arXiv:2603.04741v1 [cs.AI] 5 Mar 2026

CONE：用于保留单位和变量语义的复杂数值数据嵌入

Gyanendra Shrestha
佛罗里达州立大学

Anna Pyayt 南佛罗里达大学

Michael Gubanov
佛罗里达州立大学

摘要

大型预训练模型（LM）和大语言模型（LLM）通常擅长捕捉语言语义和上下文关系。然而，在涉及数字的任务中，这些模型难以保持最佳性能。将数值或结构化数据一概视为词项并不够；模型必须充分理解并编码其语义。本文提出 CONE，这是一种混合式 Transformer 编码器预训练模型，能够将数字、范围和高斯分布编码到保留距离信息的嵌入向量空间中。我们提出了一种新颖的复合嵌入构建算法，将数值、范围或高斯分布与其对应的单位和属性名称相结合，从而精确捕捉其复杂语义。我们在涵盖多个领域（网络、医疗、金融和政府）的多个大规模数据集上进行了广泛的实验评估，结果证明 CONE 具备强大的数值推理能力：其在 DROP 上的 F1 分数达到 87.28%，F1 分数较最先进的基线模型最高提升 9.37%，并且在 Recall@10 上较主要的最先进模型最高提升 25%。

CCS 概念

• **信息系统** →**数据管理**。

关键词

嵌入、语言模型、Top-k 检索

ACM 参考格式：

Gyanendra Shrestha、Anna Pyayt 和 Michael Gubanov。2026年。CONE：保留单位和变量语义的复杂数值数据嵌入。载于《*2026*年数据管理国际会议论文集（*ACM SIGMOD '26*）》。ACM，纽约，NY，美国，15 页。https://doi.org/XXXXXXX.XXXXXXX

1 引言

自然语言文本通常同时包含数值和词语。数值数据在科学、技术、工程和数学（STEM）领域尤为普遍且重要，在这些领域，精确的数值表示至关重要。数值可以用数字表示（例如“15”），也可以用字母形式表示（例如“fifteen”），还可以附带单位（例如“15 英里”）。单位

允许免费制作本作品全部或部分的数字或纸质副本，供个人或课堂使用，前提是这些副本不得为营利或商业利益而制作或分发，且副本须在首页载明本声明及完整引文。对于本作品中由作者以外的他人拥有的组成部分，必须尊重其版权。允许在注明出处的情况下进行摘要。以其他方式复制、再版、在服务器上发布或向列表再分发，均须事先获得特别许可和/或支付费用。请通过 permissions@acm.org 申请许可。*ACM SIGMOD '26,* 印度班加罗尔 © 2026 年 版权由所有者/作者持有。出版权已授权给 ACM。ACM ISBN 978-1-4503-XXXX-X/2026/04 https://doi.org/XXXXXXX.XXXXXXX

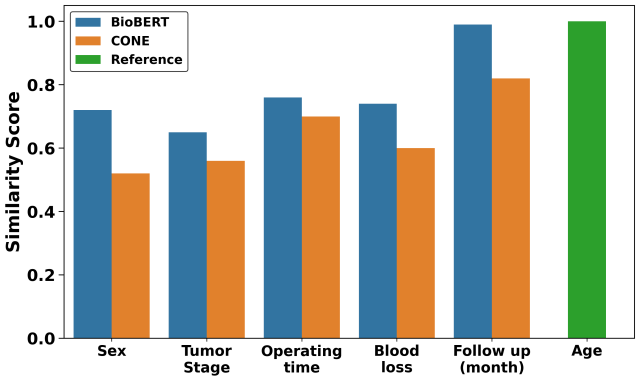


图1：年龄 与其他属性之间的距离，使用 BioBERT 和 CONE 计算。

在正确理解和区分数量方面发挥着重要作用（例如，10 英里与 10 千克）。结构化数据中的数值还承载着额外的语义，例如作为特定属性的值，并且也可能附带单位。尽管近期语言模型在自然语言理解方面表现出色 [14],，但它们在处理涉及数值和结构化数据的任务时仍然面临困难。多项研究表明，预训练语言模型中的数值嵌入无法正确捕捉数值的语义 [14, 18, 34, 47, 51, 56, 67, 71, 75, 80]。造成这一局限的原因之一在于这些模型采用的分词策略：在生成嵌入向量时，它们不加区分地对待数字和词语。例如，BERT [14] 采用子词分词方法 [61],，会像拆分词语一样将数字拆成多个部分。以数字 28,600 为例：它可能被分词为“28”和“–600”，从而完全扭曲原本的语义。还需注意，不同的分词器可能产生不同的切分结果。最后，所有这些模型都忽视了测量理论 [66]所述的词语与数字之间的根本差异。词语通常是名义的（用于标记，不具有内在顺序或大小），而数字则遵循区间或比率 尺度，在这些尺度中，距离、顺序和比例关系至关重要。常规词嵌入能够捕捉词语之间的语义相似度，但缺乏正确编码数值数据的机制，也无法保留数值的基本属性。与一维文本不同，结构化数据具有二维上下文，其中数字与特定属性相关联，因此这一问题更加突出。例如，图 2 中的两列年龄 和随访（月） 包含相似的数值分布，但语义不同，这种差异应体现在相应的向量化表示中。预训练语言模型往往难以区分这类属性（图1），导致其在许多数值推理任务中表现不佳。

近年来，数值嵌入开始受到更多关注 [18, 56, 67, 75, 80]。表 1 将 CONE 与最先进基线进行了比较。

表 1：CONE 与最先进模型在关键数值推理能力及对复杂数值数据的支持方面的比较。

特征	BERT	ELMo	NumBERT	BioBERT	DICE	AeNER	GenBERT	NumNet	CONE
数词表达	有限	有限	yes	有限	yes	yes	yes	yes	yes
大小	yes	yes	yes	yes	yes	yes	yes	yes	yes
列表最大值	有限	优于 BERT	-	有限	yes	yes	yes	yes	yes
解码	有限	优于 BERT	-	有限	yes	yes	yes	yes	yes
加法	有限	有限	-	有限	yes	yes	yes	yes	yes
标量探测	some	有限	good	有限	-	-	-	-	yes
Text	yes	yes	yes	yes	是（以文本形式的数字）	yes	yes	yes	yes
表格数值数据	no	no	no	no	no	yes	no	no	yes

比较结果基于原始论文和先前文献中报告的结果。这些方法都无法充分捕捉具有单位、范围和高斯分布的数值数据的语义。这一根本局限阻碍了它们全面理解并正确编码文本和结构化数据中的复杂数值数据，因为这些数据的语义取决于相关属性和单位（例如，年龄：*50* 年 与 体重：*50* 千克）。

为解决这些局限性，我们介绍了 CONE，这是一种用于构建复合嵌入的新模型，能够正确编码并涵盖复杂数值数据的语义，包括带单位的范围和高斯分布。CONE 构建于 BERT [14] 和 BioBERT [34] 等预训练语言模型之上，将它们处理复杂数值语义的能力扩展到更广范围。通过在复杂的结构上下文中编码数值，复合 嵌入方法确保单位和/或属性/变量不同的数字不会被盲目地视为相同，例如*5* 千米 和*5* 千克，或*5* 毫克：剂量和 *5* 毫克：体重，尽管它们的数值相同（ 5 ）。此外，CONE 还能正确理解并编码数值范围（例如， *[5-10* 年*]*：年龄）和高斯分布（例如， *[1302±0.25* 纳米，均值 ± 标准差*]*：粒径）的语义，使模型不会丢失其真实含义。通过在复杂结构化数据的热门下游任务上开展大量实验，我们证明，CONE 的表现显著优于广泛使用的最先进基线，例如 TAPAS [25], NumNet [56], NC-BERT [31], Magneto [44] 以及通用嵌入模型 [11, 26, 79, 81]。通过融入感知单位的嵌入和属性特定嵌入，我们的模型确保复杂数值数据得到正确表示，从而提升检索和预测类下游任务的表现。

我们的主要贡献如下：

- 一种新颖的复合嵌入结构，用于处理复杂数值数据，纳入数值、单位和属性，使模型能够准确编码并理解具有复杂关联上下文的数字。这确保具有不同单位和属性的数字彼此区分，且编码能够保留其基本数值属性。
- 用于数值范围和高斯分布的特殊嵌入，同时保留其语义。
- 用于预计算嵌入向量分量（算法1）和完整嵌入向量组合（算法 2）的两种新算法。
- 在来自不同领域的大规模数据集上对 CONE 进行了广泛的实验评估，展示了其在热门下游任务上的卓越性能。

本文其余部分的结构如下：首先介绍预备知识，然后阐述我们的方法论，接着进行严格的嵌入距离分析和全面的实验评估，最后介绍相关工作并作出总结。

Age	Sex	肿瘤分期	手术 time	血液 loss	随访（月）	BP（毫米汞柱）	BMI（千克/m²）
28	F	S1–3	7 小时	3,000 毫升	30	76-118	21.8 ± 2.9
34	F	S1–3	580 分钟	5 L	32	80-122	23.4 ± 3.2
36	F	S1–3	6 小时	3.5 升	28	82-125	22.5 ± 3.17
42	M	S1–2	500 分钟	3,000 毫升	25	85-130	24.1 ± 3.5
33	F	S1–3	400 分钟	3.2 升	20	78-120	22.2 ± 3.0
31	M	S1–2	500 分钟	3,000 毫升	15	84-128	23.6 ± 3.4

图 2：包含数值数据、范围和高斯分布并附带单位的表。

2 预备知识

虽然语言模型在捕捉文本语义方面表现出色，但在捕捉数值或复杂结构化数据的语义方面仍有局限，因为数字的语义取决于其属性 （例如，年龄、剂量、身体质量指数）、表示这些数值的单位 （年、毫克、千克/米²），以及数值本身的结构（单个值、范围或高斯分布）。数值“30”可能表示*30* 岁、随访 *30* 个月或肿瘤大小为 *30* 毫米；其含义完全取决于相关属性和单位。标准语言模型（例如 BioBERT）无法正确编码这些区别，因此可能会混淆数值相同或词汇相似、但语义不同的属性 。

设表为 $T = (C, R)$，其中 $C = \{c_1、\ldots\ldots、c_m\}$ 是列的集合，其属性名称为 $A = \{a_1、\ldots\ldots、a_m\}$，而 $R = \{r_1、\ldots\ldots、r_n\}$ 是元组的集合，其中包含相应的单元格值。

定义 1（列相似度）。当两列 c_1 和 c_2 具有相似的属性值 [23] 时，它们相似。下面将更精确地描述列相似度度量。

每列 c_j 包含属性值 $\{v_{1j}、\ldots\ldots、v_{nj}\}$。对于每个 v_{ij}，我们计算复合嵌入 $E^{\text{comp}}(a_j、v_{ij}、u)$（算法 2），对属性名称 a_j、v_{ij} 和单位 u进行编码。随后通过聚合得到列 c_j 的嵌入向量：

$$E(c_j) = \sum_{i=1}^{n} E^{\text{comp}}(a_j, v_{ij}, u) \tag{1}$$

通过计算两个列 c_1 和 c_2 聚合后的嵌入向量 [23]之间夹角的余弦，来计算它们的相似度：

$$Sim_{col}(c_1, c_2) = \cos\left(E(c_1), E(c_2)\right) \tag{2}$$

定义 2（元组相似度）。当两个元组 r_1 和 r_2 的单元格值相似时，它们即为相似元组。下文将更准确地描述元组相似度度量。

每个元组 $r_i = [v_{i1}, \ldots, v_{im}]$ 由与每个属性 $a_j \in A$对应的单元格值 v_{ij} 组成。元组 r_i的嵌入通过聚合获得：

$$E(r_i) = \sum_{j=1}^{m} E^{\text{comp}}(a_j, v_{ij}, u) \tag{3}$$

两个元组 r_1 和 r_2 之间的相似度通过计算其聚合嵌入向量之间夹角的余弦值来确定：

$$Sim_{tuple}(r_1, r_2) = \cos\left(E(r_1), E(r_2)\right) \tag{4}$$

示例 1。 请看图 *2* 中的一张表。“年龄（岁）”可以表示为 *{28, 34, 36, 42, 33, 31}*、*{29, 31, 35, 42, 32, 32}*，也可以表示为诸如*“30–45”*的范围。尽管表示方式不同，它们都属于同一人口统计学概念，因此在向量空间中的嵌入位置应彼此接近。相比之下，“随访时长（月）”等属性的值为 *{20, 25, 30, 36, 40, 32}*，或“肿瘤大小（毫米）”的值为 *{30, 35, 42, 22, 26, 32}*，虽然数值可能与“年龄”重叠，但编码的是完全不同的变量。因此，它们在嵌入向量空间中的位置不应接近“年龄”。在图*1*中，只有可能包含某些“年龄”值的其他属性，才应与“年龄”表现出较高的相似度。“随访”或“手术时间”等属性必须保持足够远离，无论它们因语义不同而与“年龄”存在何种数值重叠。

我们可以从图1中看到，BioBERT 无法区分图 2 中表格里的年龄 和随访（月）。它们的余弦相似度得分为 0.9998，因此角距离非常小—— 1.02°，这意味着由 BioBERT 构建的这两个向量几乎完全重叠。这是因为这两列中的数值分布非常相似，而且原始 BERT 嵌入向量对单位和属性差异视而不见（即缺乏数值与单位语义）。BioBERT 的高相似度得分主要由属性名称 之间的文本相似性驱动（例如*month* 在语义上与年龄相似），而非数值所蕴含的更复杂语义。不仅 BERT 和 BioBERT，其他 SOTA通用嵌入模型 [11, 26, 79, 81] 在涉及数字的任务上也表现不佳，这同样是因为它们将数字视为普通词项。

在大规模数据集中，属性异质性和数据表示多样性会加剧这一问题。这些表格来自各种不同的来源，采用不同的命名约定、模式和格式。此外，它们在表头中使用不同的同义词和缩写，这使问题更加复杂。即使是相同的属性，其文本和数值数据格式也会有所不同，这给构建能够准确模拟数据语义的嵌入带来了额外挑战。为克服这些挑战，我们提出一种更复杂的嵌入向量结构——复合结构，并对编码器架构进行修改，以准确编码复杂数值数据的语义。

定义 3（序列化）。在列级，我们将每一列 c_j col c_j [ ] a_j [ ] v_{1j} [ ] $\ldots$ v_{nj} [ ] 独立序列化为：S ( ) :- CLS SEP SEP SEP 。即，先放置列名称，随后放置所有单元格值，并使用 [SEP] 词元分隔各个值，在开头放置一个 [CLS]词元（其中 [CLS] 和 [SEP] 遵循 BERT 词元化标准 [14]）。在元组级，我们将一行 r_i 序列化为 $S_{\text{row}}(r_i)$:- [CLS] a_1 v_{i1} [SEP] $\ldots$ a_m v_{im} [SEP]。即，每个属性名称都与其对应的单元格值交替排列，属性—值对之间用 [SEP] 词元分隔。

我们尝试了多种其他序列化策略，但效果都不如预期。为保持一致性，在我们的实验中，BioBERT 和其他模型均采用相同的序列化方式。通篇论文中，我们使用 属性 来指代列标题（即列名），而 列 则表示整列，包括标题及其单元格值。

示例 2。 考虑图 *2* 所示表中的元组和列。我们将表中具有“年龄”属性的列序列化为：*[ ] [ ] 28 [ ] 34 [ ] 36 [ ] 42 [ ] 33 [ ] 31 CLS* 年龄 *SEP SEP SEP [ ] SEP SEP SEP SEP*。类似地，我们将表中的元组（第一行）序列化为：*[CLS]* 年龄 *28 [SEP]* 性别 *F [SEP]* 肿瘤分期*S1-3 [SEP]* 手术时间 *7 [ ] 3000 [ ] 30* 小时 *SEP* 失血量 毫升 *[ ] 76 118 [ ] 2 21 8 ± 2 9 [ ] SEP* 随访（月） *SEP* 血压（毫米汞柱）- *SEP* 身体质量指数（千克/米 ）。。*SEP*。

我们修改了 BERT 架构 [14, 34] ，以适应包含单位、数字、范围和高斯分布的复杂数值数据的高级语义。每个 Transformer 层都包含一个多头自注意力子层，其中每个词元都会关注所有词元。设层输入 $H = [h_1, h_2, \ldots, h_n]^T \in \mathbb{R}^{n \times d}$ 对应于序列 S，其中 d 为隐藏维度， $h_i \in \mathbb{R}^{d \times 1}$ 为位置 i处的隐藏表示。对于单头自注意力子层，输入 H通过三个矩阵 $W^Q \in \mathbb{R}^{d \times d_K}$、$W^K \in \mathbb{R}^{d \times d_K}$ 和 $W^V \in \mathbb{R}^{d \times d_V}$投影为对应的表示 Q、 K 和 V：

$$Q = HW^Q, \quad V = HW^V, \quad K = HW^K \tag{5}$$

然后，该单头自注意力子层的输出计算如下：

$$\text{Attn}(H) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_K}}\right)V \tag{6}$$

如图1所示，我们的方法将年龄和随访之间的相似度降至 0.82。在对与年龄相似的列进行排序时，我们的方法将所有与年龄相关的列排在最前面，而随访则不会出现在候选项前列。我们的方法专门通过联合捕捉属性、单位和数值来编码这类差异，确保语义等价的数值属性在嵌入空间中彼此接近，而语义不同的属性则保持明显区分。类似地，当属性名称相同但单位不兼容时，也会产生歧义。例如，属性剂量可能指 40–60 毫克（药物质量）、4–6 毫升（药物体积）或 1.0–1.5 国际单位/千克（生物效价）。基于文本的嵌入会将这些内容视为几乎相同，因为术语“剂量”主导了其嵌入表示，且未考虑单位不匹配的问题。我们的方法明确纳入单位语义，确保剂量（毫克）、剂量（毫升）和剂量（国际单位/千克）被嵌入为不同的概念。我们的嵌入表示具有通用性，无需针对不同的下游任务进行任何修改。

3 CONE模型

我们的模型架构如图3所示。它通过数值感知嵌入增强标准Transformer 编码器，从而提升模型理解和推理数值的能力。

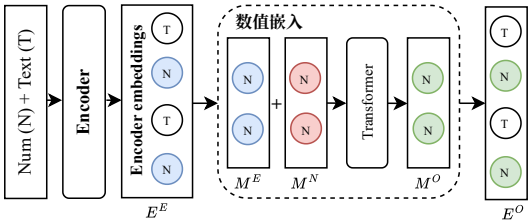


图3：CONE架构，包括编码器和数值嵌入。数字的编码器嵌入（M^E，以蓝色显示）与数值嵌入（M^N，粉色）融合，并通过 Transformer 进行细化，以生成上下文化表示（M^O，绿色）。

3.1 编码器

我们采用基于Transformer的语言模型作为编码器 [14,75], ，以BioBERT [34] 权重初始化，并在我们的数据集上微调。在本工作中，我们关注表格中的数值列/元组。编码器的输入是[CLS], 与表的元组或列经分词后的拼接，其中表格沿行/列展平，各单元格通过序列化函数S（见定义3）以 [SEP] 词元分隔。给定长度为 l的分词后的输入序列，编码器生成相应的上下文化表示，记为：

$$E^E = [e_1, e_2, \ldots, e_l] \in \mathbb{R}^{l \times d} \tag{7}$$

其中 d 表示嵌入维度（我们始终使用 $d = 768$ ）。

我们不修改底层编码器针对文本词元的默认分词器，因此它们遵循标准的分词过程 [34]。然而，我们覆盖分词器对数值数据的行为（见第3.2节）：每个数字都被视为单个词元，而不是像在BERT/BioBERT中那样，在不存在对应嵌入时被拆分为多个子词标记。每个数值标记的嵌入通过我们在图3中的自定义架构学习得到。

3.2 数值嵌入

我们提出了数值融合 嵌入，能够联合捕捉符号上下文和定量大小。具体而言，我们通过为数字的标准标记嵌入增添数值嵌入 [6, 21, 29, 65, 67, 68] ，并通过注意力机制 [75]将二者整合起来，从而实现这一目标。尽管本文聚焦于基于编码器的模型，但CONE框架并不特定于某种架构。数值注意力融合机制可以通过将编码器和解码器架构中的标记嵌入替换为融合表示，并在其后加入用于数值特定推理的轻量级Transformer块，从而应用于这两类架构。

我们从输入（例如列或行）中提取数值词元（即数字），并使用DICE [67] 生成数值嵌入 M^N，这是我们嵌入向量的组成部分之一，可增强数值属性（例如大小）的表示。也可以使用其他现有方法 [6, 21, 29, 65, 68]。

设 $I^{\text{num}} = \{i_1, i_2, \ldots\ldots, i_s\}$ 为数值词元的索引集合。每个索引对应输入序列中一个数字词元的相对位置。对于每个 $i_j \in I$，我们从 E^E 中提取相应的编码器嵌入，并将这些嵌入的子集 $M^E \in \mathbb{R}^{s \times d}$ 表示如下：

$$M^E_j = e_{i_j}, \quad \text{for } j = 1, \ldots, s \tag{8}$$

我们通过将上下文嵌入 M^E 与数值嵌入 M^N逐元素求和，计算得到一个融合表示，并将结果输入一个 Transformer 块，以实现数值特定推理。随后，使用所得输出 M^O（其为每个数字词元提供上下文化嵌入）更新编码器输出 E^E中的相应嵌入。由此得到最终嵌入 E^O，其中融合了上下文语义和数值语义。

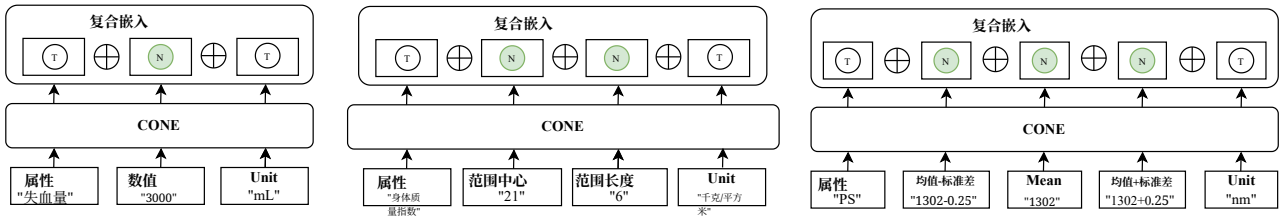
算法1： 通过掩码数字预测训练 CONE
tion

输入： 分词后的输入 $x = [x_1, \ldots, x_l]$；编码器 f_θ（已初始化与 BioBERT）；数值嵌入 NumEmbed(·)；Transformer 块 g_ϕ；掩码率 p；Adam 优化器 O
输出： 训练后的参数 θ 和 ϕ
$I^{\text{num}} \leftarrow \{i \in [1..l] \mid x_i$ 是一个数字$\}$ // 数字的索引词元
$\tilde{x} \leftarrow$ 数字掩码(x, I^{num}, p) // 用 [MASK] 替换子集
$E^E \leftarrow f_\theta(\tilde{x}) \in \mathbb{R}^{l \times d}$ // 上下文嵌入
$M^E \leftarrow [E^E_i]_{i \in I^{\text{num}}} \in \mathbb{R}^{s \times d}$ // $s = |I^{\text{num}}|$
$M^N \leftarrow$ [数字嵌入$(x_i)]_{i \in I^{\text{num}}} \in \mathbb{R}^{s \times d}$
$M^F \leftarrow M^E + M^N$ // 通过逐元素求和进行融合
$M^O \leftarrow g_\phi(M^F)$ // 数值特定推理
$(\hat{y}, \hat{c}) \leftarrow$ 预测头(M^O) // 大小 $\hat{y}$ 和类别 $\hat{c}$
$(y, c) \leftarrow$ 真实量级与类别(x, I^{num})
$\mathrm{L}_{\text{mag}} \leftarrow \sum_{j=1}^{s} (\log y_j - \log \hat{y}_j)^2$
$\mathrm{L}_{\text{cls}} \leftarrow \sum_{j=1}^{s} \mathrm{CEL}(c_j, \hat{c}_j)$
$\mathrm{L}_{\text{num}} \leftarrow \mathrm{L}_{\text{mag}} + \mathrm{L}_{\text{cls}}$
$\theta, \phi \leftarrow \mathrm{O}(\theta, \phi, \nabla_{\theta, \phi} \mathrm{L}_{\text{num}})$
返回更新后的参数 θ 和 ϕ

我们使用掩码数字预测任务 [39,58, 64], 训练模型，如算法1所述，模型需要预测输入中的一个被掩码数字。目标函数 L_{num}（公式 9）结合了幅值回归和分类损失：

$$\mathcal{L}_{\text{num}} = \sum_{i=1}^{N} \left\{ (\log(y_i) - \log(\hat{y}_i))^2 + \mathrm{CEL}_i \right\} \tag{9}$$

其中， y_i 表示真实数值大小，$\hat{y}_i$ 是模型预测的数字的大小，N 是训练过程中被遮蔽的数字数量，而 CEL_i 是第 i个数字词元的交叉熵损失。训练完成后，数字预测头会被移除，最终的上下文化嵌入



(a) (b) (c)

图4：复合嵌入 （a）复杂数值数据、（b）数值范围和（c）高斯分布的结构。T：文本，N：数值 ，标准差：标准差偏差。

E^O用于各种下游任务。E^O既编码数字的语义角色（上下文），也编码其在数轴上的定量值（大小）。

3.3 复合嵌入结构

先前研究 [9, 41, 42, 47, 72, 74, 77] 表明，拼接后的嵌入在相似度和分类等任务上的表现优于单个嵌入。重要的是，拼接（CONC）会将每个嵌入保留在其原始子空间中，从而保留源信息，而不会强行将其映射到单一空间。因此，所得表示既具有表达能力（通过整合互补信号），又具有区分性（例如，年龄 *= 50* 年 与 体重 *= 50* 千克，相同的值 “50” 会因属性和单位上下文不同而有所区别）。受此启发，我们提出一种复合嵌入结构，将表示属性、其值（标量、范围或高斯分布）和单位的嵌入进行拼接（⊕） E^O （见图3），从而保留各组成部分对结构化数值数据的互补贡献。我们的复合嵌入对应于子空间组合 [22],，确保每个组成部分（属性、值、单位）都能独立地影响整体距离（详细的距离分析见第5节）。例如，当属性和单位固定时，复合嵌入之间的距离会简化为值之间的差异，因此向量距离能够自然地反映数值接近程度；当属性或单位固定时，同样的性质也成立。

图4(a)展示了图 2 中表的一列失血量 的这一过程：属性失血量 在第一行的值为“*3,000*”，单位“毫升”与该值一并给出。这种复合嵌入结构保留了数值的实际语义。我们将范围 （*a-b*）称为一个区间，它包含两个边界 a 与 b 之间的所有x，使得 $a \le x \le b$。为了为数值范围 值（例如，年龄*1–5*年）构建复合嵌入，我们用其中心 $\frac{a+b}{2}$ 和长度$|b-a|$来表示每个范围，二者正交地表示该范围的位置与尺度。随后，通过拼接属性、中心、长度和单位的嵌入，得到复合嵌入。在图4(b)中，我们展示了身体质量指数 （即身体质量指数）“*18-24*”千克/m^2的这一复合结构。为简洁起见，我们将高斯分布 称为高斯分布 $mean \pm SD$，其中 *mean* 和 SD，SD 表示标准差。针对高斯分布 值的复合嵌入与范围的结构类似，我们将属性、“$mean - SD$”、“*mean*”、“$mean + SD$”和单位的嵌入进行拼接。图4(c)以示例属性*PS*（纳米） （粒径）及其值“1302 ± 0.25”说明了这一结构。

CONC 嵌入的维度与分量数量成正比（即，对于 k 个分量，维度为 $k \cdot d$）。为避免 CONC 中维度过高且不固定，并获得固定大小的表示，我们采用带有零填充和掩码的基于槽位的拼接方案。我们预留一组固定的槽位 S（在我们的情形中为|S | = 5），每个槽位大小为 d。缺失的分量用零填充，并将相应的二进制掩码位设置为 $m_s = 0$。拼接所有槽位得到 $E \in \mathbb{R}^{|S|d}$，随后使用线性自编码器 [4, 5]将其投影到维度为 $d_C = 768$ 的向量空间。重构结果为$\hat{} E = W^\top W E$，其中 $W \in \mathbb{R}^{d_C \times (|S|d)}$ 表示编码器投影矩阵， $W^\top$ 表示绑定解码器，训练则最小化掩码重构损失 [24, 70]。

$$\mathcal{L}_{\text{embed}} = \|M \odot (\hat{E} - E)\|_2^2 \tag{10}$$

其中 $M \in \{0,1\}^{|S|d}$是由 $\{m_s\}_{s \in S}$ 导出的广播掩码， $\odot$表示哈达玛（逐元素）积。编码表示 $z = WE$ 经过层归一化，得到最终的复合嵌入：

$$E^{\text{comp}} = \text{LayerNorm}(z) \tag{11}$$

在我们的实现中，属性名称 a、数值 v 和单位符号 u 通过改编自 [10] 的基于规则的正则表达式解析器，从表格数据的每一行或每一列中提取。 [10] 中的方法还支持单位规范化，即将所有单位变体归一化为一致的规范形式。所得三元组 (a, v, u) 随后用于按照算法 2 所述构建复合嵌入。

4 数据集

我们在实验中使用以下五个公开可用的大规模数据集，并采用 [30] 中的方法进行复杂元数据识别，其中包括检测多行、分层表头和双维表头（即带有元数据的列和行）。这有助于为我们的实验构建大规模训练和测试集（第 6.3 节）。

CancerKG [62] 数据集包含 350 万个表格，这些表格取自 PubMed.com 上有关不同癌症的所有最新医学出版物（截至 2024 年 12 月）。这些表格总计有 1,975 万列和 1,844 万个元组。列值和元组值包含字符串、带单位或不带单位的数字以及范围。**CovidKG** [62] 取自 PubMed.com 上有关 COVID-19 的医学出版物（截至 2022 年）。其中包含 82 万篇论文和 140 万个表格。**Webta- bles** [37] 包含从维基百科页面提取的约 165 万个表格。该语料库涵盖大量有关各种主题的事实性知识，范围从体育赛事（例如奥运会）到艺术

算法 2： 复合嵌入构造

输入： 属性 a，数值 v（标量、范围 $[a,b]$ 或高斯分布（μ，σ）），单位 u，槽位大小 d，槽位总数 $|S|$，投影矩阵 W
输出： 复合嵌入 E^{comp}
如果 a 存在**则** $E_a \leftarrow \text{CONE}(a)$ **否则** $E_a \leftarrow$ 零填充(d) **结束 if**
如果 v 是标量**则** $E_v \leftarrow \text{CONE}(v)$
否则如果 v 是范围 $(a,\ b)$ **则** $E_v \leftarrow$ 拼接(CONE($\frac{a+b}{2}$), CONE($|b-a|$))
否则如果 v 是高斯分布 $(\mu,\ \sigma)$ **则** $E_v \leftarrow$ 拼接(CONE($\mu-\sigma$),CONE(μ), CONE($\mu+\sigma$))
否则 $E_v \leftarrow$ 零填充(d) **结束条件判断**
如果 u 存在 **则** $E_u \leftarrow \text{CONE}(u)$ **否则** $E_u \leftarrow$ 零填充(d) **结束 if**
拼接嵌入：$E \leftarrow [E_a, E_v, E_u]$ 到 $|S|$ 个大小为 size d
对未使用的槽位进行零填充并构建二进制掩码 M
投影：$z \leftarrow W\,E$ // 线性自编码器投影
重构：$\hat{E} \leftarrow W^{\top} z$
计算掩码重构损失：
$L_{\text{embed}} \leftarrow \|M \odot E - M \odot \hat{E}\|_2^2$ (·)：归一化：层归一化(z)
返回 E^{comp}

表 2：通过皮尔逊 r 和斯皮尔曼 ρ 相关系数，比较解析距离（第 5.1 节）与嵌入余弦距离的相关性。

	数字		范围				高斯分布	
	绝对差		欧几里得距离		雅卡尔交并比		瓦瑟斯坦距离	
方法	r	ρ	r	ρ	r	ρ	r	ρ
BioBERT	0.067	0.064	0.398	0.355	0.267	0.248	0.038	0.039
CONE	**0.989**	**0.798**	**0.997**	**0.786**	**0.498**	**0.465**	**0.689**	**0.663**

作品（例如电视剧）。**CIUS** [19] 数据集来自美国犯罪（CIUS）数据库，包含 1,050 个介绍犯罪罪行的表格。**SAUS** [19], 美国人口普查局发布的《2010 年美国统计摘要》（SAUS）包含 1,368 个表格，涉及金融、商业、犯罪、农业和医疗等主题。

我们还在 **DROP** 数据集 [17], 上评估模型。该问答（QA）基准测试包含问题及答案对，用于测试计数、排序和加法等数值推理能力。该数据集包含 77,409 个训练样本、9,536 个开发样本和 9,622 个测试样本。此外，我们使用来自 [33, 44] 的六个数据集 GDC、Magellan、WikiData、Open Data、ChEMBL 和 TPC-DI，评估并比较近期的模式匹配（SM）解决方案 [16, 44]。受篇幅限制，关于这些数据集的统计信息请参阅 [33, 44] 。

5 距离分析

为验证我们针对复杂数值数据的复合嵌入，我们检查其三个组成部分是否各自如预期般分别 对距离产生贡献。因此，我们分别评估了数字值、单位和属性的微调嵌入的有效性。

数字$\mathbf{x} = [x_1, y_2, \ldots, x_d]$和$\mathbf{y} = [y_1, y_2, \ldots, y_d]$对应的两个CONE嵌入向量之间的欧几里得距离 [35]为:

$$D_{xy} = \sqrt{\sum_{i=1}^{d} (x_i - y_i)^2} \tag{12}$$

5.1 解析距离度量

我们定义了用于评估数字、范围和高斯嵌入的距离。

5.1.1 数字（绝对差）。对于两个标量 $x, y \in \mathbb{R}$，距离为

$$d_{\text{num}}(x, y) = |x - y| \tag{13}$$

5.1.2 范围。对于两个范围 $R_i = [a, b]$ 和 $R_j = [c, d]$，我们考虑两种距离:

欧几里得距离 [35]: 中心和长度定义为 $c_i = \frac{a+b}{2}$，$\ell_i = |b - a|$ 和 $c_j = \frac{c+d}{2}$，$\ell_j = |d - c|$，距离为

$$d_{\text{CL}}(R_i, R_j) = \sqrt{(c_i - c_j)^2 + (\ell_i - \ell_j)^2} \tag{14}$$

基于Jaccard的交并比（IoU）距离 [52]:

$$d_{\text{IoU}}(R_i, R_j) = 1 - \frac{|R_i \cap R_j|}{|R_i \cup R_j|} \tag{15}$$

其中交集长度为$\max(0,\ \min(b,\ d) - \max(a,\ c))$，并集长度为$(b - a) + (d - c) - |R_i \cap R_j|$。

5.1.3 高斯分布（*2-Wasserstein*距离）。对于两个高斯分布 $\mathcal{N}(\mu_i, \sigma_i^2)$和$\mathcal{N}(\mu_j, \sigma_j^2)$ 距离为 [54],

$$d_{\mathcal{W}_2}(i, j) = \sqrt{(\mu_i - \mu_j)^2 + (\sigma_i - \sigma_j)^2} \tag{16}$$

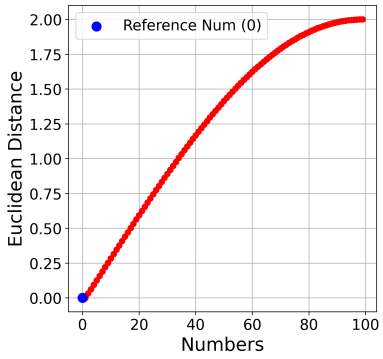


图 5：0 的嵌入与 0 到 100 之间各数字的嵌入之间的欧几里得距离.

5.2 CONE 嵌入组件

5.2.1 数字。图 5 展示了 0 与 0 到 100 之间各个数字所对应的 CONE 嵌入向量之间的成对欧几里得距离。随着数字增大，距离也如预期般相应增大，表明 CONE 能够捕捉数值大小，并有效编码其顺序和区间属性 [66]。距离单调递增（数字越大，与 0 的距离越远），但随着数值增大，其单调性开始呈现非线性。

受篇幅所限，这里展示了 0 到 100 之间数字的结果；但根据我们的实验，对于更大的数字（即 ≈1000），也呈现相同的变化规律。我们还通过比较数字（标量、范围和高斯分布）之间的实际距离（第 5.1 节）与嵌入空间中的相应距离，评估嵌入是否保留数字的实际距离。我们报告线性关联的皮尔逊相关系数（r），以及用于衡量排序一致性的斯皮尔曼秩相关系数（ρ）[36]。结果汇总于表 2。从定量结果来看，将数字（0–1000）的绝对差与相应的嵌入余弦距离进行相关性分析后发现，标准编码器嵌入（例如 BioBERT；$r \approx 0.07$、$\rho \approx 0.06$）呈现较弱的相关性，而 CONE 则具有较强的相关性（$r = 0.989$、$\rho = 0.798$）。
这表明，与标准编码器不同，CONE 能够保留数值大小。

5.2.2 范围。我们从 $[0, 100]$, $[1, 1000]$, 和 $[0, 10000]$ 中均匀采样，生成了 1,000 个随机范围。图6展示了从 $[1, 20]$ 中随机采样的范围对应的 CONE 嵌入经主成分分析（PCA）转换后的二维散点图。可以看到，CONE 保留了范围之间的相对距离：相似的范围（例如“*1-10*”和“*1-11*”）彼此位置接近，体现了它们之间的相似性。同样，其他相邻范围，例如“*2-12*”和“*3-13*”，也嵌入在彼此附近，这符合预期。随着范围左边界增大，它与下界较小且不重叠的范围之间的距离也随之增大，这也是符合预期的正确表现。例如，较大的范围，如“*4-14*”和“*5-15*”，与较小的范围，如“*1-10*”和“*1-11*”，相距较远。此外，我们评估了两种解析距离概念（第 5.1 节）：欧几里得距离（d_{CL}）[35] 和基于杰卡德系数的交并比距离（d_{IoU}）[52],，并将它们与嵌入的余弦距离进行相关性分析。BioBERT 在 d_{CL}（$r = 0.398$，$\rho = 0.355$）和 d_{IoU}（$r = 0.267$，$\rho = 0.0.248$）上呈现中等相关性，而 CONE 在 d_{CL}（$r = 0.997$，$\rho = 0.786$）上达到极高相关性，并在 d_{IoU}（$r = 0.498$，$\rho = 0.465$；表 2）上呈现显著更强的一致性。这些结果表明，CONE 嵌入空间能够准确编码范围大小及其相对距离。

5.2.3 高斯分布。 我们从 $\mu \sim \mathrm{U}(0, 100)$ 和 $\sigma \sim \mathrm{U}(1, 10)$ 中采样了 500 个高斯分布。随后，我们将嵌入余弦距离与 2-Wasserstein距离进行了比较 [54]。BioBERT 嵌入与实际预期距离不一致（$r \approx 0.04$）。相比之下，CONE 显著提升了相关性（$r = 0.689$、$\rho = 0.663$；表2），表明嵌入空间能够捕捉这些分布属性，不过其表现仍逊于标量和范围。

5.2.4 属性。 为评估属性级语义相似度，我们从数据集中采样了 100 个属性（即列名），并使用 CONE 比较它们的嵌入距离。由于篇幅限制，我们在图7所示的热图中展示了 12 个有代表性的属性。颜色深浅（从蓝色到红色）表示成对相似度，其中深红色表示相似度较高（值接近 1.0），蓝色表示相似度较低。语义相似的属性（例如血红蛋白 和糖化血红蛋白 ）距离更近，如红色区域所示。这反映了它们在嵌入向量空间中的邻近关系，也符合它们在语义上的相关性。

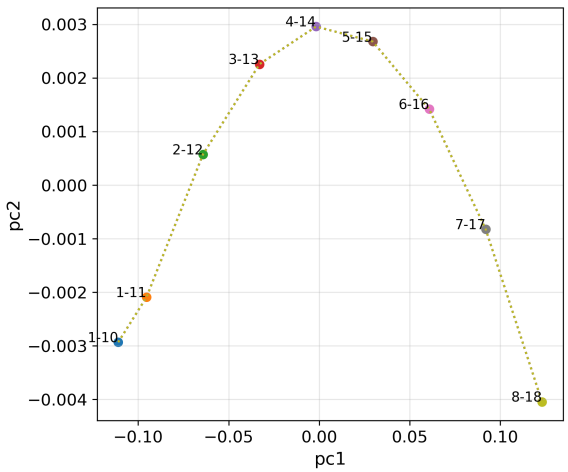


图6：经主成分分析变换的数值范围 CONE 嵌入 $[1, 20]$（不含单位）。

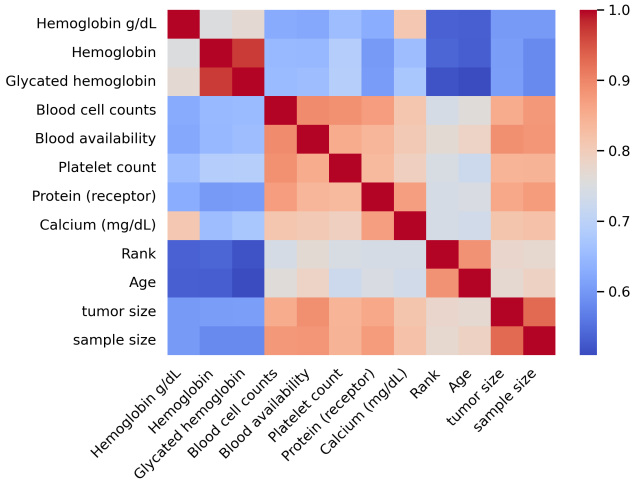


图7：属性与 CONE 嵌入的相似度。

另一方面，不同属性（如血红蛋白 和年龄 ）的相似度较低，以蓝色区域表示，说明它们的嵌入之间距离较远。这表明不同属性之间保持了适当的距离。由于此评估仅依赖属性名称，肿瘤大小 和 样本量 看起来相对接近，尽管它们指的是本质上不同的量（肿瘤测量值与总体计数）。这是预训练文本编码器在未考虑数值和单位上下文时的典型表现。CONE 的复合嵌入设计通过在列数据完整时整合属性、单位和数值组成部分，解决了这一局限（评估结果见表5）。我们还发现，单位可能会明确出现在属性名称中（例如，血红蛋白 *g/dL*），也可能隐含在列值中。仅考虑属性相似度时，血红蛋白 *g/dL* 和血红蛋白 的距离稍远，尽管它们代表相似的属性。我们的三部分复合嵌入结构解决了这一差异，其中

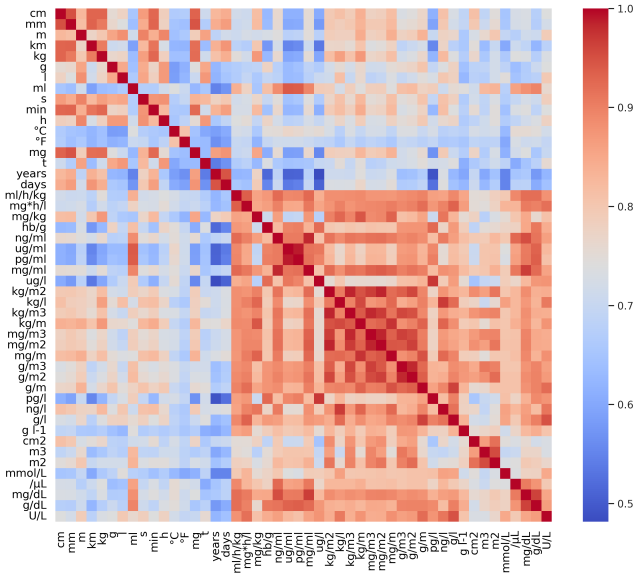


图8：使用 CONE 嵌入表示的单位相似度。

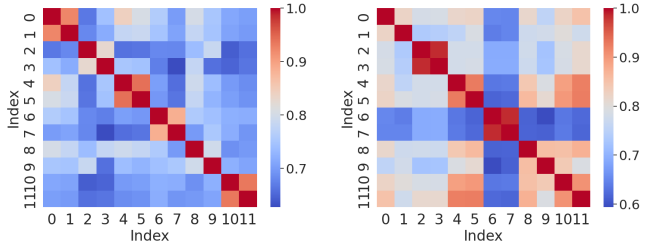
各部分能够从列数据中提取缺失的单位，并将其与属性语义联合编码。有趣的是，由于共享相同的单位（*g/dL* 或 *mg/dL*），一些属性（如血红蛋白 *g/dL*和钙（*mg/dL*））的相似度相对较高，尽管它们代表不同的生物学测量指标。将这三个组成部分合并到同一个向量中，也能解决这一问题。将单位嵌入与属性嵌入拼接起来，将进一步增强模型正确、精确地理解和区分数值的能力。我们将 CONE 的属性相似度结果与微调后的 BioBERT 进行比较，未发现明显差异，这证实了 CONE 在有效编码文本数据的同时，也能捕捉数值和单位级语义。

5.2.5 单位。为了评估我们的单位嵌入，我们从数据集中抽取了 50 个不同的单位，并对两两余弦距离进行了可视化。图 8 展示了不同单位之间的距离。长度单位，如 *cm*、*mm*、*km* 和 *m* 彼此高度相似（以红色区域表示），反映出它们在物理测量方面的相关性。同样，重量相关单位，如*kg* 和 *g* 彼此位置接近，这符合预期，因为它们测量的是相同的物理量。时间相关单位，如 *h*（小时）、*min*（分钟）和 *s*（秒），也表现出较高的相似度，表明它们通过时间测量相互关联。*°C*（摄氏度）和 *°F*（华氏度）等单位则表现出中等相似度，反映出它们同属温度单位，但采用不同的尺度。生物和化学测量中常用的单位，如 *mg/mL*、*ng/mL*和 *pg/mL*，表现出显著的相似度；这是意料之中的，因为它们表示与质量浓度相关的量，只是尺度不同。另一方面，*mg/kg* 和 *mg/dL*等单位与 *mm* 和 *cm* 等单位相对不同，如蓝色区域所示。这种区别是合理的，因为这些单位测量的是完全不同的物理

Index	Ranges	Gaussian
0	Blood Pressure: 120-140 mmHg	Particle Size: 1302.0 ± 0.25 nm
1	Blood Pressure: 125-145 mmHg	PS: 344.5 ± 15.6 nm
2	Temperature: 36.5-37.5 °C	Elasticity: 112 ± 1.3 mg s^{-1} cm^{-2}
3	Temperature: 30-40 °C	Elasticity: 127 ± 4.3 mg s^{-1} cm^{-2}
4	Heart Rate: 60-80 bpm	Blood loss: 229 ± 128 ml
5	Heart Rate: 50-80 bpm	Blood loss: 207 ± 188 ml
6	Dosage: 4-5 ml	Ferritin: 13.8 ± 1.3 ng/ml
7	Dosage: 0.5-1 ml	Ferritin: 14.3 ± 1.4 ng/ml
8	Dose: 40-50 mg	Overall survival: 4.4 ± 0.5 years
9	Drug Dose: 37.6-50.0 mg	Overall survival (OS): 42 ± 8 years
10	Injection time: 1-2 days	Mean operating time ± SD: 98.4 ± 60.4 min
11	Time of injection: 1-3 days	Operating time: 231 ± 60 min

图 9：示例数值范围和高斯分布。

量（质量、长度和浓度）。这展示了单位嵌入组件的作用。

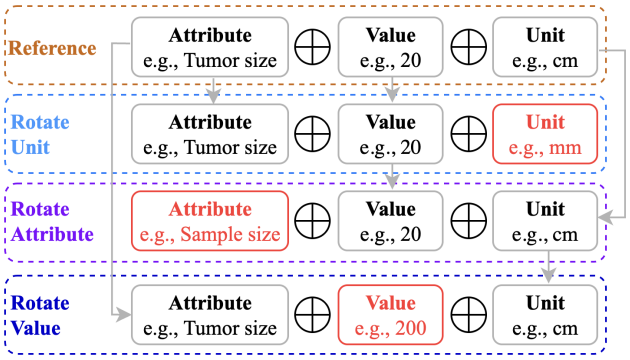


(a) **(b)**

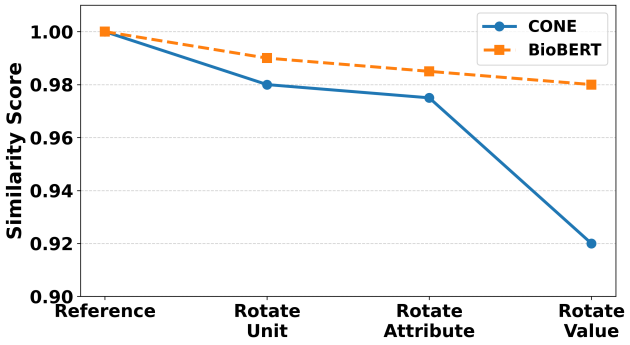
图10：图 9 中带有属性和单位的数值（a）范围和（b）高斯分布的 CONE 复合嵌入之间的距离。

接下来，为了验证我们针对范围设计的复合嵌入结构，我们展示它通过纳入结构上下文（属性、单位、范围中心和长度）捕捉复合语义的能力。图 9 展示了从我们的数据集中抽取的示例范围和高斯分布，以及与之关联的属性和单位。在图 10(a) 的热图中，结构上下文相似的范围（例如，索引 0，血压：*120-140* 毫米汞柱 和索引 1，血压：*125-145* 毫米汞柱）具有较高的相似度，因为它们在复合嵌入空间中彼此更接近。相反，尽管索引 6（剂量：*4-5* 毫升）、7（剂量：*0.5-1* 毫升）和 8（剂量：*40-50* 毫克）具有相似的属性，但索引 8 的单位不同，因此在复合嵌入空间中与索引 6 和 7 的距离更远，这是合理的。这表明复合嵌入避免了文本相似性单独占据主导地位；否则，这些索引之间的距离会近得多。索引 6 和 7 的嵌入比索引 8 和 9 的嵌入更接近（ 0.89 对比 0.8 ），这主要是由于属性措辞略有差异（剂量 与药物剂量）。尽管如此，索引 8 和 9 之间的相似度得分仍高于将索引 8 或索引 9 与图 9 中任何其他索引进行比较时得到的得分。重要的是，移除属性后，数值范围仍保留比例距离（表 2）。

同样，图 10(b) 热图展示了图 9 中各高斯分布复合嵌入之间的距离。上下文结构相似的高斯分布对（例如，索引 6，铁蛋白： 13.8 ± 1.3 纳克/毫升 和索引 7，铁蛋白： 14.3±1.4 纳克/毫升）的距离比其他配对更近。我们还观察到一些临床参数可能会形成相关性较强的分组，例如失血量 （索引：4，



(a)



(b)

图11：对CONE复合嵌入中的各个组件进行“旋转”（见第5.2.6节）的效果：(a) 逐组件“旋转”示意图；(b) 与原始复合嵌入相比，余弦相似度的变化。

5）、总生存期（索引：8、9）和手术时间。然而，与对应的相似索引对之间的距离相比，这些组之间的距离更小。

5.2.6 复合组件的消融研究。最后，我们通过观察在其他组件保持不变时扰动某一组件所引起的相似度得分（嵌入距离）变化，分析复合嵌入中各组件的重要性。我们将这一过程称为“旋转”。具体而言：（1）固定属性名称和值，对单位进行“旋转”；（2）固定单位和值，对属性进行“旋转”；（3）固定属性和单位，对值进行“旋转”。以一个元组（属性、值和单位）作为参考示例，其内容为（‘肿瘤大小’、‘*20*’、‘*cm*’）（图11(a)展示了复合输入参考示例，而图11(b)展示了每个组件“旋转”时相对于原始嵌入的余弦相似度距离变化）。旋转单位（$cm \rightarrow mm$）会使相似度从1.0小幅下降至0.98。旋转属性（肿瘤大小 $\rightarrow$ 样本量）会使相似度降至0.975，而旋转值（$20 \rightarrow 200$）导致的下降最大，降至0.92。这项分析表明，值组件对嵌入的影响最为显著，说明它在保留语义含义方面发挥着关键作用。属性和单位的影响则较为适中。这些逐组件变化全面展现了嵌入的敏感性，并进一步验证了我们的设计：每个组件（属性、值和单位）在复合嵌入结构中都以独特方式促进了对复杂结构化数值数据语义的捕捉。需要注意的是，各组件的相对贡献可能会因具体的结构上下文而有所不同。从图11(b)可以看出，CONE比BioBERT更有效地捕捉语义结构，这主要得益于其将数值上下文融入嵌入中。

6 实验评估

为考察CONE的数值属性，我们针对列表最大值、解码和加法这三项合成任务进行评估，这些任务由 [71]提出。随后，我们在DROP问答基准[17], 上评估性能，该基准要求具备计数、排序和加法等数值推理能力。最后，我们使用大规模数据集，通过衡量复合嵌入表示在结构化数据上的两项热门下游任务中的表现，评估其质量，并将其与SOTA基线方法进行比较。

6.1 实验设置

实现：我们在一台配备 112 核 Intel Xeon Platinum 8180 2.50 GHz CPU、3 块 NVIDIA V100 GPU、1.5 TB 内存和 16 TB 磁盘空间的服务器上进行了实验。对于编码器，我们使用了 BioBERT-baseuncased，其隐藏层为 = 12，隐藏单位为 = 768 = 12 ，以及注意力预测头。我们使用 Adam 优化器，学习率为 2×10^{-5}，从 $\{1.5 \times 10^{-5}, 2 \times 10^{-5}\}$ 中选取，批量大小为 12。最大序列长度设为 512。我们采用了 BioBERT 的词汇表和预训练嵌入权重，并在我们的训练数据集上对其进行了微调。BioBERT 的配置与 $BERT_{\text{BASE}}$ 模型保持一致。我们训练了 50,000 步（≈ 3 天），超参数如上所述；掩码数字预测损失在最后一步之前就已充分稳定。训练集由表格元组/列组成。每个元组/列均按照 [34] 进行分词、嵌入和编码。为应对词元长度限制，超过 512 个词元的输入被分块处理，并在每个元组/列的开头放置 [CLS]，在单元格之间插入 [SEP] 词元。

6.2 数值推理

6.2.1 探测嵌入的数值能力。 我们首先通过在三个数值能力任务——列表最大值、解码和加法 [71]——上的评估，考察 CONE 的数值属性。每项任务都使用 CONE 的最终上下文化数字嵌入 E^O，结果取三个不同随机种子运行的平均值。在列表最大值任务中，模型使用一个两层多层感知机（MLP），从五个输入数字中找出最大值的索引。解码任务则通过一个五层 MLP，将单个数字的嵌入回归为其原始值。对于加法任务，模型使用类似的五层网络，根据两个数字的嵌入预测它们的和。我们从 $[0, 10^2]$ 至 $[0, 10^6]$ 的范围内采样整数，并采用 80%/20% 的训练测试划分，将 [71] 中的评估方案扩展至更大的数值范围。我们对更宽范围（例如 10^8）进行的实验也呈现出类似的性能趋势，表明 CONE 的相对优势在规模扩大时依然稳定。因此，$[0 - 10^6]$ 这一范围已足够，因为它也涵盖了常见数值能力基准中遇到的大多数数值大小

表 3：列表最大值、解码和加法的实验结果。

	列表最大值（准确率）					解码（RMSE）					加法（RMSE）				
整数范围	$[0, 10^2]$	$[0, 10^3]$	$[0, 10^4]$	$[0, 10^5]$	$[0, 10^6]$	$[0, 10^2]$	$[0, 10^3]$	$[0, 10^4]$	$[0, 10^5]$	$[0, 10^6]$	$[0, 10^2]$	$[0, 10^3]$	$[0, 10^4]$	$[0, 10^5]$	$[0, 10^6]$
随机向量	$0.17_{\pm 0.030}$	$0.22_{\pm 0.030}$	$0.20_{\pm 0.035}$	$0.18_{\pm 0.038}$	$0.16_{\pm 0.039}$	$30.1_{\pm 1.80}$	$295.88_{\pm 15}$	$2888.62_{\pm 42}$	$28k_{\pm 1k}$	$280k_{\pm 5k}$	$41.80_{\pm 0.50}$	$412.33_{\pm 12}$	$4400.39_{\pm 55}$	$40k_{\pm 2k}$	$405k_{\pm 8k}$
BioBERT	$0.94_{\pm 0.025}$	$0.63_{\pm 0.028}$	$0.50_{\pm 0.022}$	$0.47_{\pm 0.030}$	$0.42_{\pm 0.032}$	$3.28_{\pm 0.10}$	$29.7_{\pm 0.80}$	$434_{\pm 10}$	$4.1k_{\pm 0.2k}$	$40.5k_{\pm 2k}$	$4.61_{\pm 0.20}$	$69.2_{\pm 2.10}$	$461_{\pm 12}$	$4.2k_{\pm 0.3k}$	$42.1k_{\pm 3k}$
DICE	$0.97_{\pm 0.002}$	$\mathbf{0.96_{\pm 0.001}}$	$0.96_{\pm 0.002}$	$0.95_{\pm 0.002}$	$0.94_{\pm 0.003}$	$0.45_{\pm 0.03}$	$0.83_{\pm 0.03}$	$3.02_{\pm 0.08}$	$5.42_{\pm 0.15}$	$11.2_{\pm 0.4}$	$\mathbf{0.64_{\pm 0.02}}$	$3.80_{\pm 0.15}$	$29.2_{\pm 1.2}$	$55.1_{\pm 2.1}$	$108.4_{\pm 4}$
NTL-WAS	$0.95_{\pm 0.003}$	$0.94_{\pm 0.003}$	$0.92_{\pm 0.002}$	$0.91_{\pm 0.002}$	$0.88_{\pm 0.003}$	$\mathbf{0.30_{\pm 0.03}}$	$\mathbf{0.49_{\pm 0.12}}$	$2.75_{\pm 0.15}$	$18.4_{\pm 1.20}$	$45.0_{\pm 2.5}$	$1.26_{\pm 0.03}$	$4.25_{\pm 0.25}$	$36.4_{\pm 1.8}$	$82.4_{\pm 4.1}$	$158.2_{\pm 8}$
CONE	$\mathbf{0.98_{\pm 0.002}}$	$0.95_{\pm 0.002}$	$\mathbf{0.97_{\pm 0.001}}$	$\mathbf{0.96_{\pm 0.001}}$	$\mathbf{0.95_{\pm 0.003}}$	$0.42_{\pm 0.02}$	$0.73_{\pm 0.03}$	$\mathbf{2.71_{\pm 0.05}}$	$\mathbf{4.89_{\pm 0.10}}$	$\mathbf{9.85_{\pm 0.3}}$	$\mathbf{0.64_{\pm 0.02}}$	$\mathbf{2.18_{\pm 0.08}}$	$\mathbf{26.10_{\pm 0.9}}$	$\mathbf{48.9_{\pm 1.8}}$	$\mathbf{95.1_{\pm 3}}$

表 4：DROP数据集上的结果（%）[1]

方法	Dev		Test	
	EM	F1	EM	F1
NumNet [56]	83.34±0.2	86.00±0.38	83.40±0.18	86.42±0.4
NC-BERT [31]	75.18±1.2	77.68±0.75	76.02±0.9	77.91±0.8
AeNER [75]	**83.72**	86.56	83.69	86.98
CONE	83.71±0.12	**86.70±0.2**	**83.74±0.12**	**87.28±0.3**

。列表最大值任务使用准确率评估，解码和加法任务则使用均方根误差（RMSE）评估。表 3 中的结果（均值和标准差）表明，CONE 在大多数情况下优于其他基线方法，说明其复合嵌入具有很高的数值能力理解准确度。为评估统计显著性，我们在测试集上进行了配对自助重采样（$n = 1000$） [1], ，并构建了强基线方法（DICE 和 NTL-WAS）与 CONE 之间性能差异的 95% 置信区间（CI）。受篇幅限制，我们省略了全部 CI 结果，仅报告一个代表性示例：在 $[0, 10^6]$, 的加法任务中，CONE 的 RMSE 比 DICE 低 $\Delta = 13.3$，其 95% CI 为 $[9.8, 16.7]$, 该区间不包含零，因此表明这一提升具有统计显著性。不同方法会根据其数值属性编码方式，在不同任务上表现出色。对于列表最大值任务，在需要直接比较数值大小时，采用显式量级编码的方法（例如我们的方法）表现最为出色。对于解码任务，NTL-WAS [78] 因其数字级重构损失而表现良好；在不同范围内，我们的方法与其表现相当或接近。对于加法任务，我们方法中的显式位置编码和量级重构损失（即公式 9）有助于提升性能。

*6.2.2 DROP*数据集上的数值推理。在本节中，我们通过衡量需要数值推理的问题的准确率，评估所提出的 CONE 模型在DROP数据集上的表现。按照[31, 56, 75], ，我们报告精确匹配（EM）和 F1 分数，并与具备数值推理能力的仅编码器基线方法进行比较——NumNet [56], NC-BERT [31] 和 AeNER [75] ——详见第 7 节。所有实验均以三个随机种子的结果取平均值，并报告均值和标准差。由于 AeNER 未提供公开代码，我们直接引用其原论文中的结果。表 4 显示，CONE 的表现优于 NumNet，EM 高出 0.34，F1 高出 0.86；优于 NC-BERT，测试集上的 EM 高出 7.72，F1 高出 9.37。与 AeNER 相比，CONE 略胜一筹，EM 高出 0.05，F1 高出 0.3。在三个随机种子下，CONE 在测试集 F1 上显著优于 NumNet（配对 t-检验，$p < 0.01$），证实了我们数值嵌入设计的有效性。

[1]最佳结果以粗体显示，最大的差值 Δ 以红色突出显示。

6.3 列与元组匹配

这项下游任务评估我们的复合嵌入能否准确识别语义和数值上相似的列与元组。给定一个参考列或元组，目标是根据嵌入相似度检索出与之匹配的项。

6.3.1 基线方法。 我们将结果与若干 SOTA 方法进行比较，包括在我们的数据集上分别微调的 TAPAS [25], NumNet [56], 和 NC-BERT [31], 。TAPAS 是一种用于表格推理的 Transformer 模型。NumNet [56] 构建数值比较图，在有向边上编码数字之间的相对量级信息。我们在实验中使用了 NumNet+ 模型，它是 NumNet 的增强版本，采用了预训练的 RoBERTa 模型 [43]。NC-BERT [31] 将 DICE 用作损失函数，以恢复表示中数字的大小特征。NC-BERT 是一种编码器-解码器模型。因此，为了进行无偏比较，我们在实验中仅使用编码器输出。

我们还评估了四种通用检索嵌入模型：BGE-M3 [11], Stella（Stella-EN-1.5B-v5） [79],Qwen3（Qwen3-Embedding-4B） [81], 和 KaLM（KaLM-Embedding-Gemma3-12B-2511） [26, 82]。这些模型广泛用于异构文本语料库中的语义检索和相似度匹配。此外，我们将 CONE 与若干 SOTA 模式匹配（SM）方法进行了比较，包括 Valentine [33]中实现的传统方法 SimFlooding[48], ；ISResMat [16]；近期方法 Magneto [44]；以及表格基础模型 CARTE [32]。CARTE 将表格行转换为图表示，并利用预训练的图注意力 Transformer 生成上下文感知嵌入。SimFlooding 是一种基于图的方法，通过不动点计算来确定模式匹配结果；ISResMat 则采用对比学习和微调后的 BERT。然而，在所有列/元组匹配实验中，SimFlooding 和 ISResMat 的表现始终显著逊于其他方法，因此为清晰起见，我们不在报告的结果中列出它们。Magneto 使用小语言模型（SLM）进行候选检索，并使用大语言模型（LLM）进行重排序。我们在零样本和微调两种配置下评估了 Magneto。由于微调版本的表现始终优于零样本设置，为简洁起见，我们仅报告微调后的结果。

6.3.2 评估流程。我们从数据集（第 4 节）中的表格构建列和元组集合，这些表格包含严格的数值或文本-数值混合值。我们使用 [30]的方法识别属性（例如血红蛋白、肿瘤大小、样本量、失血量、血小板计数等）。数值及其对应的单位符号通过正则表达式和 [10] 中的技术进行解析。为处理不一致的单位表示形式（例如，毫升与mL），我们应用单位规范化 [10],, 以

表5：列匹配（CM）和元组匹配（TM）的平均Recall@10（R）、MAP@10和MRR@10[1]。

方法	Task	CancerKG			CovidKG			WebTable			CIUS			SAUS		
		R	MAP	MRR	R	MAP	MRR	R	MAP	MRR	R	MAP	MRR	R	MAP	MRR
TAPAS	CM	0.783	0.728	0.742	0.734	0.669	0.688	0.800	0.736	0.753	0.820	0.771	0.789	0.820	0.774	0.787
	TM	0.736	0.683	0.701	0.700	0.642	0.658	0.800	0.733	0.745	0.800	0.751	0.764	0.800	0.752	0.767
NumNet	CM	0.650	0.581	0.596	0.700	0.628	0.641	0.700	0.612	0.628	0.758	0.686	0.703	0.780	0.707	0.724
	TM	0.700	0.619	0.634	0.667	0.598	0.611	0.723	0.629	0.639	0.750	0.675	0.686	0.700	0.624	0.641
NC-BERT	CM	0.792	0.758	0.768	0.750	0.694	0.712	0.850	0.788	0.795	0.850	0.792	0.809	0.850	0.793	0.804
	TM	0.767	0.723	0.737	0.735	0.686	0.697	0.800	0.739	0.748	0.850	0.794	0.804	0.800	0.743	0.754
Magneto	CM	0.784	0.741	0.770	0.724	0.685	0.718	0.838	0.796	0.802	0.820	0.760	0.800	0.810	0.762	0.796
	TM	0.742	0.691	0.738	0.700	0.662	0.675	0.780	0.706	0.717	0.800	0.751	0.764	0.750	0.712	0.725
CARTE	CM	0.750	0.690	0.698	0.730	0.692	0.682	0.850	0.802	0.800	0.825	0.780	0.794	0.817	0.769	0.782
	TM	0.754	0.706	0.722	0.700	0.633	0.676	0.825	0.764	0.777	0.820	0.768	0.780	0.800	0.743	0.754
BGE-M3	CM	0.795	0.761	0.771	0.750	0.713	0.726	0.902	0.806	0.811	0.853	0.795	0.802	0.852	0.795	0.802
	TM	0.796	0.758	0.769	0.740	0.703	0.716	0.850	0.797	0.807	0.850	0.794	0.804	0.800	0.755	0.766
Stella	CM	0.798	0.760	0.773	0.750	0.712	0.726	0.900	0.806	0.813	0.855	0.798	0.810	0.852	0.795	0.806
	TM	0.798	0.760	0.771	0.742	0.705	0.718	0.850	0.797	0.807	0.850	0.794	0.804	0.810	0.759	0.766
Qwen3	CM	0.800	0.770	0.781	0.754	0.716	0.729	0.910	0.810	0.820	0.855	0.798	0.810	0.852	0.797	0.806
	TM	0.800	0.762	0.774	0.745	0.708	0.721	0.852	0.799	0.810	0.857	0.803	0.813	0.815	0.761	0.772
KaLM	CM	0.800	0.770	0.781	0.760	0.722	0.735	0.910	0.810	0.820	0.860	0.804	0.818	0.856	0.801	0.811
	TM	0.810	0.775	0.786	0.750	0.713	0.726	0.858	0.806	0.816	0.860	0.806	0.816	0.815	0.765	0.772
CONE	CM	**0.833**	**0.817**	**0.831**	**0.800**	**0.782**	**0.798**	**0.950**	**0.842**	**0.858**	**0.900**	**0.851**	**0.864**	**0.900**	**0.849**	**0.861**
	TM	**0.867**	**0.844**	**0.853**	**0.800**	**0.781**	**0.792**	**0.900**	**0.854**	**0.866**	**0.900**	**0.855**	**0.866**	**0.850**	**0.803**	**0.815**

表 6：模式匹配（SM）准确率；数据集 [33, 44]。

数据集	ISResMat		SimFlooding		Magneto		CONE	
	R	MRR	R	MRR	R	MRR	R	MRR
GDC	0.24	0.30	0.24	0.29	0.50	0.87	0.65	0.85
Magellan	0.98	0.99	1.00	1.00	1.00	1.00	1.00	1.00
WikiData	0.85	0.96	0.65	0.85	0.90	0.95	0.94	0.95
开放数据	0.60	0.78	0.45	0.75	0.75	0.95	0.88	0.90
ChEMBL	0.65	0.82	0.45	0.68	0.86	0.97	0.91	0.95
TPC-DI	0.82	0.93	0.68	0.82	0.85	0.97	0.92	0.94

将所有变体规范化为单一规范形式。对于缺少单位的条目，我们根据周围列或元组的上下文推断最可能的单位 [10]。当某列或元组中的所有条目都缺少单位信息时，我们为复合嵌入中的单位分量分配一个零填充向量。完成这一步预处理后，我们为数据集中的所有列和元组计算复合嵌入（第 3.3 节的算法 2）。

为实现可扩展的向量检索，我们集成了向量数据库系统 Meta Faiss [15]，它通过 IndexFlatIP 使用平面索引对 CONE 嵌入进行索引。我们在索引前对所有嵌入进行 L2 归一化，使内积搜索对应于余弦相似度。该方法将向量编码为固定大小的表示，并存储在连续数组中，从而支持高吞吐量的相似度搜索。针对 20 万个已索引向量查询一列，仅需 8.57 毫秒即可检索出最相似的前 10 列，展现了良好的可扩展性和低延迟。

我们在五位志愿者和两位独立领域专家的协助下，人工整理了真实匹配结果；对于专家间意见不一致的情况，则从最终评估集中排除。由此得到的基准测试包含 200 列和 200 个元组，并以数值作为参考。对于每个参考列或元组，我们根据其嵌入向量的余弦相似度 检索并排列前-K（$K = 10$）个候选匹配结果，按降序排列。我们使用 Recall@10 [31], 平均精度均值（MAP@10）[40],和平均倒数排名（MRR@10） [12]评估这些排序的质量。Recall@K 衡量前-K个候选结果中检索到的相关结果所占比例，而 MAP 和 MRR 则同时反映精确率和多个查询的排序质量。所有指标均基于按余弦相似度降序排列的聚类列/元组列表计算，并在所有参考输入上取均值。

我们使用余弦相似度来评估元组或列的嵌入之间的距离；它衡量的是向量之间的角度。余弦相似度是基于嵌入的聚类和分类方案中一种稳定且广泛使用的度量 [30, 38, 55, 57, 59,69] ，因为与其他方法 [57]不同，它不受表中行或列大小的影响。如果只依赖向量大小而不考虑角距离，内容相似的行或列在嵌入空间中可能会显得差异显著。其他相似度度量，例如欧几里得距离 [35],，用于衡量向量空间中两个向量之间的绝对距离，且会受其大小影响；也就是说，语义相似但尺度不同的两个向量可能会显得相距很远。杰卡德相似度 [52] 专为比较集合而设计，并根据重叠元素所占比例量化相似度，这不同于向量空间中使用的语义相似度度量。

6.3.3 结果。 表5展示了 CONE 与最先进模型的实验结果，其中最佳结果以粗体标出，最差与最佳得分之间的差值以红色显示。从图1可以看出，采用 CONE 的新型复合嵌入结构后，图 2 中 年龄 与 随访（月） 之间的相似度得分从 0.99（使用 BioBERT 嵌入）降至 0.83（使用 CONE 嵌入，见图1中的橙色和绿色柱），这是显著的。使用 BioBERT 嵌入时，这些属性之间的距离过小，几乎无法区分（图1）。相比之下，使用 CONE 嵌入计算出的新距离（0.83）能够将它们区分开来，符合预期，因为这些属性在语义上并不相同。

需要注意的是，我们的复合嵌入能够抵抗属性命名异质性。它们能够理解并正确匹配具有不同属性名称的相同列/元组（例如，‘失血量（毫升）’、‘输血量’等，在

表7：列匹配（列匹配）和元组匹配（元组匹配）的消融研究：平均Recall@10（R）、MAP@10 和平均倒数排名@10[1]。

数据集	Task	CONE$_1$			CONE$_2$			CONE$_3$			CONE$_4$			CONE		
		R	MAP	MRR	R	MAP	MRR	R	MAP	MRR	R	MAP	MRR	R	MAP	MRR
CancerKG	CM	0.800	0.789	0.808	0.766	0.764	0.772	0.790	0.773	0.786	0.797	0.782	0.790	**0.833**	**0.817**	**0.831**
	TM	0.833	0.800	0.804	0.700	0.783	0.791	0.820	0.790	0.802	0.830	0.796	0.807	**0.867**	**0.844**	**0.853**
CovidKG	CM	0.767	0.752	0.770	0.733	0.732	0.746	0.760	0.748	0.771	0.765	0.754	0.768	**0.800**	**0.782**	**0.798**
	TM	0.700	0.724	0.736	0.700	0.658	0.688	0.720	0.705	0.725	0.752	0.723	0.755	**0.800**	**0.781**	**0.792**
WebTable	CM	0.900	0.806	0.814	0.900	0.801	0.802	0.920	0.818	0.832	0.920	0.821	0.835	**0.950**	**0.842**	**0.858**
	TM	0.880	0.820	0.830	0.870	0.821	0.830	0.876	0.821	0.831	0.878	0.823	0.832	**0.900**	**0.854**	**0.866**
CIUS	CM	0.800	0.790	0.829	0.850	0.798	0.808	0.875	0.821	0.831	0.882	0.825	0.834	**0.900**	**0.851**	**0.864**
	TM	0.850	0.800	0.810	0.837	0.782	0.790	0.885	0.826	0.836	0.880	0.823	0.833	**0.900**	**0.855**	**0.866**
SAUS	CM	0.850	0.815	0.812	0.850	0.801	0.800	0.884	0.826	0.835	0.880	0.824	0.833	**0.900**	**0.849**	**0.861**
	TM	0.800	0.784	0.793	0.781	0.753	0.763	0.830	0.794	0.792	0.830	0.786	0.797	**0.850**	**0.803**	**0.815**

图12(a))。此外，即使某一对中有一方的属性名称采用缩写形式，我们仍能匹配列/元组（例如，图12(b)中的'身体质量指数（千克/平方米'和'身体质量指数（千克/平方米）'，然而基线方法未能识别这一匹配结果）。同样，对于列胆固醇_低密度脂蛋白 毫克/分升，对属性名称和单位进行朴素文本嵌入会检索到若干不相关属性（例如，胆红素_总、血红蛋白、白蛋白、游离_*T4*等），而CONE避免了此类虚假匹配结果，并正确检索到相关属性（例如胆固醇_高密度脂蛋白）。总之，属性命名差异可由上下文嵌入自然处理，这些嵌入基于上下文捕捉属性空间中的语义相似度。

列匹配（列匹配）： CONE 在所有数据集的列匹配任务中均优于所有基线方法（表5）。它取得了最高的 Recall@10、MAP@10 和 MRR@10 分数；在 Webtables 上，其无表 Recall@10 比 NumNet 高出 25%。CONE 在 Webtables 上的 Recall@10 最高，达到 95%。

元组匹配（TM）： 在元组匹配方面，CONE 在所有指标和数据集上的整体表现均为最佳。在 Webtables 上，CONE 的 Recall@10 比 NumNet 高出 17.7%，优势显著。

MAP 和平均倒数排名的持续提升进一步证实，CONE 不仅能检索到正确的列/元组，还能更准确地对排名靠前的结果进行排序。

模式匹配（SM）： 我们在 [44] 中的六个数据集上进行了额外的模式匹配实验。结果如表6所示。与所有基线方法相比，CONE 的召回率（R）相当或更高，证明了其对属性、数值和单位语义进行显式编码的有效性。尽管 Magneto 由于采用基于大语言模型的重排序阶段而取得了略高的平均倒数排名，但我们的方法无需调用大语言模型即可取得相当的平均倒数排名，因此是一种更具成本效益的替代方案。

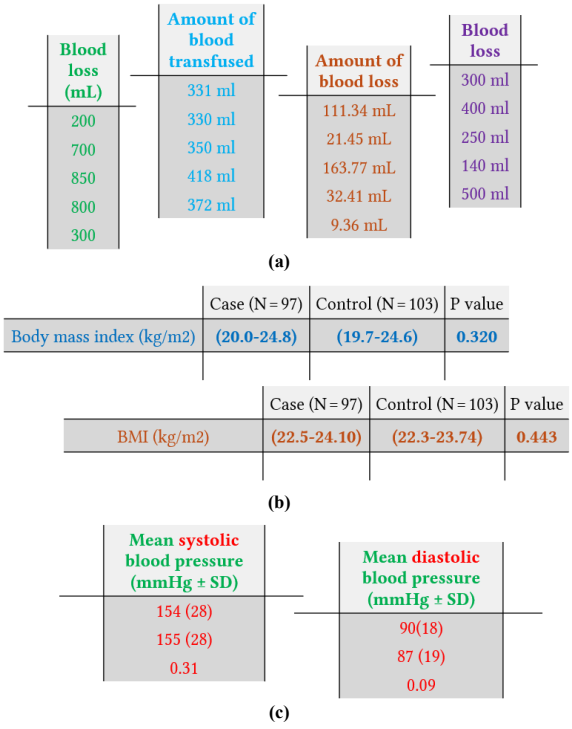


图12：CONE 复合嵌入正确匹配语义相似的复杂对象示例：(a) 列；(b) 元组；(c) 列匹配的一个挑战性案例。

6.4 消融研究

我们评估了 CONE 的四种变体：CONE$_1$, 移除了数值模块（即数值嵌入组件，图3中的虚线框）；CONE$_2$,移除了复合嵌入结构（图4(a)），因此不再联合嵌入属性、单位和值；CONE$_3$, 从复合嵌入中移除了单位组件；CONE$_4$, 移除了对数值范围和高斯分布的编码。表7展示了我们的观察结果：对于所有数据集上的列匹配和元组匹配任务，移除任何组件都会导致召回率、MAP 和平均倒数排名下降。CONE$_1$ 显示，在 CIUS 上的列匹配任务和 CovidKG 上的元组匹配任务中，召回率下降了 10%。CONE$_2$ 显示了最显著的元组匹配性能下降，在 CancerKG 上召回率降低了 16.7%。CONE$_3$ 使 CovidKG 上元组匹配任务的召回率最多降低 8%。最后，CONE$_4$ 使 CovidKG 上元组匹配任务的召回率最多降低 4.8%。消融研究结果表明，移除数值模块或复合嵌入结构都会严重损害性能。

6.5 挑战性案例

图12(c)展示了一个挑战性案例（CovidKG 数据集）的示例。我们观察到，属性名称高度相似的列（例如，'平均收缩压'与'平均舒张压'，其中 ≈ 75%的词汇重叠以绿色突出显示）以及单位相同的列（例如，毫米汞柱 ± 标准差）会被嵌入到彼此接近的位置

尽管它们的值存在显著差异。关键原因在于，带有此类单位的列在数据集中出现得极为罕见（例如，带有该单位的收缩压与舒张压 属性在整个CancerKG数据集中均仅出现一次）。此类罕见实体的嵌入向量不准确是任何架构的常见缺陷，而不仅是我们的架构，因为模型需要实体出现合理的次数，才能收集形成准确嵌入向量所需的不同上下文。出现次数越少，在任何现有最先进架构中，嵌入向量表示就会越不准确。

7 相关工作

传统词嵌入技术 [14, 46, 50, 53] 在语言建模中表现出色，但对数字的表示并不充分。它们将数字视为普通词语，忽略了其大小、单位和上下文 [51], ，而且数字在训练语料库中往往代表性不足——GloVe的400,000个嵌入中仅有3.79%对应于数值词元 [53]。

多项研究提出了数值感知模型。 [65] 使用连续密度函数单独对数字进行建模，从而改善困惑度与预测准确率。 [28] 通过高斯混合模型或自组织映射诱导原型嵌入，并将其整合到词嵌入训练中，以解决数字的词汇外问题。DICE（确定性独立于语料库的嵌入）[67] 手工构造嵌入，利用余弦和欧氏度量保留数值距离，从而辅助比较和加法等任务。我们的方法在这些基础上，通过将复杂数值数据分解为其属性、值及关联单位来进行编码，从而实现更细粒度的语义区分。我们将DICE作为复合嵌入的一个组成部分，并通过纳入属性和单位信息对其进行扩展，以解决一词多义问题（例如，*5*年 与*5*英里），并表示范围和高斯分布。

早期关于基于 Transformer 的模型进行数值推理的研究 [3] 包括 NumNet [56], ，它在图中表示数字，但缺乏上下文感知和类型感知能力。AeNER [75] 提出了注意力增强数值嵌入，用于对文本和表格进行数值推理。它通过数值感知嵌入和注意力机制增强 BERT，但无法区分单位，也不支持数值范围。NumBERT [80] 和 NumGPT [29] 使用科学计数法和数字原型编码，但它们在捕捉复杂结构化数据语义方面的有效性尚未得到充分探索。GenBERT [18] 是一种问答模型，使用预训练的 BERT 同时充当编码器和解码器。它采用数字的十进制表示法和逐位分词。NC-BERT [31] 在 BERT 之上添加了数值上下文掩码层，以强化与数字相关的上下文信息，从而扩展了 GenBERT。

近期研究（如 xVal [20] 和 NTL [78], ）凸显了在语言模型中明确引入数值归纳偏置的必要性。NTL 提出了一种用于数值解码的类回归损失函数，以替代标准交叉熵损失。MMD [2] 采用与 xVal 类似的方式编码数字，但引入了路由层，以明确分离文本和数字表示。然而，与我们的方法不同，这些方法没有明确纳入属性或单位，因此限制了其在结构化数值数据中的适用性。FoNE [83]通过使用余弦—正弦傅里叶特征编码数字，为大语言模型引入了一种替代性的数值嵌入策略，但它只捕捉句法结构，却忽略了属性、值和单位等语义方面。NumeroLogic [60]在数字前添加数字位数，使大语言模型具备位值意识，并在数字生成过程中引入一个思维链步骤。MultivariateGPT[45]改进了模型同时预测数字和类别的方式，而我们的工作则侧重于改进数字的嵌入方式。[63] 利用数学先验聚合数字嵌入，以改进标量大小表示；相比之下，CONE 编码的不仅是标量，还包括数值范围和高斯分布。

其他方法侧重于表格数据中的数值特征。 [21]提出了分段线性编码等方法，用于多层感知机和 Transformer 模型，但这些方法是为监督式设置设计的。相比之下，我们的方法以无监督方式学习表示，以捕捉数值表格的结构。TAPAS [25]为表格引入了排名/行/列嵌入；TUTA [73], 、TURL [13], 、TaBERT [76], 和 TABBIE [27]也采用了类似的结构编码。然而，这些模型在生成嵌入向量时，不加区分地将数字和单词一视同仁，而我们的方法则对二者区别处理。CARTE [32] 引入了一种用于表格学习的基于图的预训练模型，但没有明确建模数值语义。模式匹配系统（如 Rondo [49]）和模型管理框架 [7, 8] 通常依赖属性名称和数据值来推断匹配结果，但不会明确建模单位，也不会区分数值大小和范围。Valentine [33]提供了一个全面的基准测试，用于评估模式匹配方法；Magneto [44]则引入了一种基于嵌入检索和重排序、由大语言模型辅助的匹配系统；然而，这两者都没有对数值表示进行编码。相比之下，CONE 通过学习数值嵌入来弥补这一不足，明确编码数值大小、单位和属性上下文，从而更有效地进行列、元组或模式匹配。

据我们所知，现有的嵌入方法均无法处理数字多义性。例如，数字‘*2019*’既可以表示年份，也可以表示普通数字，而‘*7*磅’和‘*7*英里’等数值表达式则具有不同的语义解释。我们的方法通过显式拼接定义结构化数据中数值语义的各个组成部分来应对这一挑战，即用于对特定实体进行分类的属性名称、数值和单位。此外，我们引入了用于表示数值范围和高斯分布的专用嵌入。

8 结论

我们提出了 CONE，这是一种上下文感知嵌入模型，通过采用复合嵌入向量结构联合编码数值、单位和属性名称，保留数值、范围和高斯分布的语义。为此，我们设计了一种新颖的复合嵌入构建算法，该算法使用预先计算的 CONE 嵌入。在数值推理评估中，CONE 在 DROP 上取得了 87.28% 的 F1 分数，比最先进基线的 F1 分数最高高出 9.37%。此外，在多个大规模结构化数据集上进行的实验表明，CONE 在两项热门下游任务中超过了主要的最先进基线，Recall@10 最高显著提升 25%，由此证明了其在编码结构化数值数据方面的有效性。

参考文献

[1] 统计准确性。1986。标准误、置信区间及其他统计准确性度量的自助法。《统计科学》1, 1（1986），54–77。[2] Marvin Alberts、Gianmarco Gabrieli 和 Irina Espejo Morales。2024。交织文本和数字嵌入以求解数学问题。NeurIPS'24 第四届数学推理与人工智能研讨会（2024）。[3] Vaswani Ashish。2017。Attention Is All You Need。《神经信息处理系统进展》30（2017）。[4] Pierre Baldi。2012。自编码器、无监督学习与深度架构。载于 ICML 无监督与迁移学习研讨会论文集。JMLR研讨会与会议论文集，37–49。[5] Pierre Baldi 和 Kurt Hornik。1989。神经网络与主成分分析：从示例中学习且不存在局部极小值。《神经网络》2, 1（1989），53–58。[6] Taylor Berg-Kirkpatrick 和 Daniel Spokoyny。2020。上下文化数字预测的实证研究。载于 2020 年自然语言处理实证方法会议（EMNLP）论文集。4754–4764。[7] Philip A. Bernstein。2003。将模型管理应用于经典元数据问题。载于 CIDR。209–220。[8] Philip A. Bernstein 和 Sergey Melnik。2007。模型管理2.0：操控更丰富的映射。载于 SIGMOD '07（中国北京）。ACM出版社，纽约州纽约市，1 12 10 1145 1247480 1247482 –。doi：。/。[9] 2022 Danushka Bollegala。通过无监督加权拼接源嵌入学习元词嵌入。载于第三十一届国际人工智能联合会议论文集。国际人工智能联合会议组织，4058–4064。[10] Taha Ceritli 和 Christopher K I Williams。2021。识别表格数据中的计量单位。载于第21届 ECML-PKDD 数据科学自动化研讨会。[11] Jianlv Chen、Shitao Xiao、Peitian Zhang、Kun Luo、Defu Lian 和 Zheng Liu。2024。M3-embedding：通过自知识蒸馏实现多语言、多功能、多粒度文本嵌入。载于计算语言学协会 ACL 2024 论文集。2318–2335。[12] Nick Craswell。2009。平均倒数排名。《数据库系统百科全书》（2009），1703–1703。[13] Xiang Deng、Huan Sun、Alyssa Lees、You Wu 和 Cong Yu。2022。TURL：通过表示学习实现表格理解。《ACM SIGMOD记录》51, 1（2022），33–40。[14] Jacob Devlin、Ming-Wei Chang、Kenton Lee 和 Kristina Toutanova。2019。BERT：用于语言理解的深度双向变换器预训练。载于 2019 年计算语言学协会北美分会年会：人类语言技术论文集，第1卷（长短篇论文）。4171–4186。[15] Matthijs Douze、Alexandr Guzhva、Chengqi Deng、Jeff Johnson、Gergely Szilvasy、Pierre-Emmanuel Mazaré、Maria Lomeli、Lucas Hosseini 和 Hervé Jégou。2025。Faiss库。《IEEE大数据汇刊》（2025）。[16] Xingyu Du、Gongsheng Yuan、Sai Wu、Gang Chen 和 Peng Lu。2024。原位神经关系模式匹配器。载于 2024 年 IEEE 第40届国际数据工程会议（ICDE）。电气电子工程师学会，138–150。[17] Dheeru Dua、Yizhong Wang、Pradeep Dasigi、Gabriel Stanovsky、Sameer Singh 和 Matt Gardner。2019。DROP：需要对段落进行离散推理的阅读理解基准。载于 2019 年计算语言学协会北美分会年会：人类语言技术论文集，第1卷（长短篇论文）。2368–2378。[18] Mor Geva、Ankit Gupta 和 Jonathan Berant。2020。将数值推理技能注入语言模型。载于第58届计算语言学协会年会论文集。946–958。[19] Majid Ghasemi Gol、Jay Pujara 和 Pedro Szekely。2019。使用预训练单元格嵌入进行表格单元格分类。载于 2019 年 IEEE 国际数据挖掘会议（ICDM）。电气电子工程师学会，230–239。[20] Siavash Golkar、Mariel Pettee、Michael Eickenberg、Alberto Bietti、Miles Cranmer、Geraud Krawezik、Francois Lanusse、Michael McCabe、Ruben Ohana、Liam Holden Parker 等。2023。xVal：面向大语言模型的连续数字编码。载于 NeurIPS 2023科学人工智能研讨会。[21] Yury Gorishniy、Ivan Rubachev 和 Artem Babenko。2022。关于表格深度学习中数值特征的嵌入。《神经信息处理系统进展》35（2022），24991–25004。[22] Paul R Halmos。2017。有限维向量空间。Courier Dover出版社。[23] MD Rokibul Hasan 和 Janatul Ferdous。2024。人工智能与机器学习技术在混合电影推荐系统中的主导作用：应用文本转数字转换与余弦相似度方法。《计算机科学与技术研究杂志》6, 1（2024），94–102。[24] Kaiming He、Xinlei Chen、Saining Xie、Yanghao Li、Piotr Dollár 和 Ross Girshick。2022。掩码自编码器是可扩展的视觉学习器。载于 IEEE/CVF计算机视觉与模式识别会议论文集。16000–16009。[25] Jonathan Herzig、Pawel Krzysztof Nowak、Thomas Müller、Francesco Piccinno 和 Julian Eisenschlos。2020。TaPas：通过预训练进行弱监督表格解析。载于第58届计算语言学协会年会论文集。4320–4333。[26] Xinshuo Hu、Zifei Shan、Xinping Zhao、Zetian Sun、Zhenyu Liu、Dongfang Li、Shaolin Ye、Xinyuan Wei、Qian Chen、Baotian Hu、Haofen Wang、Jun Yu 和 Min Zhang。2025。KaLM-Embedding：优质训练数据带来更强大的嵌入模型。arXiv: 2501 01028 [ ] 2501 01028 。[27] 2021 cs.CL https://arxiv.org/abs/ 。Hiroshi Iida、Dung Thai、Varun Manjunatha 和 Mohit Iyyer。。TABBIE：表格数据的预训练表示。载于2021年计算语言学协会北美分会年会：人类语言技术论文集。[28] Chengyue Jiang、Zhonglin Nian、Kaihao Guo、Shanbo Chu、Yinggong Zhao、Libin Shen 和 Kewei Tu。2020。学习数字嵌入。载于计算语言学协会论文集：EMNLP 2020。2586–2599。[29] Zhihua JIN、Xin JIANG、Xiangbo WANG、Qun LIU、Yong WANG、Xiaozhe REN 和 Huamin QU。2023。NumGPT：提升生成式预训练模型的数值能力。(2023)。载于国际人工智能联合会议金融服务大语言模型国际研讨会。[30] Bhimesh Kandibedala、Gyanendra Shrestha、Anna Pyayt 和 Michael Gubanov。2025。利用对比学习，在异构结构化大规模数据集中实现可扩展的表格层次元数据分类。ICDE（2025）。[31] Jeonghwan Kim、Junmo Kang、Kyung-min Kim、Giwon Hong 和 Sung-Hyon Myaeng。2022。利用数值语境知识改进问答中的数值推理。载于计算语言学协会论文集：NAACL 2022。1811–1821。[32] Myung Jun Kim、Leo Grinsztajn 和 Gael Varoquaux。2024。CARTE：面向表格学习的预训练与迁移。载于国际机器学习会议。机器学习研究会议论文集，23843–23866。[33] Christos Koutras、George Siachamis、Andra Ionescu、Kyriakos Psarakis、Jerry Brons、Marios Fragkoulis、Christoph Lofi、Angela Bonifati 和 Asterios Katsifodimos。2021。Valentine：评估用于数据集发现的匹配技术。载于2021年IEEE第37届国际数据工程会议（ICDE）。电气电子工程师学会，468–479。[34] Jinhyuk Lee、Wonjin Yoon、Sungdong Kim、Donghyeon Kim、Sunkyu Kim、Chan Ho So 和 Jaewoo Kang。2020。BioBERT：用于生物医学文本挖掘的预训练生物医学语言表征模型。《生物信息学》36，4（2020），1234–1240。[35] Lam Hong Lee、Chin Heng Wan、Rajprasad Rajkumar 和 Dino Isa。2012。利用欧几里得距离函数增强文本分类的支持向量机分类框架。《应用智能》37（2012），80–99。[36] Joseph Lee Rodgers 和 W Alan Nicewander。1988。看待相关系数的十三种方式。《美国统计学家》42，1（1988），59–66。[37] Oliver Lehmberg、Dominique Ritze、Robert Meusel 和 Christian Bizer。2016。一个包含时间与语境元数据的大型公开网页表格语料库。载于第25届万维网国际会议伴随论文集。75–76。[38] Baoli Li 和 Liping Han。2013。用于文本分类的距离加权余弦相似度度量。载于《智能数据工程与自动化学习——IDEAL 2013：第14届国际会议，IDEAL 2013，中国合肥，2013年10月20日至23日，论文集》第14卷。Springer，611–618。[39] Bill Yuchen Lin、Seyeon Lee、Rahul Khanna 和 Xiang Ren。2020。鸟有四条腿？！NumerSense：探测预训练语言模型的数值常识知识。载于2020年自然语言处理实证方法会议（EMNLP）论文集。6862–6868。[40] Ling Liu、M Tamer Özsu 等。2009。平均精度均值。《数据库系统百科全书》1703（2009）。[41] Ninghao Liu、Qiaoyu Tan、Yuening Li、Hongxia Yang、Jingren Zhou 和 Xia Hu。2019。单个向量够用吗？探索网络嵌入中的节点多义性。载于第25届ACM SIGKDD知识发现与数据挖掘国际会议论文集。932–940。[42] Yang Liu、Zhiyuan Liu、Tat-Seng Chua 和 Maosong Sun。2015。主题词嵌入。载于人工智能促进协会会议论文集，第29卷。[43] Yinhan Liu、Myle Ott、Naman Goyal、Jingfei Du、Mandar Joshi、Danqi Chen、Omer Levy、Mike Lewis、Luke Zettlemoyer 和 Veselin Stoyanov。2019。RoBERTa：一种稳健优化的BERT预训练方法。arXiv预印本 arXiv:1907.11692（2019）。[44] Yurong Liu、Eduardo H. M. Pena、Acio Santos、Eden Wu 和 Juliana Freire。2025。Magneto：结合小型与大型语言模型进行模式匹配。《VLDB基金会会刊》18，8（2025），2681–2694。doi:10.14778/3742728。3742757[45] Andrew J Loza、Jun Yup Kim、Shangzheng Song、Yihang Liu、Joseph JY Sung、R Andrew Taylor 和 Dennis L Shung。2025。multivariateGPT：一种用于多变量类别与数值数据的仅解码器Transformer。arXiv预印本 arXiv:2505.21680（2025）。[46] Kevin Lund 和 Curt Burgess。1996。根据词汇共现生成高维语义空间。《行为研究方法、仪器与计算机》

28，2（1996），203–208。[47] E Matthew。2018。Peters、mark neumann、mohit iyyer、matt gardner、christopher clark、kenton lee、luke zettlemoyer。深度上下文化词表示。载于NAACL论文集，第5卷。[48] Sergey Melnik、Hector Garcia-Molina 和 Erhard Rahm。2002。相似度泛洪：一种通用的图匹配算法及其在模式匹配中的应用。载于第18届国际数据工程会议论文集。电气电子工程师学会，117–128。[49] Sergey Melnik、Erhard Rahm 和 Philip A. Bernstein。2003。Rondo：通用模型管理编程平台。载于SIGMOD，Alon Y. Halevy、Zachary G. Ives 和 AnHai Doan（编）。193–204。[50] Tomas Mikolov、Ilya Sutskever、Kai Chen、Greg S Corrado 和 Jeff Dean。2013。词语和短语的分布式表示及其组合性。《神经信息处理系统进展》26（2013）。[51] Aakanksha Naik、Abhilasha Ravichander、Carolyn Rose 和 Eduard Hovy。2019。探索词嵌入中的数值能力。载于第57届计算语言学协会年会论文集。3374–3380。[52] Suphakit Niwattanakul、Jatsada Singthongchai、Ekkachai Naenudorn 和 Supachanan Wanapu。2013。使用杰卡德系数计算关键词相似度。载于国际工程师与计算机科学家多学科会议论文集，第1卷。380–384。[53] Jeffrey Pennington、Richard Socher 和 Christopher D Manning。2014。GloVe：词表示全局向量。载于2014年自然语言处理实证方法会议（EMNLP）论文集。1532–1543。[54] Gabriel Peyré、Marco Cuturi 等。2019。计算最优传输：及其在数据科学中的应用。《机器学习基础与趋势》11，5 6 2019 355 607 - ( )，–。[55] Faisal Rahutomo、Teruaki Kitasuka、Masayoshi Aritsugi 等。2012。语义余弦相似度。载于第7届国际先进科学与技术学生会议（ICAST），第4卷。韩国首尔市立大学，1。[56] Qiu Ran、Yankai Lin、Peng Li、Jie Zhou 和 Zhiyuan Liu。2019。NumNet：具备数值推理能力的机器阅读理解。载于2019年自然语言处理实证方法会议与第9届国际自然语言处理联合会议（EMNLP-IJCNLP）论文集。2474–2484。[57] Putri Yuni Ristanti、Aji Prasetya Wibawa 和 Utomo Pujianto。2019。基于余弦相似度的经济学期刊标题与摘要分类。载于2019年第5届国际信息技术科学会议（ICSITech）。电气电子工程师学会，123–127。[58] Taku Sakamoto 和 Akiko Aizawa。2021。利用考虑数字定量方面的语言模型预测自然语言文本中的数字。载于《深度学习由内而外（DeeLIO）：深度学习架构知识抽取与集成第2届研讨会论文集》。140–150。[59] H Schütze、CD Manning 和 P Raghavan。2008。《信息检索导论》，剑桥大学出版社，剑桥。（2008）。[60] Eli Schwartz、Leshem Choshen、Joseph Shtok、Sivan Doveh、Leonid Karlinsky 和 Assaf Arbelle。2024。NumeroLogic：通过数字编码增强大语言模型的数值推理能力。载于2024年自然语言处理实证方法会议论文集。206–212。[61] Rico Sennrich、B Haddow 和 A Birch。2016。利用子词单元对低频词进行神经机器翻译，载于第54届计算语言学协会年会论文集。德国柏林：计算语言学协会。[62] Gyanendra Shrestha、Chutian Jiang、Sai Akula、Vivek Yannam、Anna Pyayt 和 Michael Gubanov。2025。结合双维层级元数据与嵌套的表格嵌入。载于欧洲数据库技术会议。92–105。[63] Jasivan Alex Sivakumar 和 Nafise Sadat Moosavi。2025。如何利用数字嵌入表示数字？载于第31届计算语言学国际会议论文集。7685–7697。[64] GP Spithourakis、I Augenstein 和 S Riedel。2016。用于语义纠错的数值基础语言模型。载于2016年自然语言处理实证方法会议论文集，第2016卷。计算语言学协会（ACL），987–992。[65] Georgios Spithourakis 和 Sebastian Riedel。2018。语言模型的数值能力：评估与改进其预测数字的能力。载于第56届计算语言学协会年会论文集（第1卷：长篇论文）。计算语言学协会。[66] Stanley S mith Stevens。1946。论测量尺度理论。《科学》103，2684（1946），677–680。[67] Dhanasekar Sundararaman、Shijing Si、Vivek Subramanian、Guoyin Wang、Devamanyu Hazarika 和 Lawrence Carin。2020。保留数值能力的词嵌入方法。载于2020年自然语言处理实证方法会议（EMNLP）论文集。4742–4753。[68] Avijit Thawani、Jay Pujara 和 Filip Ilievski。2021。数值能力提升语言模型的语言素养。载于2021年自然语言处理实证方法会议论文集。6960–6967。[69] Tan Thongtan 和 Tanasanee Phienthrakul。2019。使用经余弦相似度训练的文档嵌入进行情感分类。载于第57届计算语言学协会年会：学生研究论文集

研讨会. 407–414. [70] Pascal Vincent、Hugo Larochelle、Yoshua Bengio 和 Pierre-Antoine Manzagol。2008。使用去噪自编码器提取和组合鲁棒特征。载于 第*25*届国际机器学习会议论文集。1096–1103。[71] Eric Wallace、Yizhong Wang、Sujian Li、Sameer Singh 和 Matt Gardner。2019。自然语言处理模型了解数字吗？探测嵌入中的数值能力。载于 *2019*年自然语言处理实证方法会议与第*9*届国际自然语言处理联合会议论文集（*EMNLP-IJCNLP*）。5307–5315。[72] Xinyu Wang、Yong Jiang、Nguyen Bach、Tao Wang、Zhongqiang Huang、Fei Huang 和 Kewei Tu。2021。面向结构化预测的嵌入自动拼接。载于 计算语言学协会年会第*59*届会议暨第*11*届国际自然语言处理联合会议论文集（第*1*卷：长篇论文）。2643–2660。[73] Zhiruo Wang、Haoyu Dong、Ran Jia、Jia Li、Zhiyi Fu、Shi Han 和 Dongmei Zhang。2021。Tuta：用于通用结构化表格预训练的基于树的Transformer。载于 第*27*届*ACM SIGKDD*知识发现与数据挖掘会议论文集。1780–1790。[74] Hu Xu、Bing Liu、Lei Shu 和 Philip S Yu。2018。通过元学习实现终身领域词嵌入。载于 第*27*届国际人工智能联合会议论文集。4510–4516。[75] Ramil Yarullin 和 Sergei Isaev。2023。用于文本与表格推理的数值嵌入。（2023）。[76] Pengcheng Yin、Graham Neubig、Wen-tau Yih 和 Sebastian Riedel。2020。TaBERT：文本与表格数据联合理解的预训练。载于 计算语言学协会年会第*58*届会议论文集。8413–8426。[77] Wenpeng Yin 和 Hinrich Schütze。2016。学习词元嵌入。载于 计算语言学协会年会第*54*届会议论文集（第*1*卷：长篇论文）。1351–1360。[78] Jonas Zausinger、Lars Pennig、Anamarija Kozina、Sean Sdahl、Julian Sikora、Adrian Dendorfer、Timofey Kuznetsov、Momahad Hagog、Nina Wiedemann、Kacper Chlodny 等。2025。回归而非猜测：针对语言模型数字词元的类回归损失。载于国际机器学习会议。[79] Dun Zhang、Jiacheng Li、Ziyang Zeng 和 Fulong Wang。2024。Jasper 和 Stella：SOTA 嵌入模型的蒸馏。*arXiv*预印本 *arXiv:2412.19048*（2024）。[80] Xikun Zhang、Deepak Ramachandran、Ian Tenney、Yanai Elazar 和 Dan Roth。2020。语言嵌入能否捕捉尺度？载于计算语言学协会论文集：*EMNLP 2020*。4889–4896。[81] Yanzhao Zhang、Mingxin Li、Dingkun Long、Xin Zhang、Huan Lin、Baosong Yang、Pengjun Xie、An Yang、Dayiheng Liu、Junyang Lin 等。2025。Qwen3 Embedding：通过基础模型提升文本嵌入与重排序。*arXiv*预印本 *arXiv:2506.05176*（2025）。[82] Xinping Zhao、Xinshuo Hu、Zifei Shan、Shouzheng Huang、Yao Zhou、Xin Zhang、Zetian Sun、Zhenyu Liu、Dongfang Li、Xinyuan Wei、Youcheng Pan、Yang Xi-ang、Meishan Zhang、Haofen Wang、Jun Yu、Baotian Hu 和 Min Zhang。2025。KaLM-Embedding-V2：卓越的训练技术与数据造就多功能嵌入模型。arXiv: 2506 20923 [ ] 2506 20923 。[83]

cs.CL https://arxiv.org/abs/ 。Tianyi Zhou、Deqing Fu、Mahdi Soltanolkotabi、Robin Jia 和 Vatsal Sharan。2025。FoNE：通过傅里叶特征实现精确的单令牌数值嵌入。*arXiv*预印本 *arXiv:2502.09741*（2025）。