附件 1

序号：_____

编码：_____

2024 年“丁颖杯”暨“挑战杯”广东大学生课外学术科技作品竞赛校内选拔赛参赛作品

作品名称：基于知识图谱微调的跨境电商垂类模型

学院名称：数学与信息学院、软件学院

申报者姓名

(集体名称)：花生 AI

类别：

□自然科学类学术论文

□哲学社会科学类社会调查报告

□科技发明制作 A 类

☑科技发明制作 B 类

1. 概览

为了更好地理解和描述电子商务领域中的用户需求，本文设计了一种垂类模型 Peanut。如图 1 所示，Peanut 由四个主要组成部分构成：电商概念、原始概念、分类体系和商品。核心部分在于，模型的最顶层将用户需求表达为电商概念（如图 1 中的绿色框所示）。这些电商概念是简短、连贯的短语，如“夏日海滩派对”、“父亲节高端礼品”或“冬季户外运动装备”，这些电商概念精确地勾勒出用户的具体购物场景。与传统的分层类别和浏览节点不同，实际用户需求远不止于此。例如，一个计划户外烧烤的用户，他们面临的是一个情境或问题，而非单纯的产品需求。

为了深入理解这些用户需求，本文构建了一个基础语言层——原始概念层。这些“原始概念”短语通常十分简短，如“海滩”、“派对”和“防晒”（如图 1 中的蓝色框所示）为了对所有原始概念进行分类，定义了一个电商分类体系，其中不同粒度的类别通过 isA 关系形成层次结构，例如从“类别>户外活动->季节性活动->夏日海滩派对”的自左而右的路径（如图 1 中的灰色框所示）。

此外，本文在分类体系上定义了模式，用以描述不同原始概念之间的关系。例如，“适合何时”的关系可以连接“类别:服装->泳装”与“类别：场合->海滩派对”联系起来，表明某些款式的泳装特别适合在海滩派对上穿着。原始概念类似于商品的属性，如颜色或尺寸，而电商概念与商品的相关性则表明，在特定购物场景下，某些商品是必要或推荐的。例如，“防晒霜”和“沙滩伞”与电商概念“夏日海

滩派对”紧密相关，而不是仅仅与“海滩”这一原始概念相关联。Peanut 通过将用户需求表达为电商概念，并使用带有类别层次的原始概念来描述和理解用户需求。

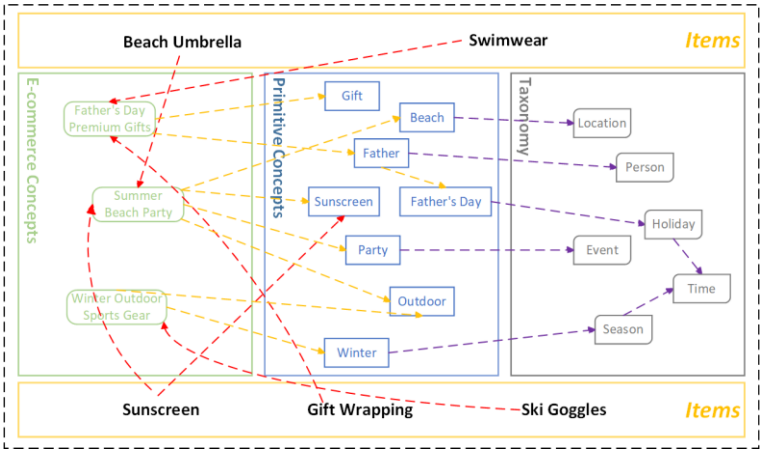

图 1 Peanut 的概览，它由四层组成：电商概念、原始概念、分类体系和商品

2. 分类体系

Peanut 的分类体系是一个预定义类别的层次结构，用于索引数百万（原始）概念。图 2 展示了分类体系的快照。在第一层中定义了 20 个类别，其中 7 个类别专门为电商设计，包括“类别”、“品牌”、“颜色”、“设计”、“功能”、“材质”、“图案”、“形状”、“气味”、“口味”和“风格”，其中最大的一个是“类别”，拥有近 800 个叶类，因为商品的分类几乎是每个电商平台的主干。其他类别如“时间”和“地点”更接近通用领域。值得一提的特殊类别是“IP”（知识产权），其中包含了数以百万计的真实世界实体，如著名人物、电影和歌曲。在 Peanut 中，实体也被视为原始概念。分类

体系第一层中定义的 20 个类别也称为“领域”。

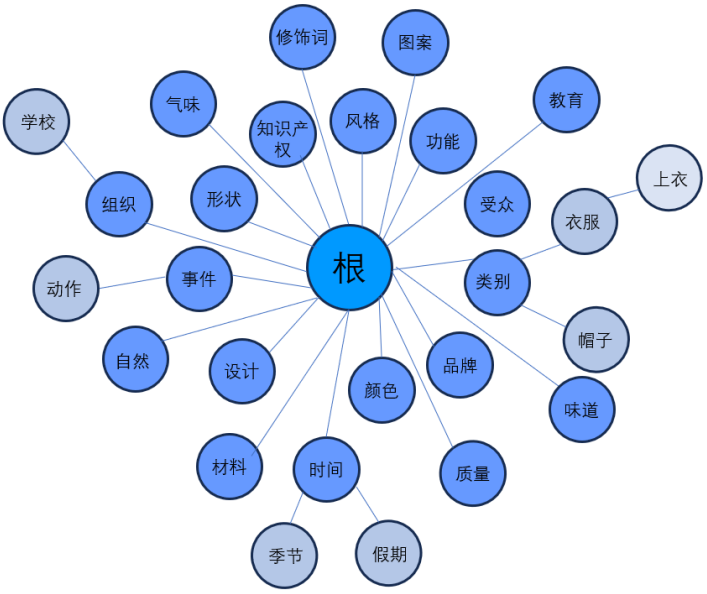

图 2　Peanut 的分类体系概览

这个层次结构的分类体系能够以有组织的方式管理和扩展概念网络。通过这种方式，可以更系统地描述用户需求，并将其与相应的商品关联起来。分类体系的构建对于整个知识图谱至关重要，是电商概念和原子概念的组织提供了基础。

3. 原始概念

原始概念是构成电商知识图谱的基础元素，它们是简短且具有明确意义的词汇，用于描述商品属性、用户需求和其他电商相关的实体。这些概念对于理解用户的购物需求和将这些需求与适当的商品关联起来至关重要。通过这些原始概念，Peanut 能够更好地理解和响应用户的购物需求，从而提升整个电商平台的智能化水平。本文将通过词

汇挖掘和超类发现的方法来构建一个全面的原始概念层。

3.1 词汇挖掘

在定义了分类体系之后，本文通过两种方式来扩大原始概念的规模。第一种是通过本体匹配从多个来源的结构化或半结构化知识库中整合现有知识。第二种方式是从电商领域的大规模文本语料库中挖掘新概念，例如搜索查询、产品标题、用户撰写的评论和购物指南。挖掘特定类别的新概念是一种序列标注任务，输入是一系列单词，输出是一系列预定义的标签。

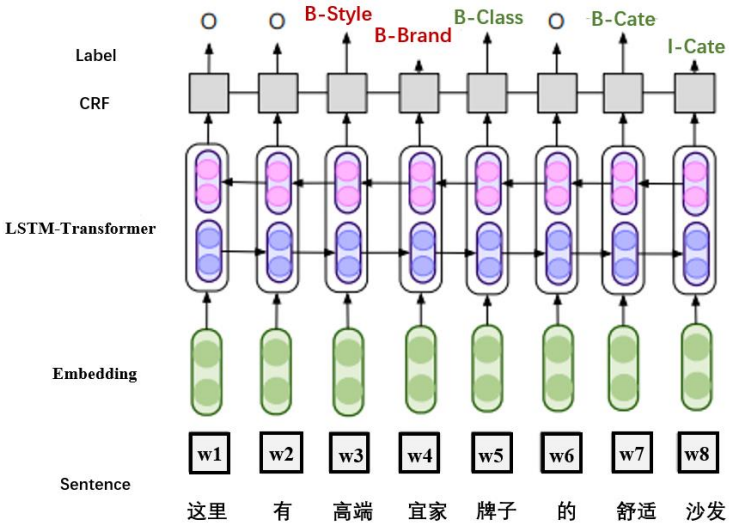

图 3　LSTM-Transformer-CRF 模型的基本原理架构

图 3 展示了 LSTM-Transformer-CRF 模型的原理架构，模型用于各种序列标注任务。LSTM-Transformer-CRF 模型由 LSTM-Transformer 层和 CRF 层组成。其中 LSTM-Transformer 层使隐藏状态能够捕获单词的历史和未来上下文信息，负责处理序列数据，捕捉

长期依赖关系。而 CRF 层则考虑当前概念与潜在类别之间的相关性，理解概念间的复杂关系，确保模型能够准确地识别和分类新出现的概念。通过这种结构，模型能够有效地从大量文本中识别出关键概念，并将它们分类到相应的类别中，从而实现自动化的知识发现和分类。然后，识别结果由人工检查以确保其正确性。

一旦确定了类别，原始概念在文本中的实际用词就成为了新的原始概念，每个概念都将被分配一个唯一的 ID。例如，在讨论“户外烧烤”时，“户外烧烤”可能对应多个不同的原始概念，因为“户外”和“烧烤”这两个词可能有多种不同的含义。一旦确定了“户外烧烤”这个短语中的“户外”和“烧烤”分别属于哪个类别（比如“地点”和“动作”），那么“户外烧烤”这个短语就与一个具体的、有明确含义的原始概念对应起来了。可以有多个具有相同名称但不同 ID（含义）的原始概念，使 Peanut 能够消除原始文本中的歧义。

3.2 超类发现

本文为 20 个顶层类别的原始概念分类到每个领域内的细粒度类别，这个步骤为超类发现，其中需要预测任意一对原始概念之间的上下位关系。本文采用了两种方法的结合：一种基于模式的无监督方法和一种基于投影学习的监督模型。

3.2.1 基于模式的方法。基于模式的超类发现方法定义了特定的文本模式，如“Y such as X”，以从语料库中挖掘上下位词对。这种方法

已知的缺点是召回率低，因为它假设上下位词对在这些模式之一中共现，但在实际的语料库中，这种共现并不总是成立。除了这些模式，本文还采用了其他规则，直接使用中文的一些特殊语法特征来发现超类，例如“XX 裤（XX pants）”一定是“裤（pants）”等。这种方法利用了语言的结构性特点，使得超类发现更加灵活和准确。 通过结合这些规则，基于模式的方法能够更有效地挖掘出上下位词对，从而丰富知识图谱的内容和结构。

3.2.2 投影学习。投影学习的一般思想是学习一个函数，该函数以可能的下位词p和候选超类词h的词嵌入作为输入，并输出p和h之间存在超类关系的可能性。要为给定的下位词p发现超类，首先将这个决策函数应用于所有候选超类，并选择最有可能的候选词，具体步骤如下：

对于给定的下位词 p 和候选超类词 h ，首先通过查找表获取它们的词嵌入。这些嵌入是在电商语料库上训练的，能够捕捉到词汇的语义信息。然后使用投影张量T来衡量p和h之间存在超类关系的可能性。在投影张量T的第k层，计算得分s^k：

$$s^k = p^T T^k h$$

其中 T^k 是矩阵，$k \in [1, \mathrm{K}]$。结合K个得分，得到相似性向量s。经过全连接层和 sigmoid 激活函数后，得到用以衡量下位词 p 和候选超类词 h之间存在超类关系的可能性y：

$$y = \sigma(Ws + b)$$

3.2.3主动学习。由于为每个领域标记大量的上下位词对不易扩展，因此本文采用了主动学习作为一种更经济的方法来选择要标记的示例，以便减少模型训练的标注成本。主动学习算法允许模型选择最有益的数据点，并从注释器那里查询这些数据点的标签，模型能够专注于那些最具挑战性或最具信息量的样本，从而提高学习效率。通过采用一种不确定性和高置信度采样策略（UCS），以选择能够有效地改进模型的样本。迭代的主动学习算法如下所示：

UCS 主动学习算法

输入：未标记的数据集D，测试数据集 T，评分函数$f(\cdot,\cdot)$，人工标注 H，每次迭代中人工标注样本的数量 K；输出：评分函数$\hat{f}(\cdot,\cdot)$，预测分数S。

procedure AL(D, D_0, T, f, H, K)
  $i \leftarrow 0$
  $D_0 \leftarrow random(D, K)$
  $L_0 \leftarrow H(D_0)$
  $D \leftarrow D - D_0$
  $\hat{f}, fs \leftarrow train\_test(f, L_0, T)$
  $S \leftarrow \hat{f}(D)$
  repeat
    $p_i = \frac{|S_i - 0.5|}{0.5}$
    $D_{i+1} \leftarrow D(Top(p_i, \alpha K)) \cup D(Bottom(p_i, (1-\alpha)K))$
    $L_{i+1} \leftarrow H(D_{i+1}) \cup L_i$
    $D \leftarrow D - D_0$
    $\hat{f}, fs \leftarrow train\_test(f, L_{i+1}, T)$
    $S \leftarrow \hat{f}(D)$
  until fs 在连续 n 轮中没有显著提升
end procedure

在主动学习的过程中，如算法的第 3 行到第 7 行所示，首先从整个未标记的数据集D中随机选择一个大小为K的样本子集D_0。这里的K是希望从数据集中选取的样本数量。选择这些样本的目的是为了进行人工标注，从而为模型提供初始的训练数据。得到了初始标记的数据集L_0，并且D_0从D中被移除后。进而，使用L_0训练投影学习模型f，并在测试数据集 T上测试性能。fs是 T 上的性能指标。

最后，利用训练好的模型 $\hat{f}$对剩余的未标记数据集 D进行预测，并计算出每个样本的得分S_0。这个得分反映了模型对未标记样本的分类置信度，为后续的主动学习步骤提供了依据。通过这种不断迭代的方式，优化模型的性能。

接下来，不断地寻找那些尚未被标记的数据样本，并选择其中一些进行人工标记。这些新标记的样本随后被用来进一步训练和改进模型。为了高效地选择这些样本，本文采用了不确定性和高置信度采样（UCS）的策略。这种策略基于两个关键因素：

1）不确定性采样：

算法寻找那些模型预测得分接近 0.5 的样本，这意味着模型对这些样本的分类不是很确定。这种情况下，通过人工标记来获取真实标签，可以帮助模型学习并提高其预测能力。

在算法的第 9 行中，通过计算 $\frac{|S_i-0.5|}{0.5}$ 来评估样本的不确定性，以此确定哪些样本最需要人工标记。

2) 高置信度采样：

除了不确定性样本，还要关注那些模型以高置信度预测为正类的样本。特别是在处理相似关系时，模型可能会错误地将一些负样本以高置信度预测为正样本。通过人工标记这些高置信度的样本，可以及时纠正模型的错误预测，从而提高模型的准确性。算法的第 10 行选择了那些模型预测得分高的样本进行人工标记。

随着算法的不断迭代，逐渐构建了一个越来越大的人工标记数据集，这些数据被用来训练模型，使其性能逐渐提升。这个迭代过程会一直持续，直到在连续 n 轮中模型的性能没有显著提升为止。这样，通过只标记那些对模型改进最有帮助的样本，可以获得了一个性能更好的模型，且有效地减少了人工标记的成本。

4. 电商概念层

在电商概念层，每个节点代表一种特定的购物场景，可以用至少一个原始概念来解释。接下来，本文先使用几个例子介绍一个好的电商概念的标准，然后展示如何生成电商概念，并介绍如何将电商概念链接到原始概念层。

4.1 标准

用户需求在 Peanut 中被概念化为电商概念。通常，好的电商概念需要满足如表 1 的标准：

表 1 电商概念的标准

标准	描述	反例	正例
电商意义	应该让任何人轻易地想到电商平台上的一些商品，代表一个特定的购物需求	“蓝天”、“母鸡下蛋”	“运动鞋”、“电子书阅读器”
连贯性	应该是一个连贯的短语	“圣诞礼物给爷爷”、“为孩子们保暖”	“给爷爷的圣诞礼物”、“为孩子们保暖”
可信度	根据常识知识，它应该是一个可信的短语	“性感婴儿裙”、“欧式韩风窗帘”	“婴儿连体衣”、“欧式窗帘”
清晰度	电商概念的含义应该清晰且易于理解	“儿童和婴儿的辅食”	“婴儿辅食”、“儿童营养餐”
正确性	应该没有发音或语法错误	“玩儿童游戏”（可能存在语法错误）	“儿童游戏”

4.2 生成

为了识别合适的概念短语，本文提出了一种两阶段的框架：候选生成和概念识别。

4.2.1 候选生成。生成概念候选有两种不同的方法。第一种是从文本中挖掘原始概念，可以使用 AutoPhrase 从电商领域的大规模语料库中挖掘可能的概念短语，包括搜索查询、产品标题、用户撰写的评论和商家编写的购物指南。另一种方法是使用现有的原始概念生成新的候选。因为 Peanut 的目标是覆盖尽可能多的用户需求，结合不同类别的原始概念生成新的候选概念，例如，“位置：室内”与“事件：烧烤”结合成“室内烧烤”。组合不同类别的原始概念的过程是通过自动化技术来快速识别潜在概念，然后通过人工审查和精细化处理来

确保概念的准确性和正确性。本文使用模式“[类别：功能] [类别：物品] 用于 [类别：事件]”生成一个具体概念。例如，功能是“保暖”，物品为“帽子”，使用事件为“旅行”，将这三个元素结合起来，就得到了概念“旅行保暖帽”，它描述了一个专为旅行设计的、具有保暖功能的帽子。表 1 也显示了一些在实践中使用的模式和相应的电商概念，包括一些在后续步骤中等待过滤的反例概念。

4.2.2 分类。为了自动评估一个候选概念是否适合作为电商概念，主要难于评估其可信度。本文采用了一个基于 Wide&Deep 框架的知识增强深度分类模型。该模型接收候选概念c作为输入，并输出一个评分，该评分反映了c成为优质电商概念的可能性。模型的深度侧主要包含两个组件。首先，字符级 LSTM 通过输入字符级别的嵌入序列$\{ch_1, ch_2, \ldots ch_n\}$，编码候选概念$c$，并通过平均池化得到概念嵌入$c_1$。另一个组件是知识增强模块，利用预训练的词嵌入、词性（POS）标签嵌入和命名实体识别（NER）标签嵌入来编码概念内每个单词之间的相互影响。将这三个嵌入连接后，获得候选概念c的输入嵌入序列$\{w_1, w_2, \ldots w_n\}$(m < n)。经过 LSTM 处理后，使用自注意力机制进一步编码单词之间的相互影响，得到序列输出$\{w_1', w_2', \ldots w_m'\}$。接着，引入外部知识，即每个单词$w$的维基百科摘要，并使用 Doc2Vec 编码得到外部知识表示$\{k_1', k_2', \ldots k_m'\}$。将$w_i'$和$k_i'$连接后，通过最大池化得到最终的知识增强表示$c_2$。宽侧采用预先计算的特征，如字符数、单词数，通过 BERT 模型计算候选概念的困惑度和每个单词在电商场景

中的流行度。这些特征经过全连接层处理，得到宽特征表示c_3。最终分数$\hat{y}_c$通过连接c_1、c_2和c_3，并经过 MLP 层计算得出。模型参数通过点式学习和负对数似然目标函数进行学习：

$$L = -\sum_{(c)\in D+} log\ \hat{y}_c + \sum_{(c)\in D-} log\,(1-\hat{y}_c)$$

其中D^+和D^-分别是好的和坏的电商概念。为了严格控制质量，随机抽取每个输出批次的一小部分，并让人工进行手动注释。当准确率达到一定阈值时，整个批次的概念才会被添加到 Peanut 中。此外，注释的样本将被添加到训练数据中，以迭代地提高模型性能。

4.3 理解

在 Peanut 系统中，从文本语料库中提取电商概念，这些概念作为独立的短语，需要与原始概念层关联以深化对用户需求的理解，这一过程被称为“电商概念标注”。例如，对于“户外烧烤”，需要识别“户外”为位置，而“烧烤”为事件，尽管“烧烤”也可能指代一部电影，从而归类为“IP”。由于电商概念短语通常很短，缺乏上下文，这使得标注任务比标准的命名实体识别更加困难。为了解决这个问题，本文提出了一个文本增强的深度 NER 模型，带有模糊 CRF，如图 4 所示。这项任务的输入是经过中文分词后的概念词序列$\{w_1, w_2, \ldots w_m\}$，而输出是相同长度的序列$\{y_1, y_2, \ldots y_m\}$，表示每个单词的类别标签，使用（I/O/B）方案。模型由两个组件组成：文本增强概念编码器和模糊 CRF 层。

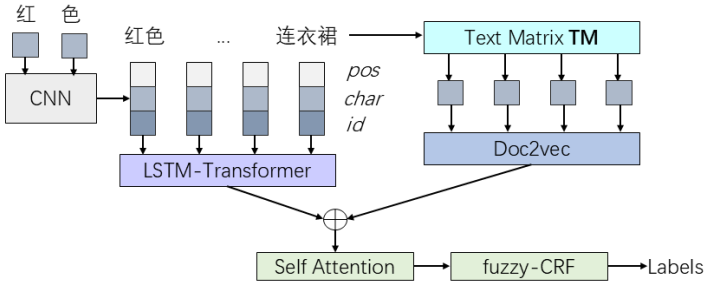

图 4 知识增强型的深度电商概念分类模型概览

4.3.1 文本增强概念编码器。为了在表示层中充分利用信息特征，本文设计了文本增强概念编码器，结合了单词级、字符级和位置特征。为字符词汇表c中的每个字符初始化一个嵌入向量，将单词w_i表示为字符向量序列$\{c_1^i, c_2^i, \ldots c_t^i\}$，其中$c_t^i$代表单词$w_i$中第$j$个字符的向量，$t$为单词长度。本文采用卷积神经网络（CNNs）来提取每个单词w_i的字符级特征。具体来说，使用窗口大小为k的卷积层来捕获每个字符及其邻近字符的信息，然后通过最大池化操作来获得每个字符的最终表示：

$$c_j^i = CNN([c_{j-k/2}^i, \ldots, c_j^i, \ldots, c_{j+k/2}^i]).$$

$$c_i = MaxPooling([c_0^i, \ldots c_j^i, \ldots]).$$

为了捕获单词级特征，使用预训练的 GloVe 词嵌入将单词映射到实值向量x_i，这将作为单词的初始特征，并在训练过程中进行微调。本文还计算了词性标注特征p_i。最终，通过连接这三个嵌入来获得单词的表示w_i：

$$w_i = [x_i; c_i; p_i].$$

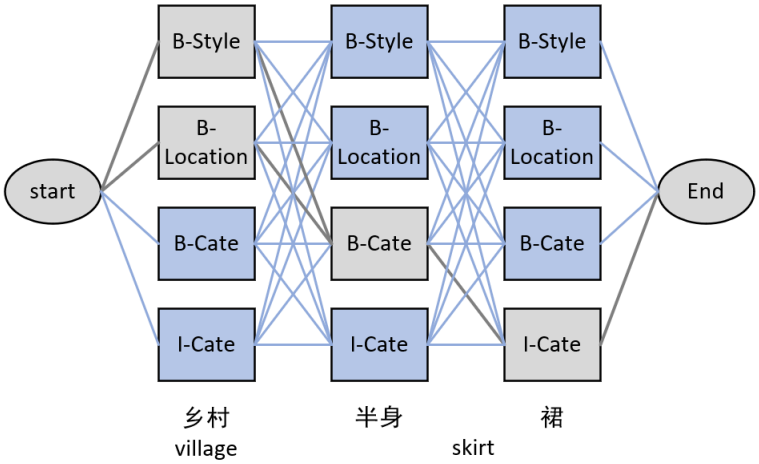

图 5 模糊 CRF 层的一个示例

类似于分类模型，利用 LSTM 处理词序列，得到隐藏嵌入$\{h_1, h_2, \ldots h_m\}$。为了增强模型，本文构建了文本嵌入矩阵TM，通过 Doc2vec 编码大规模语料库中的上下文信息。对于每个单词w_i，从TM中获取其文本增强嵌入tm_i，并将其与对应的隐藏嵌入h_i连接。然后，应用自注意力层来调整每个单词的表示，考虑其周围的增强文本嵌入，以获得更好的特征表示：

$$h_i' = SelfAtt([h_i; tm_i]).$$

4.3.2 模糊 CRF 层。在概念编码模块之后，将嵌入输入到 CRF 层。与传统 CRF 不同，本文采用模糊 CRF 来更有效地处理消歧问题，因为每个单词的有效类标签不是唯一的，并且由于概念太短，这种现象更加严重。图 5 展示了一个例子，其中单词“乡村”在电商概念“乡村半身裙”中可以链接到原始概念“空间：乡村”或“风格：乡村”。这两个标签都讲得通。因此，调整最终概率如下：

$$L(y|X) = \frac{\sum_{\hat{y} \in Y_{possible}} e^{s(X,\hat{y})}}{\sum_{\hat{y} \in Y_X} e^{s(X,\hat{y})}}.$$

其中Y_X表示序列X的所有可能标签序列，$Y_{possible}$包含所有可能的标签序列。

5. 商品关联

在构建电商知识图谱的过程中，商品是核心的实体，因为电商平台的核心目标是帮助用户快速、准确地找到他们想要的商品。本文已经通过电商概念和原始概念来理解和表达用户的需求，现在需要做的是将这些概念与实际的商品连接起来，以形成完整的知识图谱。

原始概念与商品之间的关联相对简单，因为它们通常对应于商品的直接属性，比如颜色、尺寸或者品牌。这些属性就像是商品的标签，可以直接与商品匹配。然而，电商概念与商品之间的关联更加复杂。电商概念代表了用户的购物场景或需求，比如“户外烧烤”或“儿童保暖”。这些概念包含了更丰富的语义信息，并且可能涉及到多个原始概念和商品。例如，“户外烧烤”可能涉及到烤架、木炭、食材等多种商品，而这些商品之间可能没有直接的属性关联，但它们共同满足了“户外烧烤”这一场景的需求。但由于“语义漂移”现象，某些商品虽然与特定的电商概念相关，但并不是因为它们直接具有相关的原始概念属性。以“户外烧烤”为例，木炭是户外烧烤时必需的商品，

但它与“户外”这一原始概念并无直接联系。因此，需要一种方法来识别这种间接的、基于场景的关联。

为了解决这个问题，本文设计了一个知识感知的深度语义匹配模型，通过分析电商概念和商品描述之间的文本信息，来确定它们之间是否存在关联。这种方法不仅考虑了商品的直接属性，还考虑了商品如何适应特定的购物场景。通过这种方式，可以确保用户在搜索特定的电商概念时，能够找到所有相关的商品，从而提供更加丰富和个性化的购物体验。这个模型的输入包括电商概念词序列和候选商品标题的词序列。首先，将这两组词序列的预训练词嵌入、词性（POS）标签嵌入和命名实体识别（NER）标签嵌入进行拼接，以获得输入嵌入，分别表示为序列 $\{w1, w2, \ldots, wm\}$ 和$\{t1, t2, \ldots, tl\}$。

这里，每个词都会被转换成一个多维的向量，其中预训练词嵌入捕捉了词的语义信息，POS 标签嵌入捕捉了词的语法角色，而 NER 标签嵌入则捕捉了词所代表的实体类型。接下来，使用宽卷积神经网络（CNNs）分别对电商概念和商品标题进行编码。

$$w_i' = CNN([w_{i-k/2}, \ldots, w_i, \ldots, w_{i+k/2}])$$

$$t_i' = CNN([t_{i-k/2}, \ldots, t_i, \ldots, t_{i+k/2}])$$

宽 CNNs 能够捕捉局部的语义模式，并且由于其局部感受野，可以有效地处理电商概念和商品标题中词汇的排列和组合。

在进行电商概念与商品之间的语义匹配时，模型要能够识别出概念中的哪些单词对匹配特定商品更为重要。同样，商品标题中的哪些单词对匹配特定概念更为关键。由于每个单词对整体匹配的贡献可能

不同，本文引入了注意力机制来为这些单词分配不同的权重。在模型中，使用一个注意力矩阵来同时模拟概念词与商品标题词之间的双向交互。这个矩阵的每个元素代表了一对单词（概念词和标题词）之间的匹配程度或相关性。具体来说，注意力矩阵的值可以通过以下方式定义：

$$att_{i,j} = v^T tanh\,(W_1 w_i' + W_2 t_j')$$

其中$i \in [1, \mathrm{m}]$和 $j \in [1, \mathrm{l}]$，v，W_1和W_2是参数。然后，可以计算出每个概念词w_i和标题词t_i的权重：

$$\alpha_{w_i} = \frac{exp\,(\sum_{j=1}^{l} a\,tt_{i,j})}{\sum_{i=1}^{m} exp\,(\sum_{j=1}^{l} a\,tt_{i,j})}$$

$$\alpha_{t_j} = \frac{exp\,(\sum_{i=1}^{m} a\,tt_{i,j})}{\sum_{j=1}^{l} exp\,(\sum_{i=1}^{m} a\,tt_{i,j})}$$

然后，获得概念嵌入c：

$$c = \sum_i \alpha_{w_i} w_i'$$

以及商品嵌入i类似。为了引入更多有助于语义匹配的信息知识，获得知识嵌入序列：

$$k_i = Doc2vec(Gloss(w_i))$$

此外，得到了与当前电商概念相关联的第j个原始概念的类别标签 id 嵌入cls_j。因此，概念侧有三个序列：

$$\{k_{w_i}\} = \{kw_1, kw_2, \ldots kw_{2*m+m'}$$

$$= \{w_1, w_2, \ldots w_m, k_1, k_2, \ldots km, cls_1, cls_2, \ldots cls_{m'}\}$$

其中m'是原始概念的数量。在商品侧，本文直接使用词嵌入序列{ti} = {t1, t2, … tl}。然后，采用匹配金字塔的思想，第k层的匹配矩阵值定义如下：

$$match_{i,j}{}^{k} = kw_i{}^{T} W_k t_j$$

其中$i \in [1, 2 * \mathrm{m} + \mathrm{m}']$ 和 $j \in [1, \mathrm{l}]$。每个匹配矩阵层然后被输入到 2 层 CNNs 和最大池化操作中，以获得匹配嵌入ci^k。最终的匹配金字塔嵌入ci通过以下方式获得：

$$ci = MLP([; ci^k;])$$

最终得分衡量概率的计算如下：

$$score = MLP([c; i; ci])$$

6. 结果评估

为了全面评估 Peanut 模型的性能，本文将其与其他几种先进的模型进行了比较，包括生成型模型和序列模型。这些模型在三个不同的任务上进行了测试：产品对齐（Product Align）、评论主题分类（Review Topic Classify）和产品选择（Product Select），测试结果如表 2 所示。

表 2 在任务上的表现

Models	Product Align	Review Topic Classify	Product Select
Generative model	**Micro F1**	**Macro F1**	**Rouge score**
LLaMA-2-7B	42.73	50.39	65.14
GPT3 175B	53.60	63.22	77.39
Sequence model	**Micro F1**	**Macro F1**	**Rouge score**
BiLSTM	56.25	61.74	76.39
Peanut	**57.90**	**65.22**	**83.60**

从表中可以看出，Peanut 模型在所有三个任务上都取得了最好的性能。在产品对齐任务中，Peanut 模型的 Micro F1 得分为 57.90，高于其他模型。这表明 Peanut 模型在理解和匹配产品方面具有较高的准确性。在评论主题分类任务中，Peanut 模型同样表现优异，其 Macro F1 得分为 65.22，超过了其他所有模型。这表明 Peanut 模型在理解和分类用户评论方面具有较好的性能。在产品选择任务中，Peanut 模型的 Rouge 得分为 83.60，也是所有模型中最高的，这显示了 Peanut 模型在生成相关产品推荐方面的高效率。结果证明了 Peanut 模型在处理电子商务领域的复杂任务时的有效性和优越性。Peanut 模型结合了深度学习网络和知识图谱，能够深入理解用户需求，并将其与适当的商品关联起来，从而提供更准确的购物服务，进而有效地提高用户体验和提高转化率。