基于多模态特征融合的图像文本检索

摘要

随着互联网与社交媒体的迅猛发展，全球范围内产生的跨模态数据呈现出前所未有的增长态势。其中，图像与文本作为两大主要信息载体，因其直观易懂与高度抽象的特性，成为人们日常交流、知识传播与信息检索的核心组成部分。本文正是聚焦于这一关键需求，深入探讨如何构建一个功能强大且性能优越的双向跨模态检索系统，即既能基于文本内容精确检索相关图像，也能依据特定图像快速找到对应的描述文本。针对问题一和问题二：本文借助预训练的**Chinese-CLIP模型**，旨在整合文本与图像数据，利用先进深度学习模型**BERT**与ViT-B/16提取双模态特征。首先，预训练BERT捕获文本深度语义，**ViT-B/16**提取图像高级视觉特征，两者共同提供高质量多维向量，为后续跨模态交互与检索打下坚实基础。接着，运用**KNN**算法，通过计算查询与数据库样本间余弦相似度，确定最相似的K个实例，以此反映跨模态数据语义匹配程度。评价模型性能时，主要采用Recall指标，衡量检索结果中正确关联样本占比，直观反映模型在实际应用中的检索精度。

模型训练阶段，采用**InfoNCE损失函数**，嵌入对比学习框架，强化文本与图像间关联学习。InfoNCE通过最大化正样本对相似度并最小化负样本对相似度，促使模型在特征空间中拉拢正样本、分离负样本，从而增强跨模态检索准确性。 在优化过程中，选择**AdamW算法**，结合动量项与自适应学习率调整，有效收敛模型参数，提升训练效率，同时借助权重衰减抑制过拟合，增强模型泛化能力。

综上所述，本文融合BERT与ViT-B/16模型提取双模态特征，运用KNN算法进行检索，以Recall指标评估性能。模型训练中采用InfoNCE损失函数与AdamW优化算法，有效提升检索准确性与泛化能力。

关键词：双向跨模态检索；**Chinese-CLIP；KNN**算法；

**目录**

[一、 引言 3](#_Toc16480)

[1.1 研究背景 3](#_Toc29195)

[1.2 研究目标 3](#_Toc4230)

[1.3 研究思路 4](#_Toc76)

[二、 数据探索与数据预处理 5](#_Toc24134)

[2.1 数据探索 5](#_Toc20392)

[2.2 技术探索 6](#_Toc24580)

[2.2.1 CLIP 7](#_Toc11798)

[2.2.2 ViT 9](#_Toc93)

[2.2.3 RoBERTa 10](#_Toc26172)

[2.3 文本处理 12](#_Toc10229)

[2.3.1 数据清洗 12](#_Toc11386)

[2.3.2 异常值处理 12](#_Toc5835)

[2.4 图片处理 13](#_Toc12652)

[2.4.1 异常值处理 13](#_Toc20797)

[2.4.2 尺寸调整 15](#_Toc28001)

[2.4.3 数据增强 15](#_Toc1568)

[2.4.4 图片编码与序列化 16](#_Toc5268)

[2.4.5 划分训练集 17](#_Toc16019)

[三、 模型构建 18](#_Toc25135)

[3.1 模型选择 18](#_Toc28029)

[3.1.1 Chinese-CLIP模型 18](#_Toc6238)

[3.1.2 Chinese-CLIP的优势 19](#_Toc25549)

[3.1.3 Chinese-CLIP模型规模与配置 19](#_Toc17913)

[3.2 特征提取 20](#_Toc6139)

[3.2.1 文本特征 20](#_Toc9876)

[3.2.2 图像特征 22](#_Toc25182)

[3.3 定义损失函数和优化器 23](#_Toc22515)

[3.3.1 模型中损失函数选择 23](#_Toc7498)

[3.3.2 模型中优化器的选择 24](#_Toc14386)

[3.4 微调模型 24](#_Toc20253)

[四、 模型评估 26](#_Toc6695)

[4.1 相似度与检索和排序 26](#_Toc30105)

[4.1.1 KNN检索分析 26](#_Toc10688)

[4.1.2 ANN搜索分析 27](#_Toc1352)

[4.1.3 采用KNN检索与余弦相似度 28](#_Toc26733)

[4.2 Recall计算与分析 28](#_Toc1206)

[4.2.1 Recall公式说明 28](#_Toc10718)

[4.2.2 Recall的计算 29](#_Toc10952)

[4.2.3 Recall结果分析 30](#_Toc31639)

[五、 效果展示 32](#_Toc24695)

[5.1 针对问题一展示 32](#_Toc30241)

[5.2 针对问题二展示 33](#_Toc18870)

[六、 研究总结 34](#_Toc11500)

[6.1 回顾项目 34](#_Toc22104)

[6.2 研究的不足 35](#_Toc23069)

[参考文献 36](#_Toc8991)

# 

# 引言

## 研究背景

在当前信息化社会进程中，随着智能终端设备普及率的不断提升以及多媒体社交网络平台的爆炸式增长，多模态数据（尤其是文本和图像）呈现出前所未有的指数级积累态势。这一发展趋势不仅塑造了信息高度密集的网络生态环境，同时也对信息管理和检索技术标准提出了更为先进和严谨的挑战。用户在面对海量数据洪流时，对能够跨越模态边界，实现在文本、图像等多种类型数据间无缝衔接及高效检索的需求日益迫切。

不同模态数据内在的语义关联并非直观显现，其间显著的语义鸿沟和分布不均匀性构成了大数据时代下深度解析和充分利用多模态信息[[[1]](#endnote-0)]的核心难题。文本与图像作为信息传播过程中的两大关键载体，各自蕴含独特而丰富的语义内容；突破视觉与语言之间固有的认知屏障，构建两者间精准理解和有效映射的理论和技术框架，已成为当今信息技术领域亟需攻关的重要课题。

## 研究目标

本项目旨在运用提供的附件1数据集，采用针对性适用于图像特性的有效特征提取技术，对图像关键特征进行精准捕获，并与相应文本特征深度集成，构建一个高性能、多模态融合的图像检索模型。

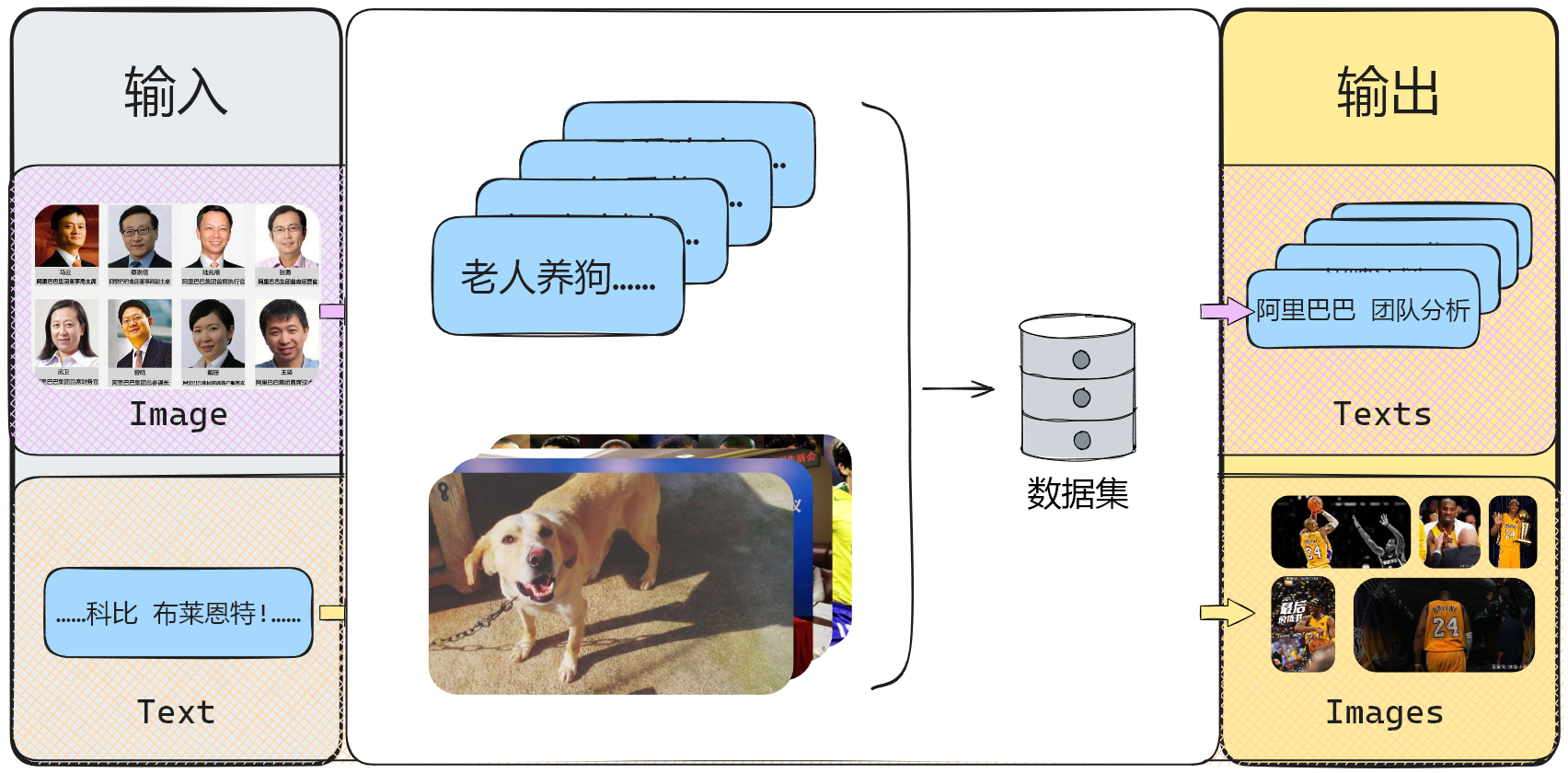


图 1实现多模态特征融合的图像文本检索

该模型的核心目标是依据附件2中“word\_test.csv”文本描述信息，对附件2内“ImageData”文件夹所包含的图像进行精准匹配与高效检索，旨在精确识别并呈现与查询文本语义最贴合的前五张图像。

对于问题二，处理重点在于创新设计并严谨实施一种基于文本驱动的多模态特征融合模型及其配套算法。具体而言，本文将利用附件3中“image\_test.csv”提供的图像ID作为线索，深入挖掘这些图像所蕴含的内在特性，以此为依据，在附件3的大型文本数据库“word\_data.csv”中精准筛选出与给定图像内容关联性最强的前五条文本记录。这一环节的核心理念在于打破单模态信息孤岛，通过深度融合图像与文本的多元异构特征，构建起从图像到文本的高效率、低误差跨模态检索桥梁。最终，所有检索结果将按照相似度严谨排序，并统一汇编入“result2.csv”文件中，以此全面评估模型在跨模态检索任务，特别是从图像出发快速定位高度相关文本的应用性能与实效性，为进一步提升跨媒体信息检索系统的智能化水平提供有力支撑。

## 研究思路

针对上述目标的初步分析，本文拟定了大概的思维导图 如图 2所示：

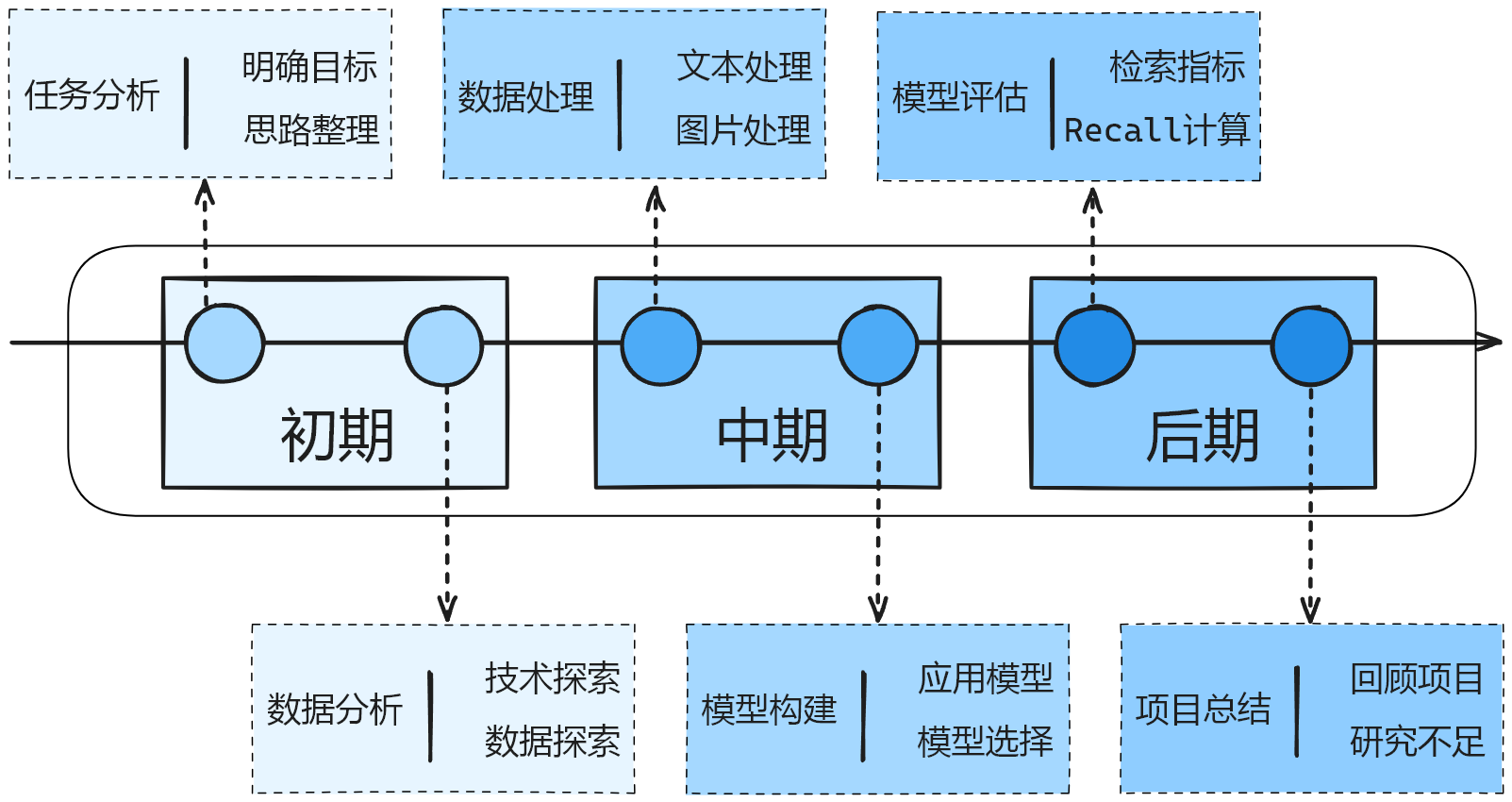


图 2研究思路图

# 数据探索与数据预处理

## 数据探索

在本文中，采用了包含四个核心附件的数据集，对这些附件进行了详尽的数据探索，旨在全面揭示数据集的结构、内容特征，以为后续的模型构建和数据分析提供坚实的基础。

附件1包括“ImageData”文件夹和名为“ImageWordData.csv”的CSV文件。其中，“ImageData”文件夹收纳了五万张图像，用作图像文本检索任务的训练和验证数据源；而“ImageWordData.csv”记录了相关图像的文本信息，如：image\_id作为每张图像独一无二的标识符及图像文件名，用于图像数据的定位和索引；caption则对应每张图像详细的文本描述。以下为示例表 1：

表 1 ImageWordData.csv表格

|  |  |
| --- | --- |
| **image\_id** | **caption** |
| Image14001001-0000.jpg | 《绿色北京》摄影大赛胡子<人名>作品 |
| Image14001001-0001.jpg | 只看阵容对手就已经吓尿了,巴西队黄金一代神挡杀神,佛挡杀佛! |
| …… | …… |

附件2提供了“word\_test.csv”、“image\_data.csv”两个CSV文件以及另一个“ImageData”文件夹。其中，“word\_test.csv”作为图像检索任务（1）的测试集，包含text\_id（唯一文本标识符）和caption（待检索文本内容）。同时，“image\_data.csv”记录了“ImageData”文件夹内图像的image\_id，以便与“word\_test.csv”中的文本信息进行配对，其内的图像文件名与对应的image\_id保持一致。

表 2 word\_test.csv 表格

|  |  |
| --- | --- |
| **text\_id** | **caption** |
| Word-1000004254 | 后来美国历史学家及情报部高官说:金无怠的的间谍活动是导致韩战延迟 |
| Word-1000030077 | 茶主题商业综合体的未来当下,如果专业市场只是安于做一个收商铺租赁 |
| …… | …… |

附件3的配置与附件2相似，包含“word\_data.csv”、“image\_test.csv”以及“ImageData”文件夹，分别服务于文本检索任务（2）。其中，“word\_data.csv”的结构与附件2的“word\_test.csv”相同，包含text\_id和caption字段，因此不在做过多赘述；而“image\_test.csv”则对应图像ID信息，并与“word\_data.csv”中的文本信息匹配，其结构与附件2的“image\_data.csv”一致。

通过深入探究这四个附件，本文将明晰了各个数据集的组成结构、字段含义及其内在联系，从而为后续的数据预处理[[[2]](#endnote-1)]、模型搭建、性能评估以及结果提交等步骤提供了明确的方向指引。

## 技术探索

在深度学习推动的计算机视觉与自然语言处理[[[3]](#endnote-2)]（NLP）领域中，一系列创新模型正在塑造新的技术范式。其中ViT（Vision Transformer）[[[4]](#endnote-3)]、BERT（Bidirectional Encoder Representations from Transformers）与CLIP（Contrastive Language-Image Pre-Training）模型尤为突出。这三者分别代表了Transformer架构在图像处理、文本理解和跨模态交互方面的成功应用，它们以独特设计、强大性能及广泛适用性共同推动人工智能技术向前发展。

### CLIP

在人工智能领域，尤其是计算机视觉和自然语言处理（NLP）的交叉地带，CLIP（Contrastive Language-Image Pre-Training）[[[5]](#endnote-4)]模型以其卓越的跨模态理解能力和广泛的应用前景，成为近年来备受瞩目的研究焦点。CLIP模型由OpenAI于2021年初首次发布，其创新性的设计理念与强大的泛化性能不仅推动了多模态研究的进步，也为诸多实际应用带来了革新性的解决方案。本文旨在深入探讨CLIP模型的原理、技术创新、关键优势及其在各领域的应用。

CLIP是一种深度学习模型，专为跨模态任务设计，旨在建立图像与文本之间的高效且通用的关联。模型的核心思想在于利用大量的无标注图像-文本对进行对比学习，通过端到端的训练使模型学会在统一的嵌入空间中，将视觉信息与对应的自然语言描述紧密对齐，同时拉开与无关描述之间的距离。这种训练方式无需依赖传统的类别标签，而是直接从大规模互联网数据中挖掘潜在的语义关联，极大地扩展了模型的泛化能力和应用场景。

CLIP的核心训练机制是对比学习，它将每一对图像、文本视为正样本，其余组合视为负样本。通过最大化正样本对的相似度得分，同时最小化负样本对的相似度，模型逐渐学会捕捉图像与文本之间的语义一致性。这种自监督学习方法摒弃了对大规模人工标注数据的依赖，实现了从海量未标记数据中自动学习跨模态表示。

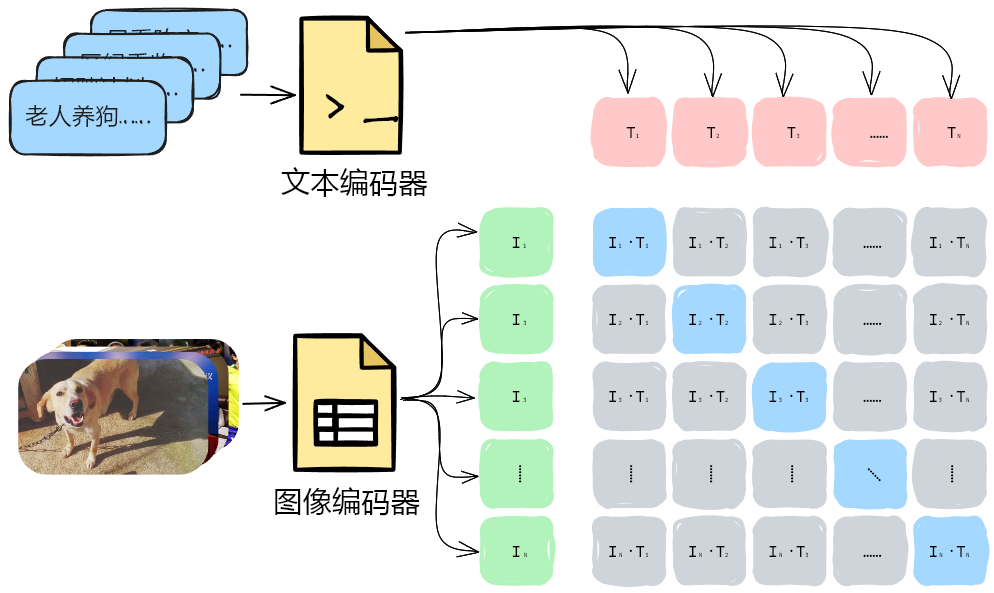


图 3 Clip原理图

CLIP采用独立的视觉编码器（如ResNet或Vision Transformer）和文本编码器（如Transformer），分别对图像和文本进行特征提取。随后，两个模态的高维特征被投影到共享的嵌入空间，使得模型能够直接计算图像与文本之间的余弦相似度，作为它们语义相关程度的度量。这种跨模态融合策略使得CLIP能够在不同任务中灵活应用，只需根据新任务的需求生成文本描述，即可对未知类别的图像进行零样本（zero-shot）分类或检索。

CLIP的成功很大程度上得益于其庞大的预训练数据集。OpenAI收集了约4亿对图像-文本对，涵盖了广泛的视觉内容和语言描述，确保模型在训练过程中接触到丰富多样的场景和表达方式。这种大规模数据驱动的训练策略赋予CLIP出色的泛化能力，使其能够在未见过的视觉概念和语言表述面前保持稳健的表现。

**零样本（Zero-Shot）能力：**CLIP最显著的优势在于其无需针对特定任务进行微调即可进行零样本分类。只需为每个类别提供一个文本描述，模型就能依据这些描述与待分类图像的相似度来进行分类决策。这种能力极大地降低了模型在新任务或新类别上部署的成本，使其能在快速变化的现实环境中快速响应。

**泛化性能：**由于其对比学习训练方式和大规模预训练数据的支持，CLIP在面对未见过的视觉内容或语言描述时展现出卓越的泛化能力。无论是跨领域迁移、抵抗噪声干扰还是处理细粒度分类任务，CLIP都能保持较高的准确率和稳定性。

**无需特定领域专业知识：**CLIP的学习过程不依赖于任何特定领域的专业知识或人工设计的特征，而是直接从大规模数据中学习通用的跨模态表示。这使得CLIP能够应用于广泛的领域，不受限于特定领域的知识壁垒。

CLIP则在计算机视觉与NLP的交叉地带崭露头角，通过对比学习将图像与文本紧密联系起来。CLIP利用大量无标注图像-文本对训练，使模型学会在统一嵌入空间中对齐视觉信息与对应文本描述，实现跨模态理解。其独立的视觉编码器（如ResNet或ViT）与文本编码器（如Transformer）提取各自模态特征，然后映射到共享空间计算相似度，实现零样本分类或检索。CLIP的成功得益于其庞大的预训练数据集（约4亿对图像-文本对）及对比学习策略，赋予模型出色的零样本能力、泛化性能以及对特定领域专业知识的低依赖性。CLIP在图像分类、检索、生成、问答、对话及内容审核等多领域展现巨大应用价值，成为推动多模态人工智能发展的关键力量。

综上所述，ViT、BERT与CLIP分别在图像处理、文本理解和跨模态交互中彰显了Transformer架构的强大通用性和革新性。它们通过各自的独特设计与训练策略，不仅在各自领域取得了突破性进展，还促进了跨领域的知识交融与技术创新。

### ViT

随着深度学习在计算机视觉领域的持续发展，一种名为视觉Transformer（Vision Transformer, ViT）的新型架构正在重塑我们对图像理解和处理的认知。自2020年首次提出以来，ViT凭借其独特的设计和优越的性能，迅速成为图像识别、目标检测、语义分割等任务中的重要工具，标志着从传统的卷积神经网络[[[6]](#endnote-5)]（Convolutional Neural Networks, CNNs）向Transformer架构的成功迁移。

ViT最显著的优点在于其强大的全局建模能力。自注意力机制摒弃了CNN中的局部视野限制，允许模型直接学习图像中任意两个patch之间的长距离依赖，这对于识别全局布局、物体间相互作用等复杂场景尤为关键。这种全局视角有助于提高模型在处理大规模、复杂结构或稀疏分布的视觉特征时的性能。

Transformer架构最初在自然语言处理领域取得巨大成功，ViT的出现进一步证明了其在跨模态任务中的通用性。由于ViT与NLP中的Transformer架构高度一致，这促进了计算机视觉与自然语言处理领域的知识共享和技术融合，为多模态学习（如CLIP、DALL-E等模型）的发展奠定了基础。

ViT的出现标志着图像处理领域从传统的卷积神经网络（CNNs）向Transformer架构的成功迁移。ViT将输入图像切割成固定大小的patch，并将其转化为向量序列，类似NLP中将单词转化为词嵌入。每个patch向量携带位置编码，以保持其在原始图像中的空间信息。这些patch序列随后进入Transformer Encoder，通过自注意力机制实现全局信息交互，捕获长距离依赖关系，而FFN则进行非线性变换增强表达能力。ViT的全局建模能力和并行计算友好性使其在图像识别、目标检测、语义分割等任务中展现卓越性能。其与NLP Transformer的高度一致性也促进了跨模态任务的知识共享和技术融合。

鉴于视觉Transformer（ViT）在现代计算机视觉领域的革新性地位及其在全球建模和跨模态学习方面的出色表现，Chinese-CLIP[[[7]](#endnote-6)]项目巧妙地借鉴了这一先进技术，将ViT架构引入到中文情境下的跨模态研究中，以应对中文图文检索、特征生成和零样本图像分类等更具挑战性的任务。Chinese-CLIP通过结合ViT强大的全局上下文感知能力与专门针对中文优化的文本编码器（如RoBERTa-wwm系列模型），成功构建了一种能够有效理解和关联中文文本与图像信息的统一表示空间。在Chinese-CLIP中，不同的ViT变体（如ViT-B/16、ViT-L/14以及ViT-H/14等）被精心设计并训练以适应大规模中文图文对数据集，从而提取出高质量的图像特征向量。这些向量与来自文本编码器的相应文本特征紧密结合，使得模型不仅能够实现精准的图文匹配，还能在没有见过特定类别示例的情况下进行有效的零样本图像分类。

### RoBERTa

RoBERTa[[[8]](#endnote-7)]是一种对BERT（Bidirectional Encoder Representations from Transformers）模型的重要改进与优化，由Facebook的研究团队于2019年提出。BERT作为自然语言处理领域的突破性进展，通过预训练和微调机制成功捕获了词汇和句子级别的语义信息。

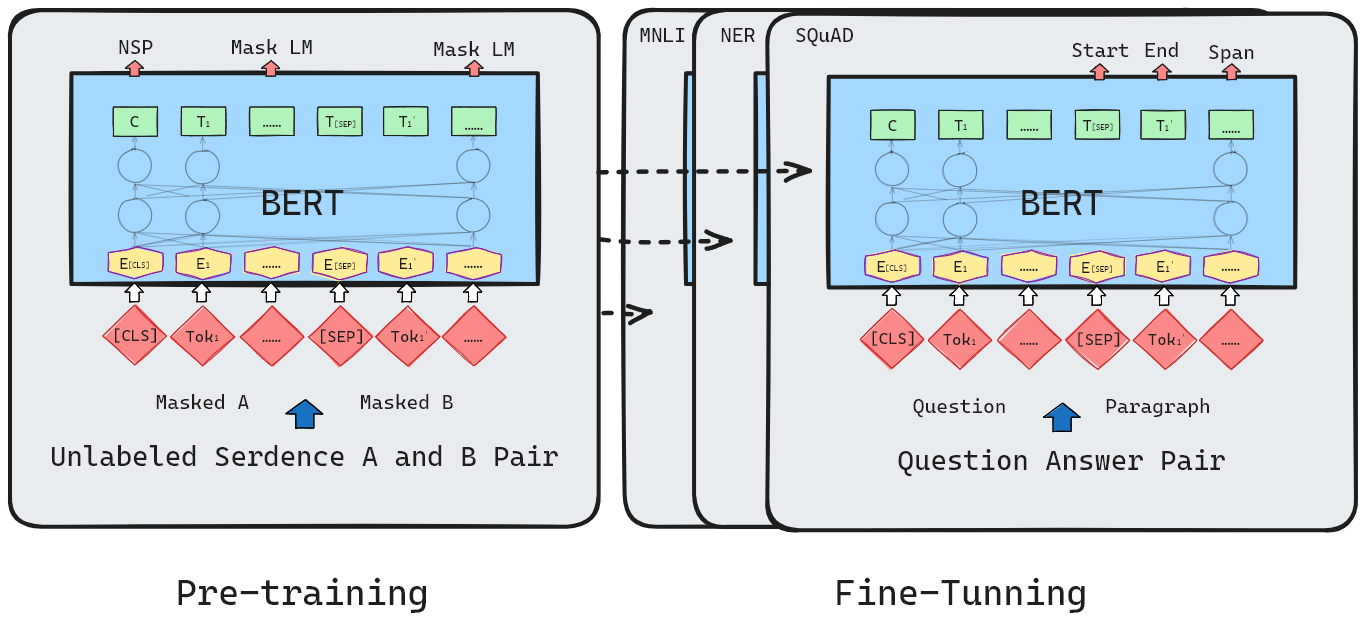


图 4 BERT的训练方法与原理

RoBERTa进一步强化了BERT的预训练过程，RoBERTa使用了更大的训练数据集，并取消了BERT原模型中的“下一个句子预测”（Next Sentence Prediction, NSP）任务，认为连续段落之间的关联性在大量数据中自动学习更为有效。并且延长了训练时间，并增加了每次训练的批量大小，这有助于模型更充分地挖掘和学习语言规律。RoBERTa不再在两个连续样本之间插入特殊的分隔符，使得模型能够更好地理解文本序列的整体性。

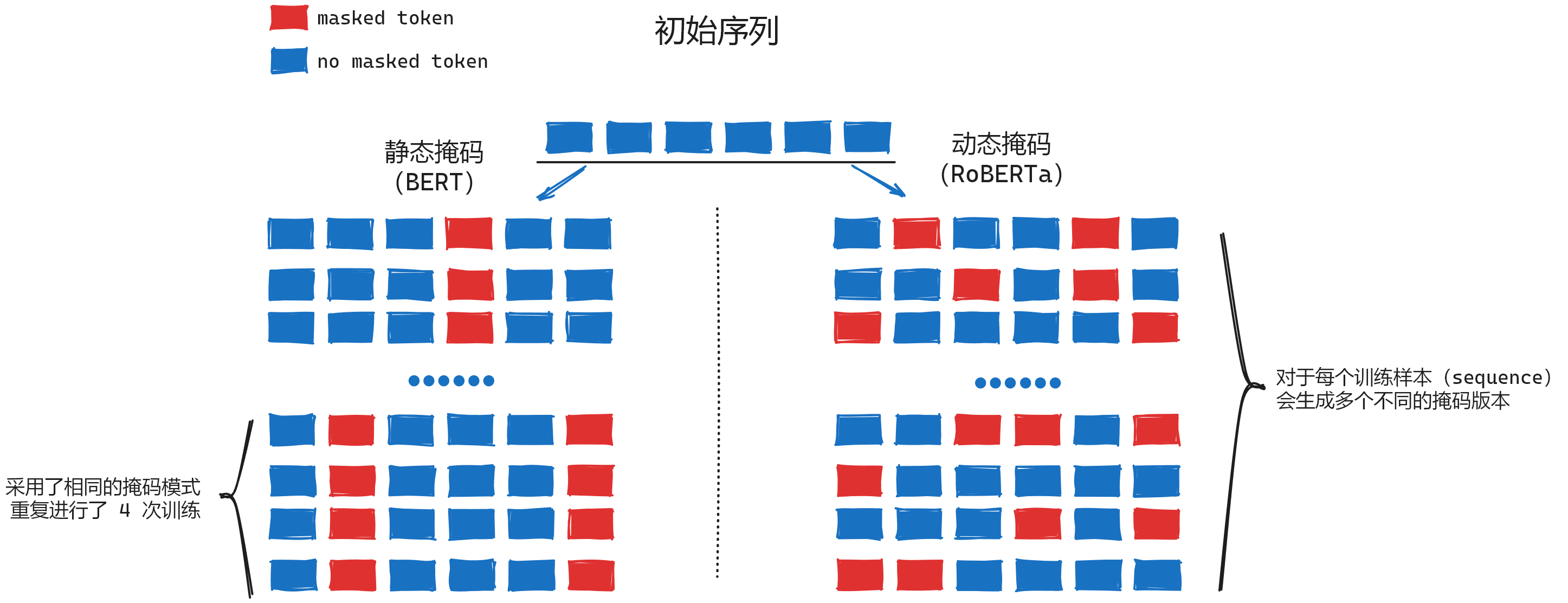
虽然RoBERTa项目本身不涉及动态mask策略的变化，但一般在讨论时指出它采用更严格的掩码策略，即每次训练都会重新遮蔽输入序列的一定比例。

图 5静态掩码vs动态掩码

综上所述，RoBERTa通过对预训练过程的优化，显著提升了模型在众多下游自然语言处理任务中的表现，从而成为BERT之后的一个重要基准和广泛应用的基础模型。在Chinese-CLIP项目中，RoBERTa框架被应用于中文文本的编码阶段，利用其对中文文本的强大理解能力来实现跨模态检索和特征生成等任务。

## 文本处理

CLIP模型特别之处在于它通过对比学习将图像和文本特征空间对齐，以便在没有见过的类别上进行零样本分类。这意味着模型对输入文本的语义理解能力直接影响其跨模态检索性能。清理文本数据，确保其清晰、规范且无干扰，有助于CLIP模型更好地捕捉图像与文本之间的语义关联，进而提升其在检索任务中的表现。

为了使得模型在训练过程中能够更高效地学习关键特征和模式。去除冗余和噪声对于训练模型前期阶段非常重要，这可以减少过拟合风险，最终提升图文检索的准确性和召回率。清理文本数据是搭建图文检索CLIP模型前必不可少的环节，旨在提高数据质量，增强模型训练效果，确保合规性，并最终提升模型在实际应用中的图文检索性能。

### 数据清洗

文本数据通常来源于各种开放源、社交媒体、网页抓取等渠道。清理文本数据有助于剔除这些噪声，确保模型专注于学习图文间的关联性。

文本数据可能存在多种格式、编码、拼写或语法差异。例如，不同的标点符号使用、大小写规则、缩写形式、数字与单位表示等。清理文本数据时，可以进行统一转换（如全小写化、标准化标点符号、规范化日期和数字格式等），使得模型无需花费额外学习成本去适应这些格式差异，从而更有效地提取语义特征。

通过正则表达式[[[9]](#endnote-8)]清洗文本，可以有效剔除非文字字符、特殊符号等无关噪声，使模型聚焦于有意义的语言信号，提升特征提取的精准度。

在涉及中英混合文本的场景下，对英文字符进行小写化处理有助于模型在处理两种语言时保持一致性，特别是在需要比较中英文词汇相似性或进行跨语言检索时，这种处理方式能减少因大小写差异导致的误差。

### 异常值处理

从CSV文件中读取数据时，研究遇到编码相关的问题。由于历史原因、操作系统差异、软件兼容性等因素，文本文件可能采用不同的字符编码格式。如果不正确地处理这些编码问题，程序可能会抛出UnicodeDecodeError异常，阻碍数据的正常读取与解析。

UnicodeDecodeError是Python中与字符编码相关的标准异常之一，通常在尝试解码字节串为Unicode字符串时触发。当给定的字节串无法按照指定的编码规则正确解析为Unicode字符时，Python会抛出此异常。

本文主要通过两个函数来处理UnicodeDecodeError，使用chardet库检测CSV文件的编码方式。该库通过对文件内容进行分析，返回一个包含编码类型的字典。对于每一行数据，尝试使用decode()函数，以指定的编码方式将字节串转换为Unicode字符串，并设置errors='replace'参数。这使得遇到无法解码的字符时，将其替换为特殊字符\ufffd（Unicode REPLACEMENT CHARACTER）。

虽然替换策略保证了数据读取的完整性，但\ufffd字符对后续数据分析无实际意义。通过正则表达式将其替换为空字符串，既保持了原始数据结构，又消除了无效字符的影响（详情代码请见提交文件夹Code）。

通过检测文件编码、替换无法解码的字符、正则清理\ufffd字符以及捕获并优雅处理UnicodeDecodeError，有效地应对了CSV文件读取过程中的编码问题。这种处理策略兼顾了数据完整性、错误容忍度和程序稳定性，为后续的数据分析工作提供了可靠的数据源。

## 图片处理

本文根据训练的图片数据，采用了一套自定义的图片预处理流程，旨在统一不同格式和尺寸的图像数据，使之适应深度学习模型的要求。以下是图片预处理的核心步骤。在本次的项目中，针对训练用的多样化的图片数据集，本文设计并实施了一种定制化的图像预处理流水线，其主要目标在于规范化和标准化各种格式与分辨率的图像素材，确保它们能够满足深度学习模型对于输入数据的一致性和适配性要求。

### 异常值处理



图 6图片内部格式异常

进行图像文件的深度分析与处理过程中，时常遭遇一种典型的数据一致性问题，即图像文件的实际内在编码格式与其表面所标识的文件扩展名存在不匹配现象。这一情况可能导致数据解析错误、图像无法正确显示或潜在的信息丢失等问题，对于自动化图像处理流水线和基于深度学习的图像识别模型尤其具有挑战性（如在预处理阶段对图像格式统一化的需求）。例如，在本文的研究案例中（参见图 6），某些图像文件虽然其扩展名为.jpg，但实际上它们可能是以.png或其他非JPEG格式编码的。

为了准确地识别此类格式冲突的图像，本文提出了一种基于文件头信息检测的方法，特别设计了一个名为 check\_if\_gif 的函数，用于核实一个图像文件是否本质上属于 GIF 格式，即使其扩展名可能被误标记为.jpg等非GIF格式。这一现象的具体实例可在附录1、附件2和附录3中找到，分别包含了1350条、447条和68条格式异常的图片记录。

针对上述异常情况，本文的解决方案是对目标文件采用二进制读取模式（binary read mode），确保能够无损地获取原始的字节流信息。通过调用 check\_if\_gif(img\_path) 函数并将待检测的图片路径作为参数传递，这这策略可以有效地鉴别出该图片文件的真实格式是否为GIF。这一过程对于校正由文件命名不准确、文件损坏或其他未知原因造成的图像格式误判具有重要意义。

在表 3中详细列举了研究过程中发现的一些具体实例，它们的内部格式与扩展名所标识的格式明显不符。

表 3内部格式为GIF的图片ID

|  |  |
| --- | --- |
| **附件** | **内部格式与其扩展名不符合片ID** |
| 附件1 | Image14001005-9133.jpg |
| 附件2 | Image14001008-9006.jpg |
| 附件3 | Image14105004-1970.jpg |
| ....... | ......... |

### 尺寸调整

读取到图像后，将其调整至预设的目标尺寸（默认为224x224像素），这将使用OpenCV中的cv2.resize()函数，并设定插值方式为线性插值[[[10]](#endnote-9)]以保证较好的图像质量。本文将按照以下流程对图像进行处理。

首先，运用合适的数据读取接口从存储设备读取原始图像数据。这一步通常涉及文件I/O操作，例如在OpenCV库中，可以使用cv2.imread()函数将图像文件载入内存，并以二维数组形式表示其像素值。接下来，针对已加载的图像，执行尺寸规范化处理以匹配预设的目标尺寸。这是许多深度学习模型对于输入图像的标准要求，有助于统一数据格式并提高模型训练效率。在调整图像尺寸的过程中，由于新的尺寸可能并非原图尺寸的整数倍，因此需要通过插值算法生成位于新尺寸网格上的像素值。这里选用的是线性插值方法，它根据待插值位置与周围原有像素点之间的相对位置关系，通过线性组合这些邻近像素的值来估计新像素点的值。调用OpenCV库提供的cv2.resize()函数，指定原图像、目标尺寸以及插值方法。此函数会自动执行上述尺寸调整和插值过程，输出一个尺寸符合要求的新图像。

这个过程旨在通过精细化的图像预处理技术，在保持图像关键特征的同时，将原始图像适配到目标模型所需的输入规格，而线性插值作为一种常见的低复杂度、高保真度的插值策略，能够有效地在尺寸变换过程中维持图像的质量和信息完整性。

### 数据增强

在实施机器学习模型的训练流程中，为了增强模型对未知数据的概括性和鲁棒性，本文采纳了一项名为“数据增强”[[[11]](#endnote-10)]的核心技术手段。这一策略尤其适用于图像数据集，在预处理步骤中，本文会特意激活数据增强功能以扩充训练集的多样性。

具体操作上，在对接图像预处理流水线并对训练集中的图像执行操作时，本文利用了业界广泛认可的Albumentations库所提供的RandomResizedCrop()函数。该函数通过对输入图像执行随机尺度缩放与裁剪变换，生成一系列在大小、视角以及局部结构上有变化的新样本，同时确保所有经变换后的图像仍遵循预定的目标尺寸规格。这一过程不仅增加了模型在训练周期内所经历的各种视觉模式的数量，还通过模拟潜在的现实世界变化提高了模型的适应性。

然而，在对待验证集和不同测试集的图像时，则谨慎地排除了数据增强的应用。这是由于数据增强的核心价值在于训练迭代过程中减少模型对训练数据的过度拟合现象，通过引入更多的变异性帮助模型提炼出更普遍的特征表示。相反，在验证阶段以及后续的测试阶段，目标转向了对模型在未经任何艺术性修饰的原始数据上的实际预测性能进行无偏见的精确度量，因此，保持验证集和测试集数据的原始状态及其固有的内在分布特性至关重要，这样才能公正地反映模型在未知场景下的真实效能。

### 图片编码与序列化

在图像预处理阶段，首先将图像经过一系列标准化操作转化为Python Imaging Library (Pillow) 的Image对象，随后将其持久化为BytesIO内存流格式，进一步将此内存流内容以JPEG压缩标准编码为Base64格式字符串，这种表示方式方便将图像数据与其对应的文本标签一并存储在Tab-Separated Values (TSV) 文件中，确保了数据集内部图像数据的一致性、可用性和高效存储。

这一系列预处理步骤除了保证数据质量外，还通过集成数据增强技术，有效提升了模型在多种场景下的鲁棒性及泛化能力，有利于模型整体性能的优化。此外，整个流程设计有完备的错误处理机制，能实时捕获并反馈任何因图像文件损坏或无法读取导致的异常情况。

为了实现大规模数据集的高效加载与随机访问，需要执行名为build\_lmdb\_dataset.py的脚本，该脚本的核心功能是将原始图片数据与对应的图像-文本对标注信息序列化并存储至Lightning Memory-Mapped Database 文件系统中。相比于传统使用的TSV和JSON Lines 文件格式，LMDB提供了更优的I/O性能和内存映射特性，尤其适用于深度学习训练过程中的快速数据读取需求。

在脚本运行初期，利用argparse库解析命令行参数，包括待处理数据的根目录、所需处理的数据集分割名称列表以及目标LMDB文件输出目录。首先确认数据目录的有效性，然后依据用户指定的分割名称列举出待处理的具体数据集子集。

附带说明的是，本文展示了一个将图片转换为Base64编码后存储于表格形式的例子，如表 4所示。

表 4转化Base64编码后存储后列表

|  |  |  |  |
| --- | --- | --- | --- |
| **row\_id** | **image\_id** | **caption** | **iamge\_base64** |
| 140010067606 | 45000 | 而对家里的老婆则有些横眉冷对千夫指的感觉 怎么看怎么不顺眼 | \x80\x04\x95l\x00\x00\.....\x8d\x94\x87\x94 |
| 140010067608 | 45001 | 爱丝与哈特鲜榨果汁 长春 | \x80\x04\x95:\x00\x00\.....xe6\x98\xa5\x94\x87\x94 |
| .... | ...... | ...... | ...... |

### 划分训练集

在本文的研究项目中，我们对原始大数据集进行了合理且科学的数据分区策略，旨在构建独立的训练集、验证集以及测试集，以保证模型训练的有效性及准确评估模型的泛化性能。遵循经典的数据挖掘和机器学习范式，文本利用数据集中每条记录的唯一标识符字段——image\_id作为切分依据，将整体数据集按业界广泛接受的80%、10%、10%比例分配到训练集、验证集和测试集。

首先，本文通过数据预处理步骤获取了数据集中所有图片记录的全局唯一ID列表，将其定义为变量 ids。基于此，本文根据总体样本容量，精确计算了三个不同集合各自所需的样本数量：其中训练集大小被设定为原始数据集总长度的80%，表示为 train\_size；验证集则占据原始数据集的10%，记为 valid\_size；余下的10%数据用于构成测试集，标记为 test\_size。

随后，本文依据上述计算结果，运用编程手段从 ids 中等间隔抽取相应数量的ID元素，生成相应的训练集ID列表 train\_ids、验证集ID列表 valid\_ids 和测试集ID列表 test\_ids。这部分操作可以借助伪代码实现如下：

1. # 假设 dataset\_len 是原始数据集的长度
2. dataset\_len = len(ids)
3. # 计算各集合所需样本数
4. train\_size = int(0.8 \* dataset\_len)
5. valid\_size = int(0.1 \* dataset\_len)
6. test\_size = dataset\_len - train\_size - valid\_size
7. # 利用numpy随机抽样方法进行分割
8. train\_ids = np.random.choice(ids, size=train\_size, replace=False)
9. valid\_ids = np.random.choice(np.setdiff1d(ids, train\_ids), size=valid\_size, replace=False)
10. test\_ids = np.setdiff1d(ids, np.union1d(train\_ids, valid\_ids))

最后，在获得各个集合的ID列表基础上，本文对原始DataFrame结构的数据集 df 实施逻辑分割操作。具体而言，本文采用 DataFrame 的 isin() 方法过滤出与各个集合ID列表相匹配的子集，进而生成训练数据子集 df\_train、验证数据子集 df\_valid 以及测试数据子集 df\_test。

这种数据集划分策略不仅确保了训练、验证和测试集之间的统计独立性与随机性，而且能够在最大程度上模拟模型在未曾见过的新数据上的实际表现，从而有效防止过拟合并提供可靠的模型泛化性能评估。此外，它还为后续的模型超参数调优工作奠定了坚实的验证基准。

# 模型构建

## 模型选择

### Chinese-CLIP模型

近年来，Contrastive Language-Image Pre-training (CLIP) 模型因其在多模态理解上的优秀性能和广泛应用前景，引发了全球范围内研究者和开发者的高度关注。在此背景下，Chinese-CLIP应运而生，这是一款专门针对中文图文数据进行设计和预训练的模型。通过深度优化对中文文本的理解能力以及提升与中文图像间的匹配精度，该模型在面对中文环境下特定的图文检索任务时表现出显著的优势。

Chinese-CLIP作为CLIP模型家族的一个重要分支，不仅保留了原版CLIP模型的核心特性，如对比学习（Contrastive Learning）机制，而且结合中文语境进行了深度定制与拓展。研究者和开发者们广泛参与其优化和升级过程，逐步积累了大量的模型变种、优化策略和技术方案，使Chinese-CLIP成为了一个更加成熟且适应性更广的跨模态解决方案。

### Chinese-CLIP的优势

Chinese-CLIP模型在兼容性、训练优化、功能扩展等方面表现卓越，具体体现在：

**框架与格式兼容：**Chinese-CLIP模型实现了与多种深度学习框架的良好集成，例如，官方支持的Pytorch版本已更新至2.0，并提供脚本可将Pytorch模型转化为CoreML格式，以方便在iOS和macOS等移动设备上部署。此外，它还支持ONNX和TensorRT模型格式，极大地增强了模型在不同硬件环境下的部署便捷性和推理速度。

1. import coremltools
2. from torch.onnx import export
3. *# 假设model是预训练好的Chinese-CLIP Pytorch模型*
4. input\_spec = ...  *# 定义输入规范*
5. onnx\_model = export(model, ..., opset\_version=...)  *# 导出为ONNX模型*
6. coreml\_model = coremltools.converters.onnx.convert(onnx\_model)  *# 转换为CoreML模型*

**高效训练技术：**Chinese-CLIP采用了对比学习的梯度累积（gradient accumulation）技术，有效模拟了大批次训练的效果，从而提高了训练效率。同时，新增的FlashAttention模块显著加快了训练速度，减少了GPU显存占用，有利于在大规模数据集上进行训练。

**易用API接口：**该项目提供了一套图文特征提取的快速API接口，使得开发者只需编写少量代码即可实现中文图文特征的抽取以及相似度计算，极大地提升了工作效率。

### Chinese-CLIP模型规模与配置

Chinese-CLIP当前开源了五种不同规模的预训练模型供用户选择，每种模型的具体配置如下表 5所示：

表 5模型信息

|  |  |  |  |
| --- | --- | --- | --- |
| **模型规模** | **视觉侧骨架** | **文本侧骨架** | **分辨率** |
| CN-CLIPRN50 | ResNet50 | RBT3 | 224 |
| CN-CLIPViT-B/16 | ViT-B/16 | RoBERTa-wwm-Base | 224 |
| CN-CLIPViT-L/14 | ViT-L/14 | RoBERTa-wwm-Base | 224 |
| CN-CLIPViT-L/14@336px | ViT-L/14 | RoBERTa-wwm-Base | 336 |
| CN-CLIPViT-H/14 | ViT-H/14 | RoBERTa-wwm-Large | 224 |

综上所述，Chinese-CLIP模型凭借其强大的兼容性、先进的训练优化手段、丰富多元的功能扩展、多样化的预训练模型选项以及高效的API调用方式，为跨模态研究领域带来了显著的进步，并在实际应用场景如图文检索、分类等任务中提供了强有力的技术支撑。

## 特征提取

### 文本特征

在构建深度学习模型的文本特征提取阶段，首先采用bert\_tokenizer.py脚本对原始文本进行细致的预处理和分词操作。此脚本的核心功能包括但不限于以下几个方面：

**词汇表管理：**加载预先定义好的词汇表，支持Unicode字符转换，并实现词汇表之间的相互转换以适应不同场景需求。

**分词算法集成：**包含了空白字符分词策略及可配置的全功能分词器，能够灵活运用多种分词方案，如基于规则的分词、最长匹配等。

**BERT特定预处理：**实现BERT所需的WordPiece分词算法[[Devlin et al., 2018]]，将文本分割成token片段，并附加上相应特殊符号（如CLS、SEP等）。

**模型一致性校验：**在初始化tokenizer时，会对模型参数和预训练模型的checkpoint进行一致性校验，并提供将tokenized结果序列化回字符串的功能。

**预处理流水线：**通过上述步骤构建了一条完整的BERT模型输入文本预处理流水线，确保输入数据与模型期望格式一致。

另一方面，在model\_definition.py中设计了一种深度学习架构，它围绕着一个定制版的BERT模型展开，并结合了一系列自注意力层组件。定制的CustomizedBert模型继承并拓展了原始BERT结构，主要体现在初始化时对词汇表大小、隐藏层维度（hidden\_size）以及最大位置嵌入数的设定，并保证隐藏层维度可以被注意力头数量整除。

1. class CustomizedBert(nn.Module):
2. def \_\_init\_\_(self, vocab\_size, hidden\_size, num\_heads, max\_position\_embeddings):
3. super(CustomizedBert, self).\_\_init\_\_()
4. self.bert\_embeddings = BertEmbeddings(vocab\_size, hidden\_size, max\_position\_embeddings)
5. self.transformer\_layers = nn.ModuleList([BertLayer(hidden\_size, num\_heads) for \_ in range(num\_layers)])
6. *# ...*

在模型处理输入序列时，首先通过BertEmbeddings层生成词嵌入、位置嵌入和类型嵌入，并将它们合并后经过LayerNorm标准化和dropout操作。Transformer模块内部由多个BertLayer组成，每个BertLayer又细分为BertAttention、BertIntermediate和BertOutput三个子模块。BertAttention部分根据配置参数可以选择标准的多头自注意力机制或多头注意力加速版本如FlashMHA [[假设文献]]。当使用FlashAttention时，需要按照其数据格式要求调整输入张量维度，例如从(batch, seq\_length, hidden\_size)转为(seq\_length, batch, hidden\_size)，并在计算后重新调整回原始格式。

BertIntermediate层利用非线性激活函数（如GELU或其变种[[杨国亮, 许楠, 李放, 等. 关于非线性激活函数的深度学习分类方法研究[J]. 江西理工大学学报, 2018, 39(3): 76-83.]]），对自注意力层的输出进行复杂的特征提取；而BertOutput层通过线性映射、LayerNorm规范化和dropout，生成下一层的输入。

视觉Transformer模型VisualBert的构造中，首先将文本序列通过类似的嵌入转换过程，同时利用位置编码和可能存在的类别嵌入增强输入特征。随后，模型通过一系列BertLayer组成的BERT编码器捕获跨文本序列的信息，并生成全局的输出向量。

此外，为了提高内存使用效率和训练速度，该模型支持梯度检查点（gradient checkpointing）技术。而对于视觉数据，通常会先通过预训练的CNN模型提取图像特征，然后将这些特征转换成适合BERT编码器接受的形式。

针对不同数据集切分（如训练集、验证集），其对应的特征会分别存放在valid\_texts.txt\_feat.jsonl这样的文件中，每行以JSON格式记录单个样本的BERT特征表示：

1. {
2. "text\_id": 40000,
3. "feature": [0.1314, ..., 0.0018, -0.0002]
4. }

### 图像特征

本文在数据预处理阶段执行了一个名为extract\_features.py的Python脚本，旨在从图像数据中提取高级语义特征，并将其序列化至特定格式的文件中。具体来说，其中每个JSON对象代表一幅图像的特征描述，如以下示例所示：

1. {
2. "image\_id": 140010010000,
3. "feature": [-0.012809992767870426, -0.08950697630643845, ...]
4. }

在extract\_features.py脚本内，本文设计并实例化了一个深度神经网络模型，该模型融合了改良版的ResNet（ModifiedResNet）架构和Transformer模块。改良后的ResNet结构进行了以下关键性调整，引入了一个包含三个卷积层的前置模块（stem layer），以增强对初始输入图像的特征提取能力。用全局平均池化层取代了传统的最大池化层，从而更均匀地整合全局信息。在网络的下采样层（如残差块）中应用了反锯齿策略，以改善特征图的空间分辨率降低时的信息丢失问题。将最后一层池化操作替换为基于注意力机制的AttentionPool2d层，利用多头自注意力机制对高层特征进行聚合，提高了模型对全局上下文信息的理解。Transformer模块则由一系列堆叠的残差注意力块ResidualAttentionBlock构成，其中每个块可以根据use\_flash\_attention参数选择采用经典的多头注意力机制或是更高效的FlashAttention。当选用FlashAttention时，需注意适配其所需的输入维度格式——即将原本的(batch, channel, height, width)调整为(batch, sequence\_length, feature\_dim)，可通过以下伪代码实现：

1. if use\_flash\_attention:
2. x = x.permute(0, 3, 1, 2)  *# 转置维度以满足(batch, sequence\_length, feature\_dim)*
3. x = flash\_attention\_module(x)
4. x = x.permute(0, 2, 3, 1)  *# 恢复至(batch, channel, height, width)*

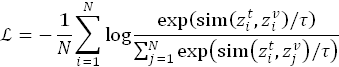
此外，本文还构建了一个视觉Transformer模型（VisualTransformer），该模型首先运用一个卷积层将图像转化为序列化的嵌入表示，接着通过加入位置编码（Positional Encoding）和类别嵌入（Class Embedding）以增强特征表达能力，最后经过Transformer编码器得到高层次的图像特征向量。此模型还集成了梯度检查点（gradient checkpointing）和随机掩码（random masking）等先进的训练技术以提升训练效率和泛化性能。

## 定义损失函数和优化器

### 模型中损失函数选择

在跨模态学习领域，Contrastive Language-Image Pre-training（CLIP）模型采用了一种以InfoNCE损失（也称为Normalized Temperature-Scaled Cross Entropy Loss，NT-Xent损失）为基础的对比学习策略，尤其针对成对输入数据（具体指图像-文本对）进行了优化。该损失函数的核心机制在于通过最大化真实配对图像与其对应文本描述在联合嵌入空间中的相似度得分，同时最小化这些配对与非对应文本或图像（负样本）间的相似度，从而促使模型在高维空间中习得一种能力，即将语义关联紧密的图像与文本映射至彼此距离尽可能近的区域，反之则将无关的图像与文本映射至较远的位置。

对于CLIP模型所采用的损失函数，可形式化表示如下：



在每次迭代过程中，模型不仅针对每个图像与其正确的文本描述计算损失，还会将其与同一批次内的其余所有文本描述（负样本）进行对比。这一损失函数的构造旨在激励模型提高正匹配对之间的相似度得分，同时降低与负样本之间的相似度得分。这种精心设计的对比学习范式使CLIP模型能够习得强大的跨模态通用表征能力，在一系列下游任务，例如图像分类、图像检索、视觉问答等场景中展现出卓越的表现效果。

### 模型中优化器的选择

在本文中，应用了AdamW优化算法进行模型训练，该算法是对原始Adam优化器的一个增强版本，特别针对权重衰减（Weight Decay）机制进行了改良设计。AdamW通过在其更新规则中整合L2正则化的权重衰减，有效解决了在常规Adam优化器中直接结合L2正则化可能导致的权重更新不一致性问题，这一问题源于Adam对梯度二阶矩估计的特殊处理方式。

在深度学习领域，权重衰减是一种常用的正则化技术，用于缓解模型过拟合现象，特别是在处理诸如CLIP这样的大规模预训练模型时，其参数空间巨大，过拟合风险较高。因此，AdamW优化器凭借其对权重衰减的有效处理能力，成为了微调CLIP模型的理想候选优化策略。

其中，args 是一个包含超参数的命名空间对象，包含了学习率（lr）、动量因子（betas）、稳定因子（eps）以及权重衰减率（wd）等关键参数。通过上述代码片段，本文可以根据不同类型的参数灵活地施加不同的权重衰减策略，从而更精细地控制模型训练过程，提升模型泛化性能。

## 微调模型

在微调模型阶段，本文对预训练的Chinese-CLIP模型进行了针对性的适应性训练以达成项目设定的目标效能标准。尽管此模型已在大规模中文图文对数据集上运用对比学习方法（contrastive learning）进行了预训练，并已习得了能够在不同模态间通用的特征表达能力，但在面对特定行业应用环境或专业领域的复杂需求时，例如高度专业化的图像分类、特定语境文本分析及精细化的图文关联推理等问题，未经微调的预训练模型可能难以充分适应并准确解析这些独特性和复杂性。

针对各类下游任务，尤其是那些包含特定领域图像和相应文本标签的实例，如医学影像搭配详细的医学术语描述、商品图片结合详尽的产品说明文本，微调过程对于模型深入理解并精准捕获此类领域内特有的细粒度特征以及上下文关联至关重要。即便预训练模型在一般的图文检索任务中表现出色，在诸如提高检索范围的精确性、增强检索结果的相关性、降低假阳性率等具体检索任务中，依然需要通过微调模型参数，使之更紧密地贴合实际任务需求，以实现性能的优化升级。

微调还涉及到对模型复杂性的调整，使之能够更好地利用小规模而带有特定约束条件的训练数据，进而有效防止模型在面临新任务时出现过拟合或欠拟合的问题。在一些实际应用场合，用户可能期待模型在特定检索评价指标上取得显著优势，比如提升Top-K召回率或降低误检率，这时微调便成为关键手段，用于针对性地提升模型在这类性能指标上的表现。

通过设计的微调流程，本文可以使Chinese-CLIP模型在特定场景和任务中展现出更高的精准度和适用性，最终形成一个针对指定数据集的高度优化、深度融合多模态信息的图像检索模型。

在启动微调训练循环时，采用前向传播算法。这一过程中反复迭代微调模型参数，目标是逐步提升模型的检索效果，特别是召回率。

为了最大限度地提高模型收敛效率和泛化能力，根据验证集性能采取动态的学习率调整策略。这通常包括但不限于余弦退火（cosine annealing）、分段恒定衰减以及其他自适应学习率调整方案，或者结合早停策略（early stopping），当验证集性能连续若干轮未见提升时停止训练。

在每一个训练周期结束后，都会在验证集上对模型性能进行全面评估。当模型满足预设条件，例如验证集性能不再明显提升、达到预先设定的最大训练轮数阈值时，保存当前最优模型权重作为最终微调成果。

以上各个步骤共同构成了微调过程的核心内容，确保了微调后的Chinese-CLIP模型能够在特定应用场景中发挥出色性能，满足甚至超越应用需求。

# 模型评估

## 相似度与检索和排序

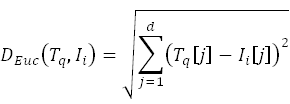
在实现快速检索与查询文本最相关的图像时，采用合适的排序策略至关重要，以确保高效、准确地从数据库中找出与输入文本匹配度最高的图像。以下将详细介绍两种主要的排序策略：KNN检索和ANN搜索。

### KNN检索分析

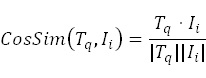
KNN（K-Nearest Neighbors）[[[12]](#endnote-11)]算法是一种基于实例的非参数学习方法，在机器学习和数据挖掘领域广泛应用，不仅可用于分类任务，也可应用于回归分析问题。特别是在跨模态检索场景，例如以文搜图或以图搜图，KNN检索过程可以概括为以下四个关键步骤：

**特征提取阶段：**对输入的文本和图像首先进行预处理，然后采用先进的深度学习模型，如Chinese-CLIP模型或其他适合的嵌入模型，将原始数据映射到一个高维特征空间中，生成对应的特征向量表示。设文本特征向量为T∈ℝd 和图像特征向量为I∈ℝd，其中d代表特征维度，理论上应确保这些特征向量能够充分捕获文本和图像的核心语义信息。

**距离计算阶段：**设定一个有效的距离度量函数，通常包括但不限于欧氏距离（Euclidean Distance）和余弦相似度（Cosine Similarity），用以衡量查询文本特征向量 Tq 与数据库中每个图像特征向量 Ii 的相似度，**欧氏距离公式**如下**:**



**余弦相似度**如下**:**



K个最近邻查找阶段：给定查询文本特征向量 Tq，通过计算其与数据库中所有图像特征向量的距离或相似度，找出距离最小的K个最近邻图像。K值的选取取决于实际应用场景和性能需求，一般而言，较小的K值可以提升响应速度，但可能降低检索的召回率。

**结果排序阶段：**将找出的K个最近邻图像按照距离度量结果由近到远进行排序，距离越小则表明文本与图像的语义关联性越强，相应图像应当在搜索结果中优先展示。KNN检索算法因其直观易懂且实现相对简便而受到青睐。然而，在处理大规模数据库时，实时计算所有样本间的距离并找出K个最近邻会导致较高的时间复杂度（通常为O(Nd)或更糟，其中N是数据库样本数量）。因此，对于大型数据库，尽管在小规模数据集上的开销可接受，仍需考虑引入索引结构（如KD树、球树等）、降维技术或采样策略来优化检索效率。

### ANN搜索分析

ANN（Approximate Nearest Neighbor）[[[13]](#endnote-12)]搜索是一种针对大规模数据集优化的近似检索方法，旨在在可接受的精度损失下，大幅提高检索速度。常见的ANN算法包括哈希方法（如LSH）、树形结构（如KD树、Ball Tree）以及基于量化、降维、图嵌入等技术的高效索引结构（如Annoy、HNSW、Faiss等）。在以文搜图或以图搜图场景中，预先对数据库中的图像特征向量使用选定的ANN算法构建索引结构。这一步通常在离线阶段完成，为在线查询提供高效的检索基础。查询处理：对于输入的查询文本，提取其特征向量，并利用已构建的索引来快速查找与其最相似的图像。ANN算法会返回一定数量（可配置）的近似最近邻，而非精确最近邻，以换取更快的检索速度。结果排序：与KNN类似，按照相似度得分对检索到的近似最近邻图像进行排序，将最相关图像排在前面。

虽然ANN搜索可能会引入一定的精度损失，但在处理大规模数据时，其显著的效率提升往往能够弥补这一点。鉴于本文数据库量级较小，直接使用KNN可能已经足够高效。

### 采用KNN检索与余弦相似度

在这个过程中使用open()打开指定的文本特征文件，逐行读取并解析为JSON对象。对每个解析出的文本，执行以下操作：

获取文本ID和特征向量。初始化空列表用于存储每个图像与当前文本的得分对。将text\_feat转化为一个形状为的CUDA张量。使用循环遍历所有图像特征，每次处理batch\_size个图像：将当前批次的图像特征转化为CUDA张量。计算文本特征与图像特征批次的内积，得到形状为(1, batch\_size)的得分矩阵。遍历当前批次的所有图像ID和得分，将它们作为元组添加到列表。对score\_tuples按得分降序排序，取前k个预测结果。将当前文本ID和对应的前k个图像预测结果写入输出文件，格式为JSON字符串。

该代码实现了基于给定图像和文本特征文件进行k近邻搜索，为每个文本找出与其最相似的k个图像，并将结果以JSONL格式保存到指定文件。整个过程利用了GPU加速计算内积，通过批处理提高效率。

## Recall计算与分析

### Recall公式说明

Recall（召回率）作为评估机器学习与信息检索等领域模型性能的关键指标之一，体现了模型识别相关实例的能力，特别是在目标检测、文本分类或任何二分类任务中的完整性。在严格的数学表述下，召回率量化了模型正确检出所有正类别样本的能力，即系统的全面覆盖程度。

给定一个二分类问题，本文用以下符号来描述预测结果与实际标签之间的比较：

TP(True Positives) 指模型预测为正类且实际也为正类的样本数量，即真正例数。

FN(False Negatives) 指模型预测为负类但实际上应为正类的样本数量，即假反例数。

召回率（Recall）按照以下公式计算：



这个比率表达了模型从所有实际存在的正例中成功找回（或者说正确分类）的比例。理想情况下，召回率应尽可能接近1，这意味着模型几乎找出了所有真实正例，没有遗漏。

1. *# 假设 y\_true 是实际标签列表，y\_pred 是模型预测标签列表*
2. *# 我们先计算TP和FN*
3. TP = sum((y\_true == 1) & (y\_pred == 1))  *# 真正例计数*
4. FN = sum((y\_true == 1) & (y\_pred != 1))  *# 假反例计数*
5. *# 计算召回率*
6. recall = TP / (TP + FN)

召回率的重要性在于其衡量了模型对目标类别的敏感性，尤其是在那些误判成本很高的场景中（例如疾病诊断或安全警报系统），模型必须确保尽量不漏过任何一个正例。然而，单纯追求高召回率可能会牺牲精确率（Precision），导致模型产生较多的假正例，因此在实践中通常需要综合考虑召回率与精确率及其平衡指标 F1 Score 进行模型性能的全面评估。

### Recall的计算

在数据科学领域，本文开发了一个基于Python的评估模块，专门针对信息检索系统的性能评估，特别侧重于计算不同模型的召回率指标（Recall）。

通过read\_submission函数实现，负责验证并解析用户提供的预测输出文件。首先确认文件路径的有效性，然后校验文件内容，确保文本ID及其关联的预测图像ID列表的存在性与格式准确性。此外，还需验证预测结果的数量是否与预设的K值（如K=1, 5, 10）相匹配，并确保提交文件覆盖了所有参考答案中的文本ID。

令P = {p1, p2, ..., pK} 表示对文本ID t 的前K个预测图像ID，确保 |P| = K，并且 t ∈ Tref，其中Tref是参考答案中包含的所有文本ID集合。

利用read\_reference函数从指定路径加载真实标签或参考答案，构建一个哈希表结构，其中键为文本ID，值为相应文本对应的一组正确图像ID列表。

令Rt = {r1, r2, ..., rn} 表示文本ID t 对应的正确图像ID集合。

通过compute\_score函数，对比提交的预测结果与参考答案，统计Recall@1、Recall@5、Recall@10等指标的命中次数。具体地，对于每个文本ID，检测其预测的前10个图像ID列表中是否包含任何与参考答案相符的图像ID，并分别计算Recall@k (k=1, 5, 10)：



report\_score函数负责将计算得出的Recall@1、Recall@5、Recall@10以及综合平均召回率转换为百分比形式，并将这些指标写入指定的JSON格式文件。当发生错误时，调用report\_error\_msg函数记录详细的错误信息、消息以及相应的零分，同样以JSON格式输出至同一文件。

评测脚本展示了在特定检索任务上不同模型的Recall@1/5/10性能表现以下为具体评测结果：

表 6不同模型的Recall@1/5/10性能表现

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模型名称 | 总体得分 | R@1 | R@5 | R@10 |
| CN-CLIPRN50 | 28.72 | 16.52 | 31.24 | 38.4 |
| CN-CLIPViT-B/16 | 39.713 | 25.86 | 42.74 | 50.54 |
| CN-CLIPViT-L/14 | 43.767 | 29.24 | 47.14 | 54.92 |
| CN-CLIPViT-L/14@336px | 44.327 | 30.18 | 48.02 | 54.78 |
| CN-CLIPViT-H/14 | 44.380 | 28.88 | 48.28 | 55.98 |

以上各模型的总体得分是基于各个Recall指标的平均值或其他加权方式得出，以全面衡量模型在给定检索任务上的性能优劣。

### Recall结果分析

通过观察上述表格，我们可以得出结论，本文所实现的Recall指标较低。对此，我们的研究小组进行了深入的数据科学分析，揭示了若干关键因素。

原因之一在于原始文本数据的特性。这些数据具有较强的日常生活语境亲和性，非刻意针对特定类型图片进行描述，从而导致用于训练集的样本整体质量偏低。由于缺乏有效的数据预处理手段以及面对有限的资源约束，研究目前无法高效开展大规模的数据标注工作，进而无法显著提升数据集的整体质量。以下两个实例将进一步阐明此问题：

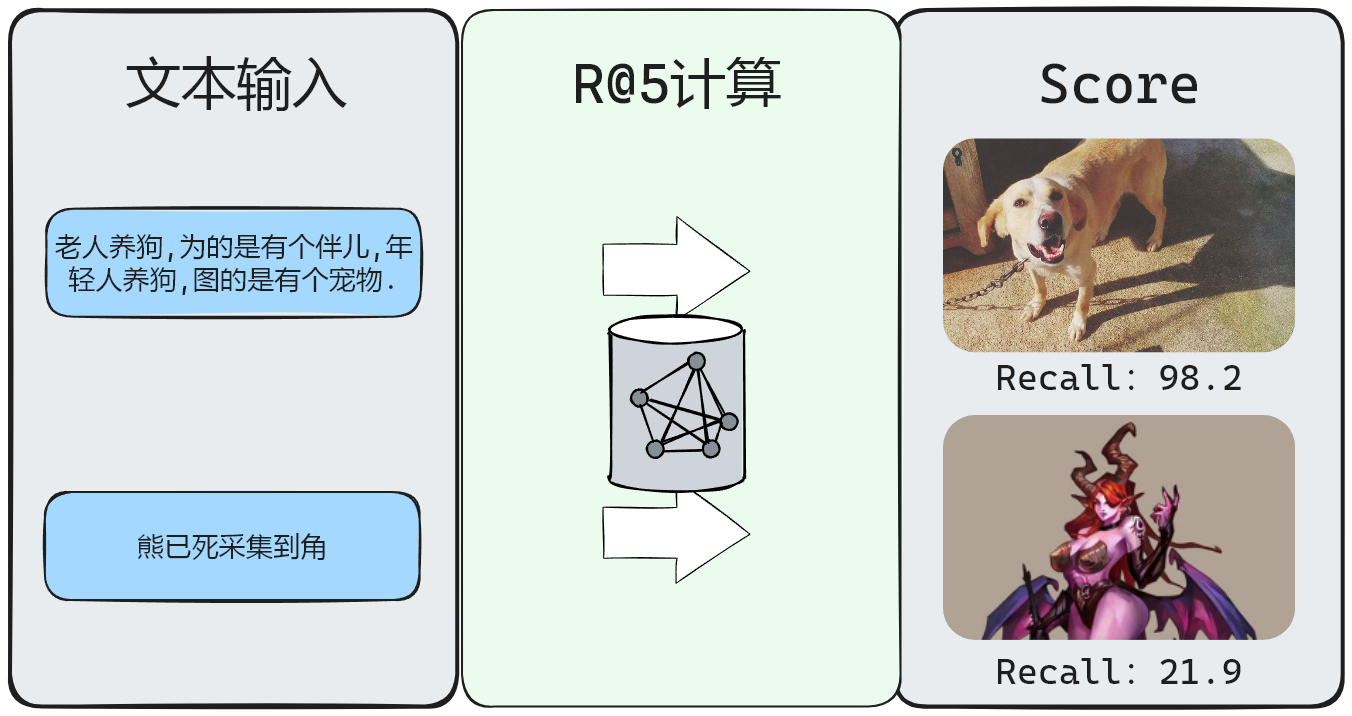


图 7异常数据结果分析

在异常数据分析部分，针对那些紧密联系现实生活的文本输入，本文当前构建的CLIP（Contrastive Language-Image Pre-training）模型可能表现出较差的理解和识别能力。例如，对于描述一种情境的第一句文本，尽管模型可能成功匹配到一张包含一只狗和人的图片，但对另一句文本“熊已死采集到角”，模型很可能仅捕获到与之部分相关的图片，如“熊死亡后被割取部位”的图像，而无法准确匹配到理想的结果。

原因之二是Recall指标的内在性质及其计算方式。在信息检索、推荐系统等领域中，Recall@K是一种常用的性能评估指标，它衡量模型在前K个预测结果中正确包含所有相关项目的比例，这一计算过程遵循固定算法，独立于输入数据的具体变化。具体数学表达式如下：



假设我们有一个数据输入示例：{"text\_id": 248816, "text": "圣诞 抱枕", "image\_ids": [1006938, 561749, 936929, 286314, 141999, 183846]}，这里的召回率量化了系统寻找并返回所有相关项目的能力。给定文本“圣诞抱枕”及其关联的一组高度相关的图像ID列表，若系统理论上应返回所有上述ID作为相关结果，但实际仅提供其中的六个，则召回率会降低，尤其是当1006938这个ID属于实际相关集合时，召回率的具体数值将受到该ID是否被成功召回的影响。简而言之，召回率越接近1，表明系统查找相关项目的效率越高，反之则表明存在遗漏现象。

# 效果展示

## 针对问题一展示

利用附件2中“word\_test.csv”文件的文本信息，对附件2的ImageData文件夹的图像进行图像检索，并罗列检索相似度较高的前五张图像，将结果存放在“result1.csv”文件中（模板文件详见附件4的result1.csv）。下面将展示部分 “result1.csv”文件中的数据并且随机的展示由文本检索出前五张相似度最高的图片。

result1.csv”结果返回数据展示，如表 7所示。

表 7 result1.csv”结果返回数据展示表

|  |  |  |
| --- | --- | --- |
| **text\_id** | **similarity\_ranking** | **result\_image\_id** |
| Word-1000004254 | 1 | Image14001011-0601.jpg |
| 2 | Image14001013-0173.jpg |
| 3 | Image14001013-4317.jpg |
| 4 | Image14001013-7267.jpg |
| 5 | Image14001010-6804.jpg |
| Word-1000030077 | 1 | Image14001007-9398.jpg |
| 2 | Image14001013-0899.jpg |
| 3 | Image14001009-3893.jpg |
| 4 | Image14001011-3648.jpg |

随机展示两条文本对应的相似度最高的前5个图片，如图 1所示：

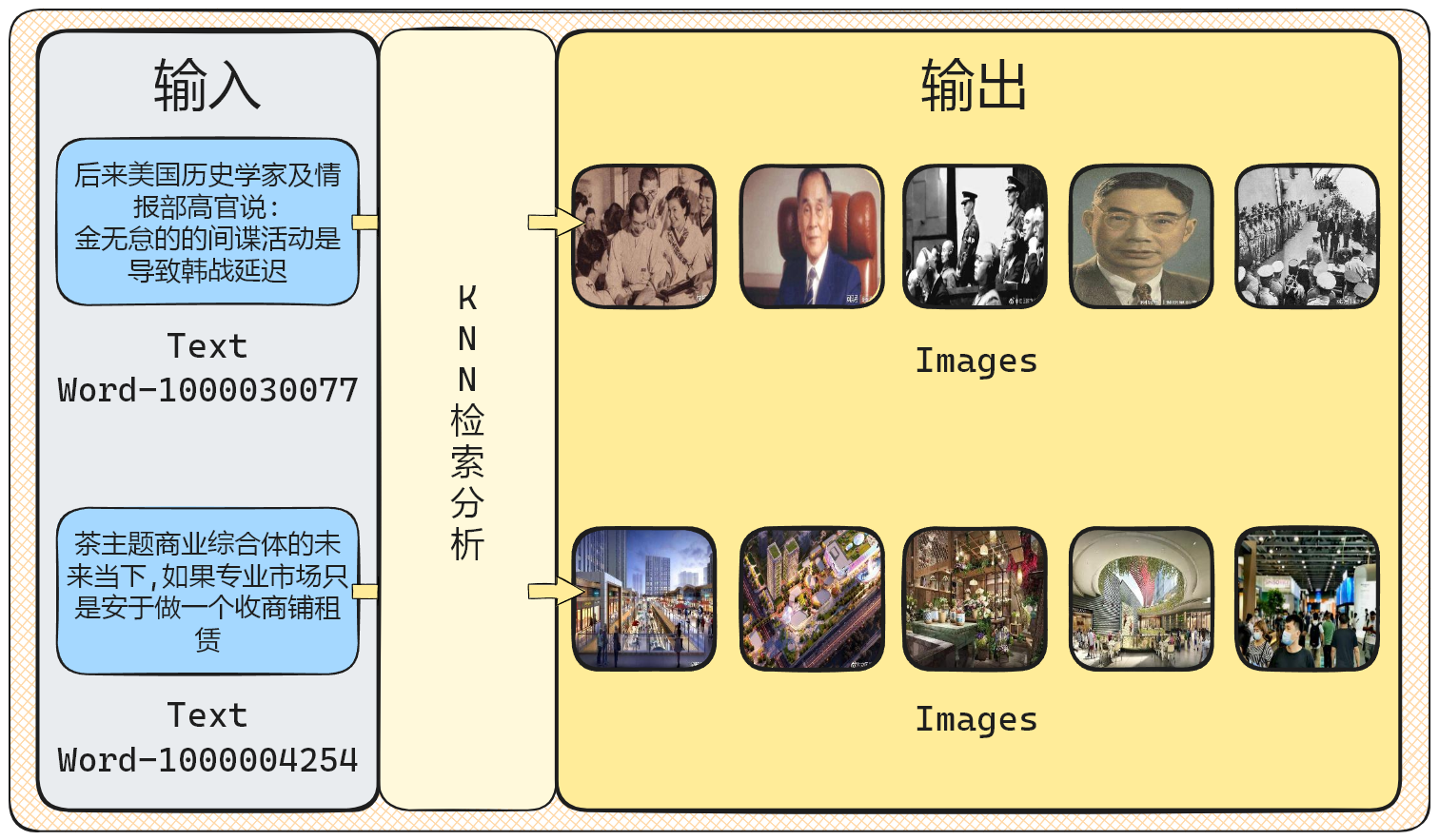


图 8 result1.csv”结果返回数据展示图

## 针对问题二展示

利用附件3中“image\_test.csv”文件提及的图像ID，对附件3的“word\_data.csv”文件进行文本检索，并罗列检索相似度较高的前五条文本，将结果存放在“result2.csv”文件中（模板文件见附件4的result2.csv）。下面将展示部分 “result2.csv”文件中的数据并且随机的展示由图片检索出前五个相似度最高的文本。

“result2.csv”返回结果数据展示，如表 8所示：

表 8“result2.csv”返回结果数据展示表

|  |  |  |
| --- | --- | --- |
| image\_id | similarity\_ranking | result\_text\_id |
| Image14001013-8213.jpg | 1 | Word-1000094713 |
| 2 | Word-1000068663 |
| 3 | Word-1000082606 |
| 4 | Word-1000096985 |
| 5 | Word-1000052501 |
| Image14001013-8214.jpg | 1 | Word-1000067705 |
| 2 | Word-1000093211 |
| 3 | Word-1000076349 |
| 4 | Word-1000053131 |
| 5 | Word-1000099490 |

图片对应的相似度最高的前5个文本，如图 9所示：

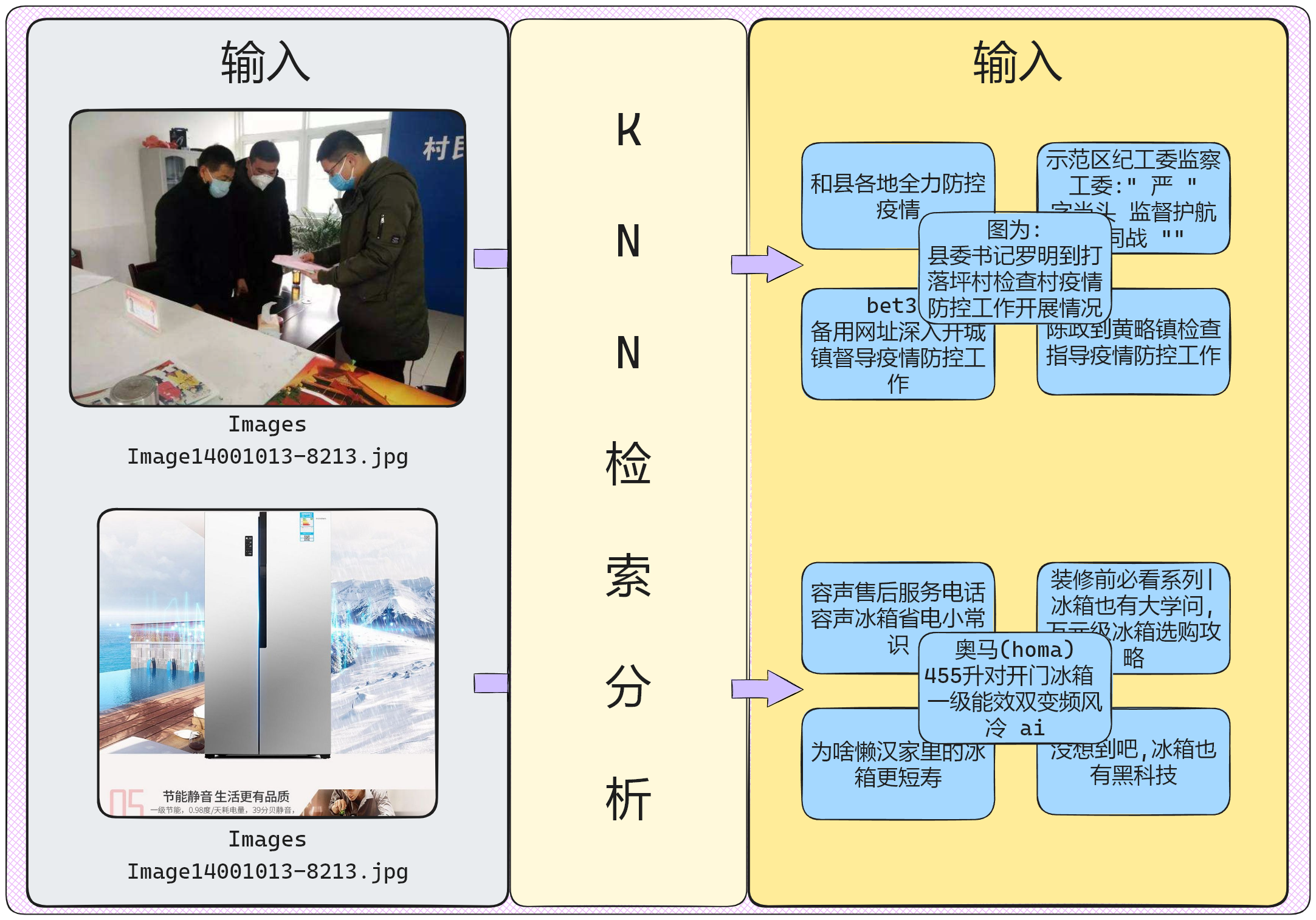


图 9“result2.csv”返回结果数据展示图

# 研究总结

## 回顾项目

本文致力于开发一个实用的跨模态检索系统，该系统旨在高效融合与整合文本和图像两种异构数据源的信息，并运用尖端深度学习架构以实现这一目标。本文采用了预训练的BERT模型以挖掘文本数据的深层次语义特性，同时采用ViT-B/16模型抽取图像数据中的高层次视觉特征。这两者均为深度学习领域的强大工具，能确保从各自模态中提炼出高质量、高维度的特征表示，为后续的跨模态交互与检索任务提供了扎实的基础。

在完成文本和图像特征向量的提取之后，本文采用K近邻（KNN）算法执行跨模态检索过程。针对任意给定的查询文本或图像，KNN通过计算其与存储库中所有样本特征向量之间的余弦相似度来定位最近邻的K个实例。选择余弦相似度作为距离度量准则，能够有效衡量高维特征空间中向量间角度差值，进而体现跨模态数据间的语义相似性。

为了量化评估所构建模型在跨模态检索任务上的表现力，本文选取Recall作为主要的性能评价指标。Recall衡量的是在检索结果集中，真正相关的样本占所有实际相关样本的比例，即模型在正确召回相关样本方面的精确性。通过对不同查询情境下Recall值的计算，我们可以直观评估模型在真实应用场景下的检索效能。

在模型训练阶段，本文采用InfoNCE（Noise Contrastive Estimation）损失函数指导模型优化。InfoNCE损失函数适合于对比学习框架，在该框架内，模型被激励去学习文本与图像之间的跨模态关联性。具体而言，它通过最大化正样本对（文本-对应图像）的相似度得分，同时最小化负样本对（文本-非对应图像）的相似度得分，促使模型在特征空间中将正样本对聚拢而将负样本对拉开距离，从而增强跨模态检索的精准度。

在优化模型参数的过程中，本文选用了AdamW优化算法。相较于原始Adam优化器，AdamW对权重衰减（L2正则化）进行了改良处理，解决了L2正则化与梯度更新联合应用时可能遇到的问题。该算法集成了动量项和自适应学习率调整策略，能够在提高模型训练效率的同时，确保模型参数的有效收敛，并通过权重衰减机制抑制过拟合现象，进一步提升了模型的泛化能力。

## 研究的不足

**一对一数据配对约束性问题**：当前研究在使用一对一的文本与图像配对数据时，模型的检索性能难以通过Recall单一指标得到全面且精确的评估。由于一对一的数据结构限制了检索场景的多样性，模型在检索过程中缺乏充足多样的正负样本对比，无法有效地模拟现实世界中复杂多变的查询条件和潜在的匹配关系。这一局限性可能导致基于Recall计算得出的评估结果存在低估风险。

**训练数据随机性及代表性不足：**目前所使用的训练数据随机性较高，缺乏足够的代表性与针对性，这在一定程度上阻碍了模型对文本与图像之间语义关联性的深入理解与精准捕获。这种状况不仅降低了模型的泛化性能和实际检索效果，也削弱了评价指标的可靠性和稳定性。未来的研究应着眼于构建更为丰富、更具代表性的训练数据集，以期提升模型在各类跨模态检索任务中的综合表现。

# 参考文献

1. 1. 任泽裕, 王振超, 柯尊旺, 等. 多模态数据融合综述[J]. Journal of Computer Engineering & Applications, 2021, 57(18).

   [↑](#endnote-ref-0)
2. 1. 刘明吉, 王秀峰, 黄亚楼. 数据挖掘中的数据预处理[J]. 计算机科学, 2000, 27(4): 54-57.

   [↑](#endnote-ref-1)
3. 1. 徐戈, 王厚峰. 自然语言处理中主题模型的发展[J]. 计算机学报, 2011, 34(8): 1423-1436.

   [↑](#endnote-ref-2)
4. 1. Zhou, Daquan, et al. "Deepvit: Towards deeper vision transformer." arXiv preprint arXiv:2103.11886 (2021).

   [↑](#endnote-ref-3)
5. 1. Li, Yangguang, et al. "Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm." arXiv preprint arXiv:2110.05208 (2021).

   [↑](#endnote-ref-4)
6. 1. 周飞燕, 金林鹏, 董军. 卷积神经网络研究综述[J]. 计算机学报, 2017, 40(6): 1229-1251.

   [↑](#endnote-ref-5)
7. 1. Yang, An, et al. "Chinese clip: Contrastive vision-language pretraining in chinese." arXiv preprint arXiv:2211.01335 (2022).

   [↑](#endnote-ref-6)
8. 1. Liu Y, Ott M, Goyal N, et al. Roberta: A robustly optimized bert pretraining approach[J].arXiv preprint arXiv:1907.11692, 2019.

   [↑](#endnote-ref-7)
9. 1. 娄宝翠. 正则表达式与文本处理技术应用[J]. 河南师范大学学报: 哲学社会科学版, 2012, 39(3): 240-242.

   [↑](#endnote-ref-8)
10. 1. 严家明, 刘诗斌, 李辉. 线性插值的误差计算方法研究[J]. 弹箭与制导学报, 2005, 25(4): 111-112.

    [↑](#endnote-ref-9)
11. 1. Shorten, Connor, and Taghi M. Khoshgoftaar. "A survey on image data augmentation for deep learning." Journal of big data 6.1 (2019): 1-48.

    [↑](#endnote-ref-10)
12. 1. Guo G, Wang H, Bell D, et al. KNN model-based approach in classification[C]//On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy,November 3-7, 2003. Proceedings. Springer Berlin Heidelberg, 2003: 986-996.

    [↑](#endnote-ref-11)
13. 1. Ingre B, Yadav A. Performance analysis of NSL-KDD dataset using ANN[C]//2015 international conference on signal processing and communication engineering systems. IEEE, 2015: 92-96.

    [↑](#endnote-ref-12)