计用于理解和表示多种语言的查询，为后续处理提供了多语言支持的基础。

Unified Pin and Product Encoder

签和产品相关的实体扮演着核心角色。为确保处理的统一性，当遇到某些特定于一个实体而不存在于另一个实体的特征时，我们会使用零填充，以维持数据输入的完整性。



根据前面的深入探讨，我们采用内部批负例策略来训练模型。先前的研究明确指出，较大的批大小以及包含诸多负例有助于获得更优表示。基于这一考量，以及为了适应大批次在GPU内存中的计算限制，我们构建了一个简洁的书签编码器模型，通过多次消融实验，我们确定了如下的编码器设计。这些实验旨在为组合组件选择最适配置，同时兼顾训练效率与实用性。 为了处理书签相关的文本特性，编码器导入了三种不同的分词方法。具体为在执行令牌嵌入学习的过程中，我们利用了大小为100,000的双哈希散列嵌入表 。每个识别的令牌通过两个散列函数$h_1(i)$和$h_2(i)$在这个表的两个位置进行哈希查找。特定令牌IDi的嵌入是由这两个位置权重的插值得到的:

$$W_{1i}h_1(i) + W_{2i}h_2(i)$$

这里的W_1和W_2是针对所有可能令牌ID大小都为$|\mathcal{V}|$的学习权重向量。

所有的嵌入和特征之后会被连接起来，作为输入进入一个包含3层的多层感知器（MLP）。这3层的维度分别为1024, 1024,256。最终，经过MLP处理的输出会被进行L2标准化，以确保与查询嵌入保持一致性。

Compatibility Encoders

在本模型中，我们为书签和产品各实施了一个独立的离散兼容性编码器。这些编码器利用了现有的书签和产品嵌入，分别通过PinSage来处理书签、ItemSage来处理商品。

Multi-Task Sampled Softmax Loss

我们将书签和商品的嵌入学习视为一个分类问题，目标是识别与查询相关的实体，采用了带对数Q修正的采样softmax损失函数进行模型训练。基于多任务学习原理，同时训练实体嵌入和查询嵌入，确保它们之间兼容。 具体操作上，我们将每一任务T视为一个查询-实体对的样本集合$\mathcal{D} = \{(x, y)_i\}$与一个实体编码器$\mathcal{E}$的组合，旨在通过训练优化各个组件之间的相互作用与适应。

$$T \triangleq \{\mathcal{D}, \mathcal{E}\}$$

针对数据批$\mathcal{B}$中的每一个数据对$\{(x, y)_i\}$，对于任务T，我们的目标是学习查询嵌入q_{x_i}与实体嵌入$p_{y_i} = \mathcal{E}(y_i)$使得它们的余弦相似度最大化。这一目标是通过最小化softmax损失函数来实现的。

知乎

在这项技术中，公式涵盖了汇编计算一组数据$\mathcal{B}$与任务T之间的查询q_{x_i}和对应实体$p_{y_i} = \mathcal{E}(y_i)$的相似性，通过最大化它们的余弦相关度$q_{x_i} \cdot p_{y_i}$。这一优化目标通过最小化softmax损失实现。 为了保证问题的求解可能性并维护解的稳定性，损失函数的分母进行了规范化，这一过程利用了实体目录$\mathcal{C}$中样本的接近。除了在批次中的正向实体

$$BN = \{y_i | (x_i, y_i) \in \mathcal{B}\}$$

还采用了目录$\mathcal{C}$的随机样本$\mathcal{C}'$进行处理。 以纠正可能由于采样带来偏差的方式，引入了logQ校正技术。它通过从现有的logits中减去负向采样的概率$\log Q(y|x_i)$，实现了对样本选择偏好的修正。此步骤确保了模型不会过度惩罚在数据中较普遍的实体。

$$L_T = L_T^{S_{bn}} + L_T^{S_{rn}} \tag{2}$$

$$L_T^{S_{bn}} = -\frac{1}{|\mathcal{B}|} \sum_{i=1}^{|\mathcal{B}|} \log \frac{\exp(q_{x_i} \cdot p_{y_i} - \log Q(y_i|x_i))}{\sum_{z \in BN} \exp(q_{x_i} \cdot p_z - \log Q(z|x_i))} \tag{3}$$

$$L_T^{S_{rn}} = -\frac{1}{|\mathcal{B}|} \sum_{i=1}^{|\mathcal{B}|} \log \frac{\exp(q_{x_i} \cdot p_{y_i} - \log Q(y_i|x_i))}{\sum_{y \in C'} \exp(q_{x_i} \cdot p_y - \log Q(y|x_i))} \tag{4}$$

$$= -\frac{1}{|\mathcal{B}|} \sum_{i=1}^{|\mathcal{B}|} \log \frac{\exp(q_{x_i} \cdot p_{y_i} - \log Q(y_i|x_i))}{\sum_{y \in C'} \exp(q_{x_i} \cdot p_y - \log Q_n(y))}, \tag{5}$$

since y is sampled independently

总损失被定义为所有特定任务损失的累计和：

$$L = \sum_{T \in \mathcal{T}} L_T$$

我们将各种任务融合于一组数据中，并通过这种方式调整各个任务对模型权重的贡献。为了提升训练效率，我们会在具有相同数据集的任务之间共享批量数据中的配对示例。

Model Serving

查询嵌入在多项搜索任务中承担关键角色，强调了对延迟的严格控制以支持即时响应。为了应对实时推理需求并减缓延时，我们借助基于C++的开源机器学习模型服务器Scorpion Model Server (SMS)，结合GPU硬件加速进行高速查询编码。借鉴Zipf定律对查询分布的分析，构建了高效缓存策略，设定30天的缓存时间至生活（TTL）周期，效果显著。这种架构允系统每秒处理30万次请求，维持中位延迟不到3ms，90百分位延迟在20ms左右，成功将推理服务器负担降至每秒500次查询，实现了显著的成本和延时优化。 商品和商品嵌入通过每天的批量GPU推理生成，离线处理并存储于信号存储系统中，供后续应用调用。

Experiments

Dataset

数据集构建基于一年内的搜索查询日志，其中收集了用户行为的唯一查询-实体配对。这些行为包括"保存"书签和"长时间点击"------指用户在没有立即返回Pinterest之前，浏览链接页面超过10秒的事件。为了保证数据集的全面性，商品信息还包括了与购买过程相关的匿名化互动，如"添加到购物车"和"结账"。面对推荐系统中受欢迎程度偏差这一挑战，我们为书签和商品设置了配对次数的限制以减小影响。书签的限制为最多50次配对，商品则设为200次，以确保数据集更加均衡地代表了用户行为。

知乎

Query-Pin	Query Logs	repin, longclick	1.5B
Query-Product	Query Logs	repin, longclick	136M
Query-Product	Offsite logs	add-to-cart, checkout	2.5M
Query-Query	Query Logs	click	195M

Table 1: Summary of the different training datasets.

Offline Results

Comparison with Baselines

在本研究中，我们将当前的搜索算法SearchSage视为对比基准，其使用固定的PinSage和ItemSage进行书签与产品的信息编码。相比之下，我们的模型OmniSearchSage采用了更为先进的策略，具体体现在:

1. 查询编码器的使用:OmniSearchSage通过集成专业的查询编码器，来定制化地生成查询的嵌入，这能更精准地捕捉查询的语义特征。
2. 统一编码器的利用:与此同时，OmniSearchSage同样配备了统一的书签和产品编码器，用于生成这些实体的嵌入，确保在处理多类对象时的一致性和高效性。 这样的设计旨在通过优化不同环节的嵌入生成过程，最终提升整体的搜索性能和用户体验。

Metric	SearchSage	OmniSearchSage	Gain
	Pin		
Save	0.39	0.65	+67%
Long-Click	0.45	0.73	+62%
Relevance (US)	0.25	0.45	+80%
Relevance (UK)	0.29	0.51	+76%
Relevance (FR)	0.23	0.43	+87%
Relevance (DE)	0.28	0.46	+64%
	Product		
Save	0.57	0.73	+28%
Long-Click	0.58	0.73	+26%
	Query		
Click	0.54	0.78	+44%

对于书签数据集，OmniSearchSage在所有衡量指标上实现了显著提升，范围从60%到90%。这一结果展示在不同国家的稳定召回率，体现了OmniSearchSage在多语言环境下的优秀性能与适应性。

在产品数据集上，OmniSearchSage的参与度预测优于SearchSage约27%，在书签数据集的基础上观察到的改进。由于原始模型基于搜索任务训练，此结果则倾向于表明，加入新特性和多任务学习策略的路径与方法对性能的正面影响。

尽管SearchSage在相关查询点击任务上表现良好，优于随机预测，然而，OmniSearchSage通过直接对此目标进行训练，实现了高达$+44\%$的显著性能提升。这一结果将在后续部分解析为直接针对相关查询优化与多任务学习策略综合效应的作用力。

Importance of content enrichment

我们首先对处理缺少标题和描述的书签的合成标题技术进行了探索。为此，我们在评估数据集中选择了包含缺失标题或描述的书签配对，构造了一个缩小后的评估数据集，共计24,000对配对。我

26%的显著改进。与此同时，相关性的评估指标保持相对稳定。这一结果指明，针对缺少标题和描述的书签使用合成标题方法，能够有效提升模型在特定评估指标上的表现。

	save	long-click	relevance
No captions	0.51	0.60	0.36
With captions	0.66	0.76	0.36
Improvement	+30.43%	+25.58%	0%

表呈现了额外文本增强对模型性能提升的影响。每次增加新的文本特性，模型性能都会以百分比方式相应提升。基准模型，构建成仅使用连续特征，其性能为第一行所示。

	save	long-click	relevance
Continuous Features Only	0.43	0.53	0.30
Adding Title, Description and Synthetic GenAI Captions	0.52 (+21%)	0.63 (+19%)	0.39 (+30%)
Adding Board Titles	0.61 (+17%)	0.68 (+8%)	0.44 (+13%)
Adding Engaged Queries	0.65 (+7%)	0.73 (+7%)	0.46 (+5%)

在基准模型基础上依次加入「标题」、「描述」和「合成生成AI标题」后，所有评价指标均出现稳定的提升。对于「参与数据集」而言，提升幅度达到总值的20%，而与模型相关性的提升则更为显著，达到了总值的30%，彰显了文本特性的重要意义。 将「版本标题」纳入特征列表后，模型性能继续攀升，不同指标增幅在8%到15%区间内。这进一步验证了版本标题在提高预测准确度方面的效用。 最终，引入「参与查询」特性大幅改善了模型的整体表现，在与初始状态相比时，尽管增长幅度较小，但依然实现了显著提升。 综上所述，每项文本增强特性均能带来显著的性能增强，这种效果通过比较与前一状态的指标变动得以直观展现。

Effect of multi-tasking

Dataset		Pin Only	Product only	Query Only	OmniSearchSage
pin	save	0.68	-	-	0.65
	long-click	0.75	-	-	0.73
	avg relevance	0.45	-	-	0.46
product	save	-	0.73	-	0.73
	long-click	-	0.73	-	0.73
query	click	-	-	0.73	0.78

在表中，我们列出了专为各自任务（书签、产品、查询）训练的模型与采用合并多任务策略训练的模型之间的性能对比。所有模型均在相同的条件下训练，包括使用相应的批次大小、计算资源和迭代次数。并且，它们都基于相同的训练和评估数据集，区别仅在于每个模型依据数据集内的部分子集进行训练，以确保公平、准确的性能评估。 对于书签任务，多任务学习策略呈现出微小的性能下降。然而，在处理产品和查询任务时，多任务模型不仅保持了中性表现，甚至在某些情况下表现出了积极的或显著的性能提升。

Effect of compatibility encoders

我们探索了引入兼容性编码器对书签和产品嵌入学习效果的增益。通过比较一个仅包含查询与统一书签/产品编码器模型，与一个整合所有编码器的模型，我们发现添加兼容性编码器对于学习到的嵌入影响甚微，甚至未明显降低指标，实现了查询嵌入与已有嵌入的平滑兼容，同时在成本方面影响不大。

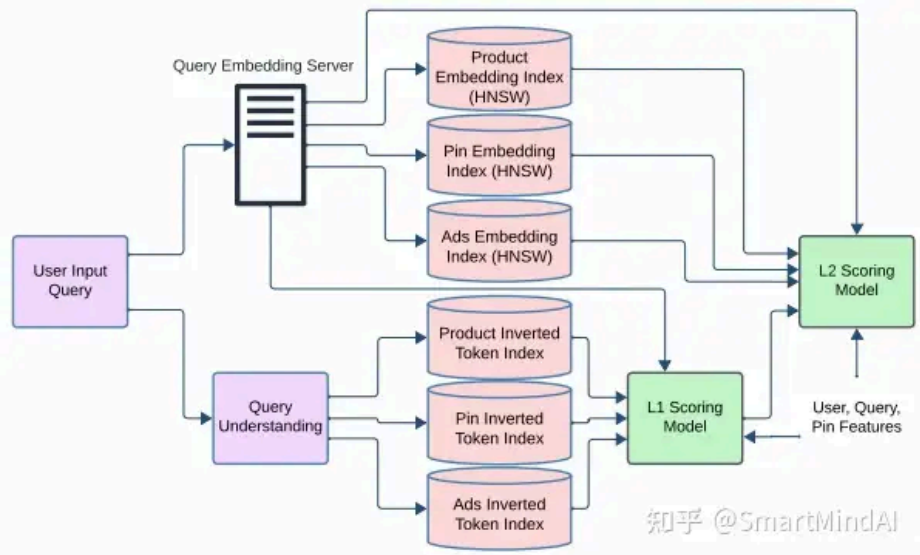
知乎

pin	save	0.39	0.39
	long-click	0.45	0.43
	avg relevance	0.26	0.26
product	save	0.57	0.57
	long-click	0.58	0.57

进一步的分析显示，在**OmniSearchSage**模型中应用兼容性编码器，其性能要么与仅使用兼容性编码器训练的SearchSage模型相当，要么甚至超越了该模型的性能。

Applications in Pinterest Search

OmniSearchSage嵌入技术在Pinterest的搜索系统中应用广泛，特别是在检索和排序相关任务中发挥关键作用。图清晰地展示了Pinterest搜索流程的简化图示，并标识了**OmniSearchSage**嵌入集成的关键位置。

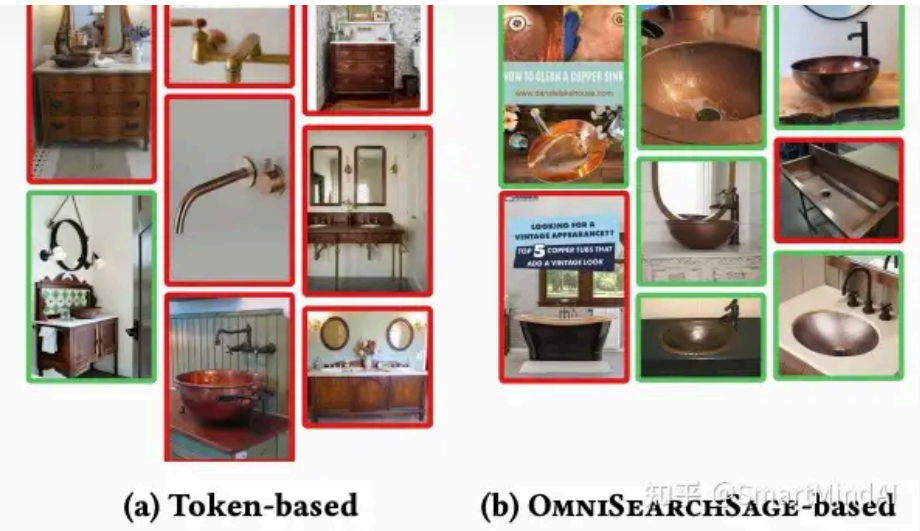


OmniSearchSage嵌入在Pinterest搜索架构的检索和排序阶段应用广泛，通过与HNSW 算法结合进行书签和产品的高效检索，进一步优化了基于令牌的检索系统[+]的效能，使得L1得分模型更加高效运行。在L2得分和相关性模型中，**OmniSearchSage**嵌入是构成这些系统的最关键特征元素之一。 通过A/B测试获得的发现，结果显示将**OmniSearchSage**嵌入替换生产环境中原有的SearchSage嵌入后，显著提升了有机内容和推广内容（广告）在搜索结果中的表现。此外，还通过实际生产样本流量，进行了基于人类评价的关联性评估，证实使用**OmniSearchSage**嵌入能带来明显的性能提升。

Human Relevance Evaluation

为了评估模型[+]**OmniSearchSage**的效能，我们安排了人类评估员对基于**OmniSearchSage**嵌入与基于令牌的检索两种方式进行了对比研究。具体应用了包含主要查询和长尾查询两类的300组查询数据，每组查询的前8名候选书签由系统生成，随后通过人类评估员对每个书签与对应查询的关联程度进行打分评估。其中，数据集内的评分为一致性高，协议系数（Krippendorff's alpha）达到了0.89，这份评分方案的平均提升为**OmniSearchSage**嵌入方式在相关性上的优势，相较于基于令牌的系统，平均高出了10%。

知乎



通过视觉化对比（图，如"古董铜质浴室水槽"（English: antique copper bathroom sink）这一查询为例，基于令牌检索系统往往只检索到部分相关结果，且难以保障一致性。反观基于**OmniSearchSage**的系统，几乎每项检索结果都与指定查询高度吻合，明晰地展示了**OmniSearchSage**模型在查询理解、以及整合相似书籍和查询空间上的独特能力。这种特性使得**OmniSearchSage**成为了一种在相关性评估中表现优异的模型选择，在提高用户搜索体验上展现出了显著价值。

原文《OmniSearchSage: Multi-Task Multi-Entity Embeddings for Pinterest Search》

编辑于 2024-06-25 11:36 · IP 属地北京

Pinterest 搜索引擎 多模态