基于大规模预训练 CLIP 模型微调的图文检索方法

摘 要

随着互联网的迅速发展，海量数据的涌现使得从大量信息中筛选出有价值内容变得日益重要。信息检索技术因此成为一项关键技术，尤其是在面对真实世界中复杂的多模态数据时。传统的信息检索模型往往只能处理单一模态数据，而现实情况通常更为复杂，涉及文本、图像等多种数据类型。为了解决这一挑战，本文提出了一种基于自然语言处理（NLP）、计算机视觉（CV）技术，以及多模态模型 **CLIP（Contrastive Language-Image Pre-training）**的信息检索模型。该模型特别针对中文数据集进行了优化，通过大量数据的预训练，并在特定比赛数据集上进行微调，显著提升了检索精度，超越了部分传统中文模型的性能。

本文首先对图像与文本数据进行了建模，通过预处理将异构的图像与文本信息转化为结构化的张量形式，使得基于深度学习的模型能更好地对其处理。接下来，我们对问题一的**图像检索文本**（Image-to-Text，I2T）任务与问题二的**文本检索图像**（Text-to-Image，T2I）任务进行了**统一建模**，将两大检索问题转化为**图像与文本的相似度衡量问题**，按照相似度对检索目标进行排序，以具有最高相似度的目标作为检索结果。为了度量图像文本多模态数据的相关程度，我们引入多模态模型 CLIP 的结构，其基于**对比学习（Contrastive Learning）**方法，训练出能够将图像和文本映射到同一嵌入空间的**图像编码器（Image Encoder）**和**文本编码器（Text Encoder）**。这一映射过程便于计算不同模态数据之间的余弦相似度，从而实现高效的图文互检索功能。

具体而言，为了兼顾模型精度与效率，我们实现的 CLIP 模型采用 **ViT-L/14** 模型作为图像编码器，以及 **RoBERTa** 模型作为文本编码器。同时，这些编码器均经过充分预训练，显著加快了后续训练的收敛速度。接下来，我们收集了一系列高质量的公开数据集，构建了一个包含大约 8 百万图像文本对的中文多模态**预训练数据集**，并基于该数据集对所实现的 CLIP 模型进行预训练，得到一个**泛化性能强大的基线模型**。基于对问题的统一建模，我们的基线模型在问题一的 I2T 任务中达到了 **77.77%** 的 R@5 精度，在问题二的 T2I 任务中达到了 **78.88%** 的 R@5 精度。

为了进一步挖掘该模型的潜力，我们针对比赛数据集（容量仅为 5 万），通过**图像裁剪**、**图像翻转**、**文本翻译**等方式进行数据增强，得到一个容量为 40 万的**增强数据集**。进而，我们使用增强数据集对预训练的 CLIP 基线模型进行微调。经过仅仅 10 个回合的微调，我们的模型便能在问题一的 I2T 任务中达到了 **88.88%** 的 R@5 精度，在问题二的 T2I 任务中达到了 **89.99%** 的 R@5 精度。该结果相较基线模型有着显著地提升，验证了微调策略与数据增强方法的有效性。

综上，本文对图像检索文本任务与文本检索图像任务进行了统一建模，基于 CLIP 框架实现了一个兼顾精度与效率的多模态模型，并整合高质量公开数据集进行预训练得到泛化性能强大的基线模型。进而，我们对比赛数据集进行增强，并对基线模型微调，显著提升了特定任务下的性能表现，对中文信息检索领域具有重要的理论和实践意义。

关键词： 多模态特征融合　图文检索　预训练—微调　对比学习　深度学习

目录

一、 问题描述与假设

1. 问题背景

随着近年来智能终端设备和多媒体社交网络平台的飞速发展，多媒体数据呈现海量增长的趋势，使当今主流的社交网络平台充斥着海量的文本、图像等多模态媒体数据，也使得人们对不同模态数据之间互相检索的需求不断增加。有效的信息检索和分析可以大大提高平台多模态数据的利用率及用户的使用体验，而不同模态间存在显著的语义鸿沟，大大制约了海量多模态数据的分析及有效信息挖掘。因此，在海量的数据中实现跨模态信息的精准检索就成为当今学术界面临的重要挑战。图像和文本作为信息传递过程中常见的两大模态，它们之间的交互检索不仅能有效打破视觉和语言之间的语义鸿沟和分布壁垒，还能促进许多应用的发展，如跨模态检索、图像标注、视觉问答等。

图像文本检索指的是输入某一模态的数据（例如图像），通过训练的模型自动检索出与之最相关的另一模态数据（例如文本），它包括两个方向的检索，即基于文本的图像检索和基于图像的文本检索，如图 1 所示。基于文本的图像检索的目的是从数据库中找到与输入句子相匹配的图像作为输出结果；基于图像的文本检索根据输入图像，模型从数据库中自动检索出能够准确描述图像内容的文字。然而，来自图像和来自文本的特征存在固有的数据分布的差异，也被称为模态间的“异构鸿沟”，使得度量图像和文本之间的语义相关性困难重重。

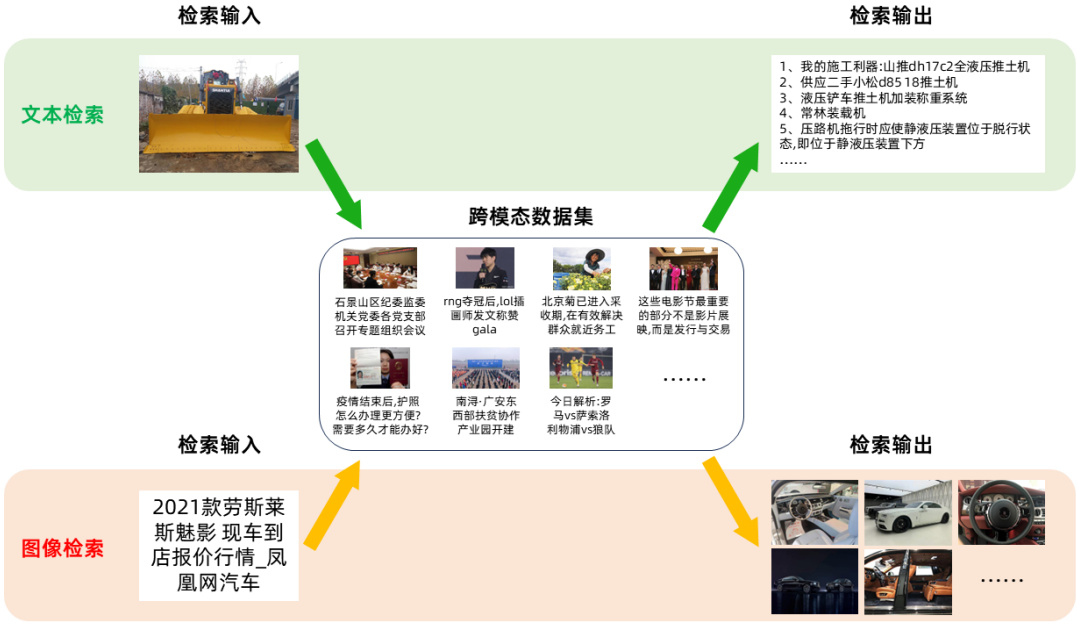

图 1: 图像文本检索

2. 解决问题

本赛题是利用附件 1 的数据集，选择合适方法进行图像和文本的特征提取，基于提取的特征数据，建立适用于**图像检索**的多模态特征融合模型和算法，以及建立适用于**文本检索**的多模态特征融合模型和算法。基于建立的“多模态特征融合的图像文本检索”模型，完成以下两个任务，并提交相关材料。

(1) **图像检索文本**

基于图像检索的模型和算法，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）。其中，ImageData 文件夹中的图像 ID 详见附件 2 的“image_data.csv”文件。

(2) **文本检索图像**

基于文本检索的模型和算法，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）。其中，“image_test.csv”文件提及的图像 ID，对应的图像数据可在附件 3 的 ImageData 文件夹中获取。

3. 评估指标

图像文本检索包括两个具体的任务，即文本检索（Image-to-Text，I2T），即针对查询图像找到相关句子；以及图像检索（Text-to-Image，T2I），即给定查询语句检索符合文本描述的图像。为了与现有方法公平地进行比较，在文本检索问题和图像检索问题中都采用了广泛使用的评价指标：召回率 Recall at K（R@K）。R@K 定义为查询结果中真实结果（Ground Truth）排序在前 K 的比率，通常 K 可取值为 1、5 和 10，计算公式如式 (1) 所示。

$$R@K = \frac{\text{Matched}_{\text{top-}K}}{\text{GroundTruth}_{\text{total}}} \tag{1}$$

其中，$\text{GroundTruth}_{\text{total}}$ 表示真实匹配结果出现的总次数，$\text{Matched}_{\text{top-}K}$ 表示在排序前 K 个输出结果中出现匹配样本的次数。$R@K$ 反映了在图像检索和文本检索中模型输出前 K 个结果中正确结果出现的比例。本赛题的评价标准设定 $K = 5$，即评价标准为 R@5。

4. 基本假设

为了构建图像文本双向检索模型，我们做出如下合理的假设：

1. 训练数据集中的图像文本匹配关系正确可靠；
2. 训练集与测试集中的图像文本对的具有一致的数据分布；
3. 测试集中，每幅图像都存在与之匹配的文本，每条文本都存在与之匹配的图像。

二、 问题建模

多模态图文检索的本质上是对图像和文本两种模态的信息进行压缩编码，压缩编码过程可以利用传统方法也可以利用深度学习方法，但最终会得到图像和文本的压缩编码嵌入 embedding。

在此基础之上，如果得到的 embedding 是空间对齐的，即两个模态的编码在一个语义空间中，那么就可以利用一般的相似度匹配进行图文检索；如果得到的 embedding 是空间不对齐的，那么就需要学习相似度匹配方法来更好地匹配两个图文编码向量的相似度，这样的效率虽然高，但得到的效果显然没有进行向量空间对齐的方法好。

空间对齐指的是公共空间特征学习方法，相似度学习指的是跨模态相似性度量方法。前者为主流方法，并且现在的方法都是基于深度学习模型，同时目前的 SOTA 模型主要为：CLIP、ALBEF、BLIP-2 这些较为成熟的方法模型。

这里，我们选择的是空间对齐的特征学习方法，将图像与文本投影到同构的特征空间，从而利用相似度匹配进行图文检索。下面，我们分别对图像与文本进行建模。

1. 图像建模

在计算机视觉（Computer Vision，CV）领域，常用的图像表示方法是使用张量。张量是多维数组的扩展，可以表示高维数据。对于彩色图像，我们使用三维张量 $x \in \mathbb{R}^{H\times W\times C}$ 描述。其中，H 表示高度，W 表示宽度，C 表示通道数（对于常见的 RGB 图像，其通道数为 3）。

由于题目数据中的图像具有不同的长宽比、分辨率，不利于模型统一处理。于是，我们按照以下规则，对所有图像进行预处理：

1. 对于所有长宽比小于 2:1 的图像，将其拉伸为 1:1，使用双立方插值法（Bicubic Interpolation）下采样至 224×224 分辨率。
2. 对于所有长宽比大于 2:1 的图像，截断其长边，仅保留长宽比小于 2:1 的部分，再按照规则 1. 进行处理。

至此，我们可以将所有图像的分辨率处理为 224×224，进而使用四维张量 $X \in \mathbb{R}^{N\times H\times W\times C}$ 表示整个数据集，其中 N 为图像数量，$H = W = 224$，$C = 3$。

2. 文本建模

在自然语言处理（Natural Language Processing，NLP）领域，文本被视作一个由单词组成的序列。为了便于表达，将所有可能出现的单词汇集成一张表，称为词汇表（Vocabulary），其中每个单词对应一个唯一的序号（Index）。

为了使用深度学习模型学习单词的语义，我们需要将每个词语用一个固定长度的向量表示，分为稀疏表示（如 One-hot 编码）和分布式表示（如 Word2Vec）。由于稀疏表示的诸多弊端，这里我们采用单词的分布式表示。分布式表示将词转化为一个定长（设为 D_{emb}）、稠密并且互相存在语义关系的向量。此处的存在语义关系可以理解为：分布相似的词，是具有相同的语义的。

如此一来，一切文本都能被映射为一个由定长词向量组成的序列。然而，文本中单词的数量或多或少，因此单词序列的长度无法确定，这是不利于语言建模的。为了解决这个问题，常用的方法

是指定一个最大序列长度（设为 L），然后按以下规则处理不同长度的文本：

1. 对于单词数量小于 L 的文本，在其后方填充若干特殊的单词“<pad>”，使其长度达到 L。
2. 对于单词数量超过 L 的文本，舍弃第 L 个单词后的内容。

于是，我们可以将所有文本处理为长度为 L 的单词序列，其中每个单词被表示为一个 D_{emb} 维向量。也就说说，一段文本可以被表示为一个形状为 $L \times D_{\text{emb}}$ 的矩阵。进而，我们对数据集中所有文本进行处理，得到一个三维张量 $Y \in \mathbb{R}^{M \times L \times D_{\text{emb}}}$，其中 M 是文本数量。

3. 图文检索建模

为了进行图文检索（包括图像检索文本、文本检索图像），关键是定义一个匹配度函数。该函数的输入为一幅图像以及一段文本，输出为图像与文本的内容匹配程度，即 $\text{Match}\,(Image, Text) \in [-1, 1]$。其数值大小表示匹配程度，-1 表示完全不匹配，1 表示完全匹配。

对于图像集合 $X \in \mathbb{R}^{N \times H \times W \times C}$，以及文本集合 $Y \in \mathbb{R}^{M \times L \times D_{\text{emb}}}$，可以得到一个匹配度矩阵 $\text{Score} \in \mathbb{R}^{N \times M}$ 表示所有“图像—文本”对的匹配情况。具体而言，Score 的定义如公式 (2) 所示。

$$\text{Score}\,[i,\ j] = \text{Match}\,(X_i,\ Y_j)\,,\ 1 \le i \le N,\ 1 \le j \le M. \tag{2}$$

(1) 图像检索文本

在图像检索文本（Image-to-Text，I2T）任务中，我们需要为每幅图像寻找与其匹配程度最高的 K 段文本。而每幅图像与 Score 矩阵中的一行所对应，为了实现该目的，我们沿着行方向对 Score 矩阵进行 ArgSort 操作，使每行的文本按照与每幅图像的匹配程度排序，并以检索形式呈现。接下来，取检索矩阵的前 K 列，得到矩阵 $\text{RowTop} \in \mathbb{R}^{N \times K}$，如公式 (3) 所示，其中 [...] 表示子矩阵检索操作。

$$\text{RowTop} = \text{ArgSort}\,(\text{Score},\ dim = 1)\,[:,\ :K] \tag{3}$$

此时，RowTop 的第 i 行对应与图像 X_i 匹配程度最高的 K 段文本的位置，则 I2T 任务的结果如公式 (4) 所示，其中 {...} 表示集合。

$$\text{I2T}\,(X_i) = \{Y_j \mid j \in \text{RowTop}\,[i,\ :]\} \tag{4}$$

(2) 文本检索图像

类似的，在文本检索图像（Text-to-Image，T2I）任务中，我们需要为每段文本寻找与其匹配程度最高的 K 幅图像。而每段文本与 Score 矩阵中的一列所对应，为了实现该目的，我们沿着列方向对 Score 矩阵进行 ArgSort 操作，使每列的图像按照与每段文本的匹配程度排序，并以检索形式呈现。接下来，取检索矩阵的前 K 行，得到矩阵 $\text{ColTop} \in \mathbb{R}^{K \times M}$，如公式 (5) 所示。

$$\text{ColTop} = \text{ArgSort}\,(\text{Score},\ dim = 0)\,[:K,\ :] \tag{5}$$

此时，ColTop 的第 j 列对应与文本 Y_j 匹配程度最高的 K 幅图像的位置，则 T2I 任务的结果如公式 (6) 所示。

$$\mathrm{T2I}\left(Y_j\right)=\left\{X_i \mid i \in \mathrm{ColTop}\left[:,\ j\right]\right\} \tag{6}$$

根据所建立的模型，我们只需实现 $\mathrm{Match}\left(Image, Text\right)$ 函数，得到图像与文本的匹配度，即可完成 I2T 任务与 T2I 任务。

三、 数据分析与处理

1. 数据统计

对爬取的数据进行统计分析是数据预处理的关键步骤，这有助于了解数据集的整体特性和潜在的问题。在统计时，我们不仅关注文件的数量和大小，还细致地分析了图像的平均分辨率，以及分辨率和长宽比的分布情况。首先，我们对数据集内所有图像文件的数量进行了计数，以获得数据集的规模。接着，我们计算了所有文件的总大小，以及单个文件的平均大小。在分辨率方面，我们计算了数据集中所有图像的平均分辨率。有助于我们了解数据集中图像的多样性以及是否存在某些分辨率的图像过于集中或稀缺的情况。长宽比是图像宽度与高度的比值，它对于图像的显示和处理非常重要。我们计算了数据集中每张图像的长宽比，并分析了其分布情况，以识别是否有异常值或一致的趋势。例如，某些图像可能具有非常宽或非常高的纵横比，这可能会对模型训练产生影响。为了进一步的数据清洗和质量控制，我们检查了图像的可读性，排除了损坏或无法解码的文件。我们还对文本数据进行了清洗，移除了重复项和包含无关信息的条目。通过这些细致的数据统计和分析，我们能够确保数据集的质量，为后续的数据清洗、增强和模型训练提供了坚实的基础。

2. 数据清洗

对爬取的数据进行数据清洗，以得到更高质量的数据集，便于模型的预训练。

1. **数据筛选：**从网络爬取的数据中可能包含不相关或质量低下的图像和文本对。通过设置一定的标准，例如图像的分辨率、文本的长度和可读性，筛选掉不符合要求的数据。
2. **数据去重：**数据集中可能存在重复的图像-文本对，这会影响模型训练的效果。通过去除重复项可以提高数据的唯一性和多样性。
3. **错误修正：**自动爬取的数据可能包含标签错误或不准确的文本描述。可以识别并修正这些错误。
4. **避免偏见：**数据集分布可能存在偏移，比如某些类别的样本数量过多或过少，需要进行删减或扩充来平衡数据分布。

3. 数据增强

(1) 图像数据增强

图像数据增强方法主要分为两类，一种类型的增强涉及数据的空间/几何变换，如裁剪和调整大小、旋转和翻转。另一种类型的增强涉及外观变换，例如颜色失真（包括颜色下降、亮度、对比度、饱和度、色调）、高斯模糊和 Sobel 过滤。

SimCLR 用实验证明了数据增强操作的组合对学习好的表征是至关重要的，而且无监督的对比学习受益于比监督学习更强的数据增强。因此，我们所使用的数据增强操作是这几种方法的组合：

1. 随机裁剪和调整大小；
2. 随机旋转；
3. 随机翻转；
4. ……。

(2) 文本数据增强

为了能在不改变原文语义的情况下，生成一定数量的训练语料文本，同时提升模型的泛化性能、干扰波动的能力，我们主要使用以下几种文本数据增强方法：

1. **同义词替换**：在这种方法中，我们从句子中随机取出一个词，将其替换为对应的同义词。
2. **翻译互转**：将文本翻译成另外一种语言，然后再翻译回来。同时，我们也可以翻译成多个语言，从而得到多条回译样本。图 2 演示了该方法的流程。

图 2: 翻译互转示意图。

4. 数据集划分

为了进行训练与测试，我们对“附件 1”中的训练集进行了划分，将其中的 90% 作为真正的训练集，剩下的 10% 作为验证集（这样恰好能使得验证集的大小等于“附件 2”和附件“3”中测试集的大小，更加贴近测试环境），用来评估模型的训练效果。根据验证集上的评估指标，我们可以更准确地掌握模型的性能表现，并量化其泛化能力，避免对训练集数据过拟合。具体而言，表 1 详细说明了数据集的划分情况，以及各个子集的组成。

表 1: **比赛数据集**划分明细

数据来源	“附件 1”		“附件 2”	“附件 3”
数据类型	**训练集**	**验证集**	**I2T 测试集**	**T2I 测试集**
图像数量	45000	5000	50000（样本）	5000（搜索空间）
文本数量	45000	5000	5000（搜索空间）	50000（样本）

四、 多模态特征融合方法研究

模态（Modal）是事情经历和发生的方式，我们生活在一个由多种模态（Multimodal）信息构成的世界，包括视觉信息、听觉信息、文本信息、嗅觉信息等等，当研究的问题或者数据集包含多种这样的模态信息时我们称之为多模态问题，研究多模态问题是推动人工智能更好的了解和认知我们周围世界的关键。

每一种信息的来源或者形式，都可以称为一种模态。例如，人有触觉，听觉，视觉，嗅觉；信息的媒介，有语音、视频、文字等；多种多样的传感器，如雷达、红外、加速度计等。以上的每一种都可以称为一种模态。相较于图像、语音、文本等多媒体 (Multi-media) 数据划分形式，“模态”是一个更为细粒度的概念，同一媒介下可存在不同的模态。比如我们可以把两种不同的语言当做是两种模态，甚至在两种不同情况下采集到的数据集，亦可认为是两种模态。

在本题目中，我们需要对图像—文本，这两种模态的数据特征进行融合。

1. 算法选择

在完成上述任务分析和数据的预处理之后，我们首先可以确定任务的解决方案是基于深度学习的多模态特征融合，不同于传统图像处理和传统机器学习算法与简单神经网络的实现，近些年来基于深度学习的多模态模型针对各种复杂应用场景表现性能更好。

在多模态领域，当前的主流模型是基于对比学习的大规模预训练模型。其代表便是 CLIP 模型。

CLIP（Contrastive Language-Image Pre-training）是一个跨模态学习模型 [1]，由 OpenAI 在 2021 年提出。CLIP 模型的核心思想是通过对比学习的方式，将图像和文本映射到同一个嵌入空间中，使得语义上相关的图像和文本在该空间中更接近。

这里简单介绍一下概念。

2. 背景知识

(1) 对比学习

多模态检索任务的核心是对多模态数据相似度的度量。在同一维度空间下，人们可以通过欧氏距离、闵式距离等方式来计算两个向量之间的距离。因此，为了比较异构数据的相似度, 一个通用的方法是将它们映射到同一个公共表示空间进行相似度的学习。

对比学习（Contrastive Learning）是一种自监督学习方法 [2]，它通过学习数据的相似性和差

异性来学习特征的一般表示。其采用的具体思想是将样例和与它语义相似的样本（正样本）及与它语义不相似的样本（负样本）进行对比，通过设计模型结构和对比损失，使语义相近的样本对应的表示在表示空间更接近，语义不相近的例子对应的表示距离更远，以达到类似聚类的效果，如图 3 所示。

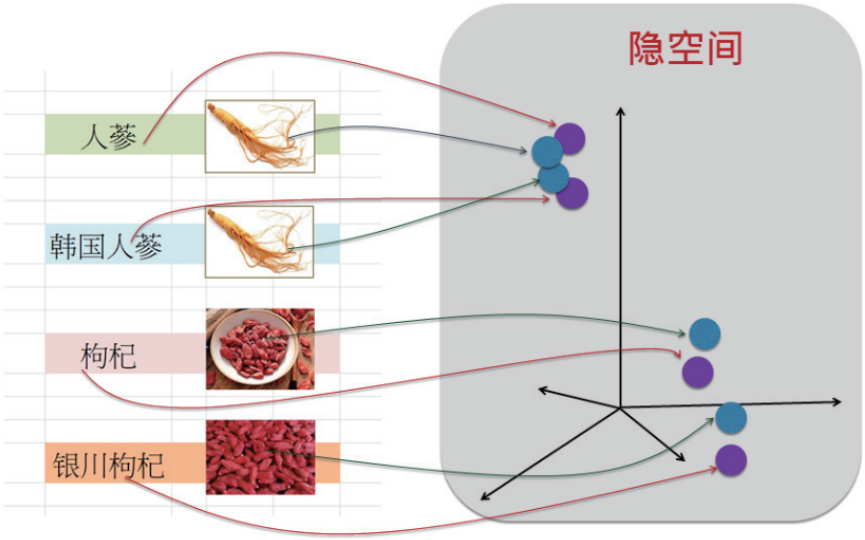

图 3: 对比学习示意图。

对比学习的实施通常包括以下几个方面：

1. **正负样本的定义：**在对比学习中，图像特征和文本特征构成特征矩阵，该矩阵中图文相匹配为正样本，不匹配为负样本，因此特征矩阵的对角线元素均为正样本，其他元素为负样本。
2. **相似度计算：**使用余弦相似度来表示特征之间的相似度，A、B 矩阵的余弦相似度可由公式 (7) 描述。

$$\text{Cosine-Similarity}(\mathbf{A},\mathbf{B}) = \frac{\mathbf{A}\cdot\mathbf{B}}{\|\mathbf{A}\|\cdot\|\mathbf{B}\|} = \frac{\mathbf{A}}{\|\mathbf{A}\|}\cdot\frac{\mathbf{B}}{\|\mathbf{B}\|} \tag{7}$$

3. **损失函数：**定义一个损失函数来训练模型，使得正样本对的相似度高于负样本对的相似度。典型的损失函数包括三元组损失（Triplet Loss）、对比损失（Contrastive Loss）和交叉熵损失等。

(2) 卷积神经网络（CNN）

(3) Transformer

(4) Visual Transformer

(5) 词向量模型

词向量模型是将自然语言中的单词转换成固定长度的向量，以便计算机能够更好地理解自然语言的含义。目前，大多数跨模态匹配领域的文本描述都采用单词到向量（Word2vec）和单词表示的全局向量（Glove）两个词向量模型生成单词嵌入。

单词到向量（Word2Vec）是一种通过对输入语料中的单词进行学习，学习其相互关系，从而生成向量表示的模型。其基本思想是通过学习单词在上下文中的分布来捕捉其语义信息。有两种策略可以实现 Word2Vec 模型: 一个是连续词袋模型（Continuous Bag of Words，CBOW），CBOW 模型是从周围的单词预测中心单词的概率。

另一个实现单词到向量的策略是采用跳字模型（Skip-gram），这个模型是从中心单词预测周围单词的概率，可以通过学习大规模文本数据的词汇共现关系，将每个单词表示成一个固定长度的向量。单词表示的全局向量（Global Vectors for Word Representation，Glove）旨在通过学习单词的全局共现信息来捕捉词语之间的语义关系。与 Word2Vec 中的模型不同，GloVe 是在全局语料库中对所有单词对的共现信息进行建模，而不是基于局部的上下文信息来学习词向量。其优点在于能够同时利用全局的词频信息和局部的上下文信息，因此得到的词向量具有较好的语义表示能力。

(6)　BERT

3.　模型结构

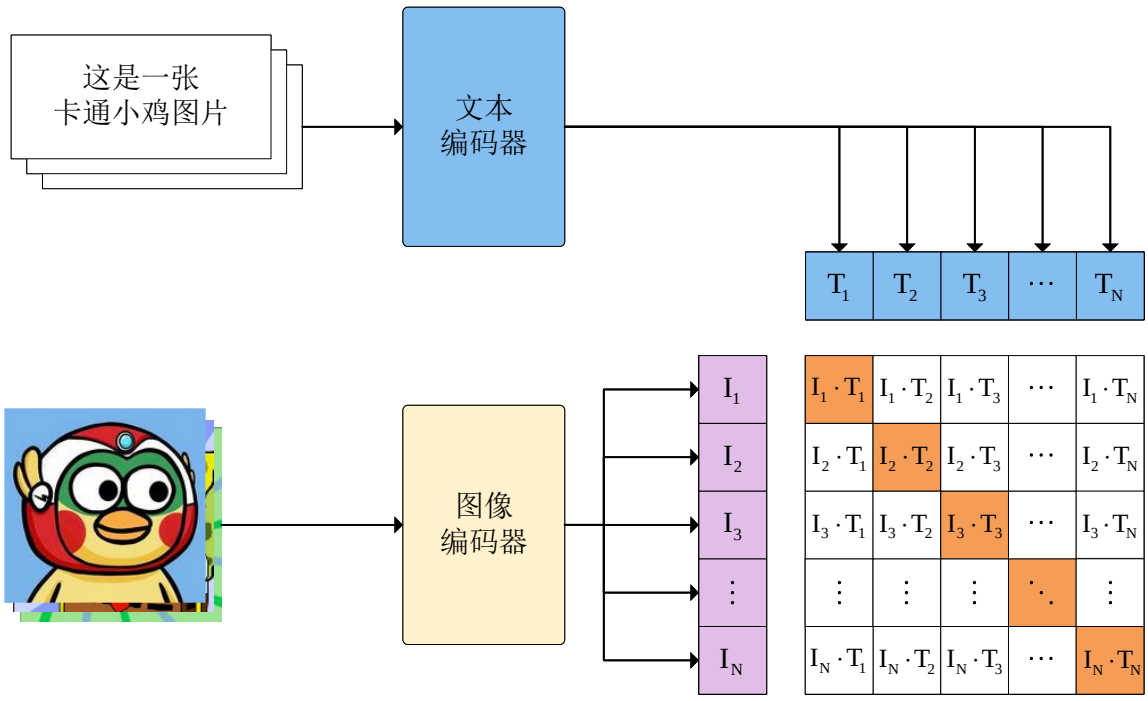

图 4: CLIP（Contrastive Language-Image Pre-training）模型结构示意图。

CLIP 模型的核心思想是通过学习图像和文本之间的匹配关系来提高模型的性能。具体来说，CLIP 模型包含两个主要组成部分：一个用于处理图像的 CNN 模型或 ViT 模型，和一个用于处理文本的 BERT 模型。这两个组件都被训练成能够将输入的信息映射到相同的嵌入空间中，并使得相似的图像和文本在嵌入空间中的距离更近。图 4 演示了 CLIP 模型的结构。

在实现上，为了兼顾模型精度与效率，我们实现的 CLIP 模型采用 **ViT-L/14** 模型作为图像编码器，以及 **RoBERTa** 模型作为文本编码器。同时，这些编码器均经过充分预训练，显著加快了后续训练的收敛速度。

下面，我们分别介绍使用的预训练图像编码器与文本编码器。

(1)　图像编码器

ViT [3] 是 Google 团队提出的将 Transformer [4] 应用在图像分类的模型，因为其模型“简单”且效果好，可扩展性强，于是成为了 Transformer 在 CV 领域应用的里程碑著作，也引爆了后续相关研究。

ViT 最核心的结论是，当拥有足够多的数据进行预训练的时候，ViT 的表现就会超过 CNN，突破 Transformer 缺少归纳偏置的限制，可以在下游任务中获得较好的迁移效果。基于该结论，我们将充分预训练的 ViT 模型作为 CLIP 模型的图像编码器，并选择参数量适中的 ViT-L/14 变种，以平衡精度与计算量的冲突。

在推理过程中，ViT 将输入图像分为多个 patch（16x16），再将每个 patch 投影为固定长度的向量送入 Transformer，后续 encoder 的操作和原始 Transformer 中完全相同。但是因为对图像分类，因此在输入序列中加入一个特殊的 token，该 token 对应的输出即为最后的类别预测。整个流程如图 5 所示。

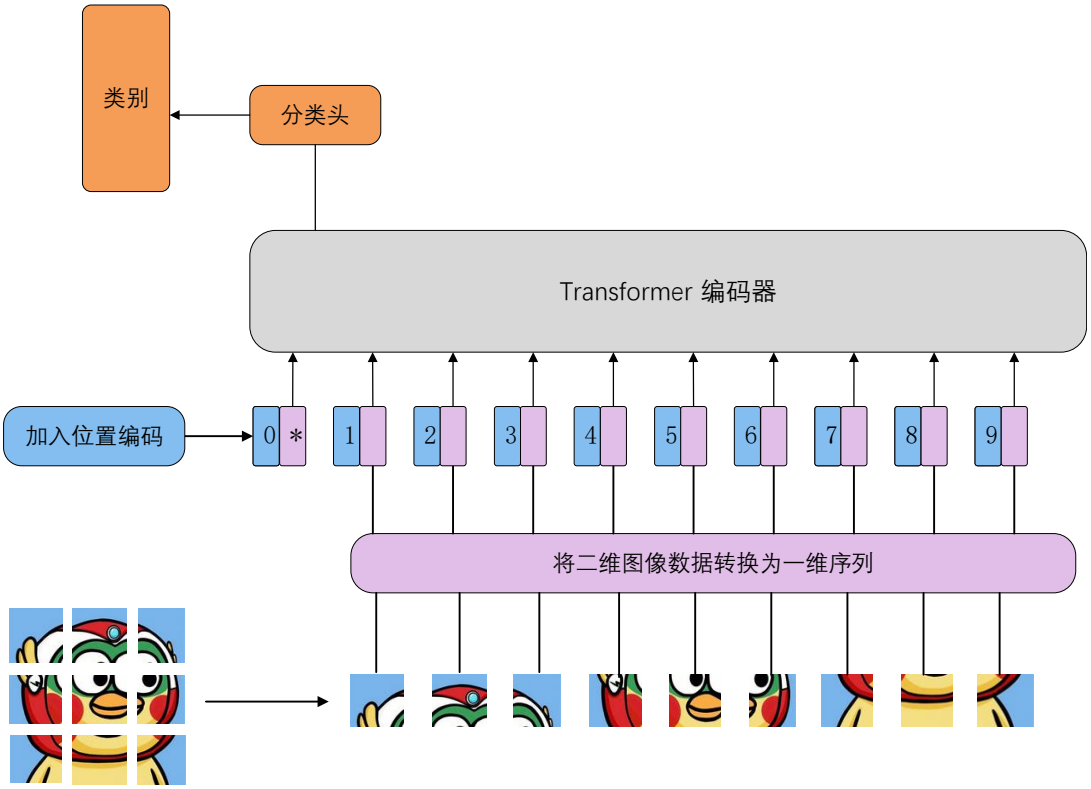

图 5: ViT（Visual Transformer）模型结构示意图。

(2)　文本编码器

RoBERTa [5] 是在论文 *RoBERTa: A Robustly Optimized BERT Pretraining Approach* 中被提出的。此方法属于 BERT 的强化版本，也是 BERT 模型更为精细的调优版本。RoBERTa 主要在三方面对之前提出的 BERT 做了该进，其一是模型的具体细节层面，改进了优化函数；其二是训练

策略层面，改用了动态掩码的方式训练模型，证明了 NSP（Next Sentence Prediction）训练策略的不足，采用了更大的 Batch Size；其三是数据层面，一方面使用了更大的数据集，另一方面是使用字节级别的 BPE（Bytes-level BEP ）来处理文本数据。

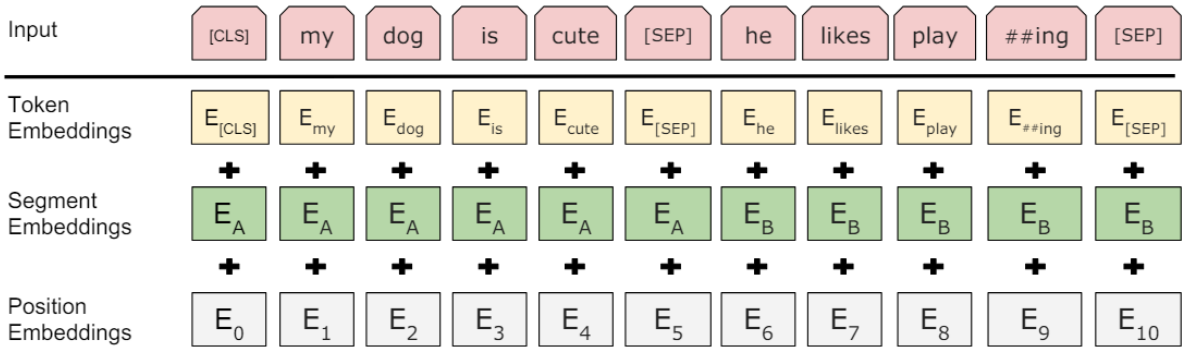

图 6: BERT（Bidirectional Encoder Representations from Transformers）模型输入示意图。

RoBERTa 在训练方面与原始 BERT 模型保持一致，使用类似“完形填空”的代理任务，让模型学习填补缺失词。如图 6 所示，输入句子中部分词被随机遮掩，替换为“[PAD]”，并在最前方添加一个特殊词汇“[CLS]”。对于缺失值填补任务，要求“[PAD]”的输出能还原原始词汇；对于分类任务，使用“[CLS]”的输出作为分类器的特征。

与图像编码器类似，我们使用基于中文数据预训练的 RoBERTa 模型 [6] 作为 CLIP 的文本编码器。该策略极大地提高了 CLIP 模型预训练与微调的效率，使得我们能在有限的训练回合中，获得更大的精度收益。

4. 训练目标

CLIP 使用图像文本对作为训练标签。这里举例一个包含 N 个图像文本对的训练 Batch，对提取的文本特征和图像特征进行训练的过程：

1. 输入图像 → 图像编码器 → 图像特征向量；输入文字 → 文字编码器 → 文字特征向量；并进行线性投射，得到相同维度；
2. 将 N 个图像特征和 N 个文本特征两两组合，形成一个形状为 $N \times N$ 的矩阵 s；CLIP 模型会预测计算出这 N^2 个图像文本对的相似度（即余弦相似度）；
3. 对角线上的 N 个元素因为图像-标签对应正确被作为训练的正样本，剩下的 $N(N-1)$ 个元素作为负样本；
4. CLIP 的训练目标是基于对比损失，最大化 N 个正样本的相似度，同时最小化 $N(N-1)$ 个负样本的相似度。

具体而言，对于任意一个图像文本对，CLIP 的对比损失如公式 (8) 所示。最后考虑所有可能存在的图像文本对，需要最小化的目标函数如公式 (9) 所示。其中，$s_{i,j}$ 表示相似度矩阵第 i 行的

第 j 列的元素，其数值含义为第 i 幅图像与第 j 段文本的相似度，由公式 (7) 计算。

$$l(i,j) = -\log \frac{\exp(s_{i,j})}{\sum_{k=1}^{N} \mathbf{1}_{[k\neq i]} \exp(s_{i,k})} \tag{8}$$

$$\mathcal{L} = \frac{1}{2N} \sum_{k=1}^{N} [l(k-1,k) + l(k,k-1)] \tag{9}$$

5. 模型推理

基于上述损失函数，我们在大规模数据集上对 CLIP 模型训练后，得到的权重可以用于推理。这里，我们以问题一的 I2T 任务为例，演示 CLIP 进行图像检索文本推理的过程。

(1) 数据预处理

设我们进行 I2T 任务的目标图像为 $Image_1$，潜在文本集合为 $\{Text_i \mid 1 \leq i \leq N\}$。模型首先对目标图像进行预处理，通过裁剪、缩放和插值操作，得到结构化的图像 $X \in \mathbb{R}^{1\times H\times W\times C}$。对于原始类别标签，我们将其拼接至模板“这是一张 {XX} 图像”中，使其更符合中文语法习惯，如图 7 所示。接下来，对拼接后的文本进行处理，将其分词后映射至词汇表，通过填充与截断操作控制序列长度，并转换为训练得到的词向量矩阵，得到结构化的文本 $Y \in \mathbb{R}^{N\times L\times D_{\text{emb}}}$。

(2) 图像文本编码

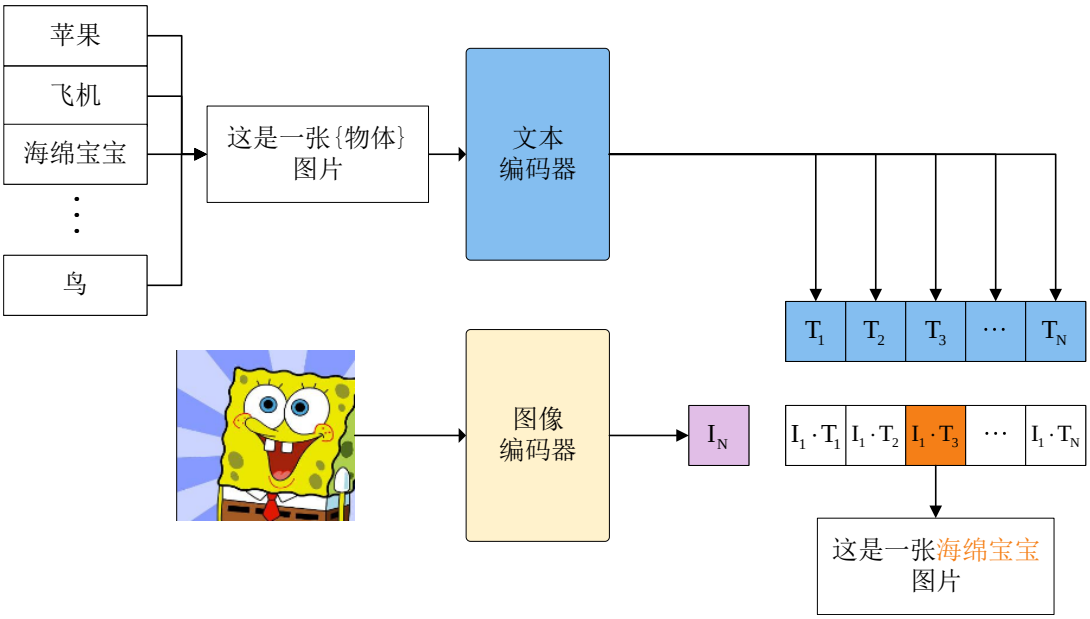

图 7: CLIP 模型推理——图像检索文本

对于结构化的图像与文本数据，使用 CLIP 的图像编码器与文本编码器，将其映射至同维度的编码向量。为了计算方便，我们对所有编码向量进行归一化，得到归一化图像编码 $I \in \mathbb{R}^{1\times D_{\text{model}}}$ 以及归一化文本编码 $T \in \mathbb{R}^{N\times D_{\text{model}}}$。对于任意图像 $Image_i$ 和文本 $Text_j$，其归一化编码的计算过程分别如公式 (10) 和公式 (11) 所示。通过进行归一化，我们可以使用向量的点乘操作来替代此前

的余弦相似度计算。

$$I_i = \frac{r_i}{\|r_i\|} \quad \text{其中，} r_i = \text{ImageEncoder}\left(Image_i\right) \tag{10}$$

$$T_j = \frac{h_j}{\|h_j\|} \quad \text{其中，} h_j = \text{TextEncoder}\left(Text_j\right) \tag{11}$$

(3)　相似度计算

接下来，对目标图像编码 I_1 与所有文本编码 T 进行点乘，得到目标图像与各段文本的相似度数组 $\text{Sims} = \{I_1 \cdot T_i \mid 1 \leq i \leq N\}$。最后，取相似度数组 Sims 的最大值，与之对应的文本即为最佳匹配文本。类似的，若取最大的 K 个值，则可得到最佳的 K 段匹配文本。图 7 演示了 CLIP 模型进行图像检索文本推理的完整流程。

6.　改进策略

(1)　累计梯度下降

(2)　掩码特征学习

我们借鉴了 FLIP 模型的图像掩码策略，对 CLIP 模型进行改进。我们首先将图像划分为不重叠的 patch，随机屏蔽掉大部分（例如 50% 或 75%）的 patch，图像编码器仅应用于可见 patch。

与原始的 CLIP 模型相比，改进后的模型在同样的训练时间内可以学习更多的图像-文本对，并且 ImageEncoder 的显存使用也下降（mask 掉 50%，显存消耗就下降 50%），这样在一定的硬件资源下就可以实现更大的 batch size，而对比学习往往需要较大的 batch size。同时，由于图像往往具有较强的冗余性，即使 mask 大部分图像，特征网络的学习性能并不会变差，反而由于更大的 batch size 而可以获得更好的性能。

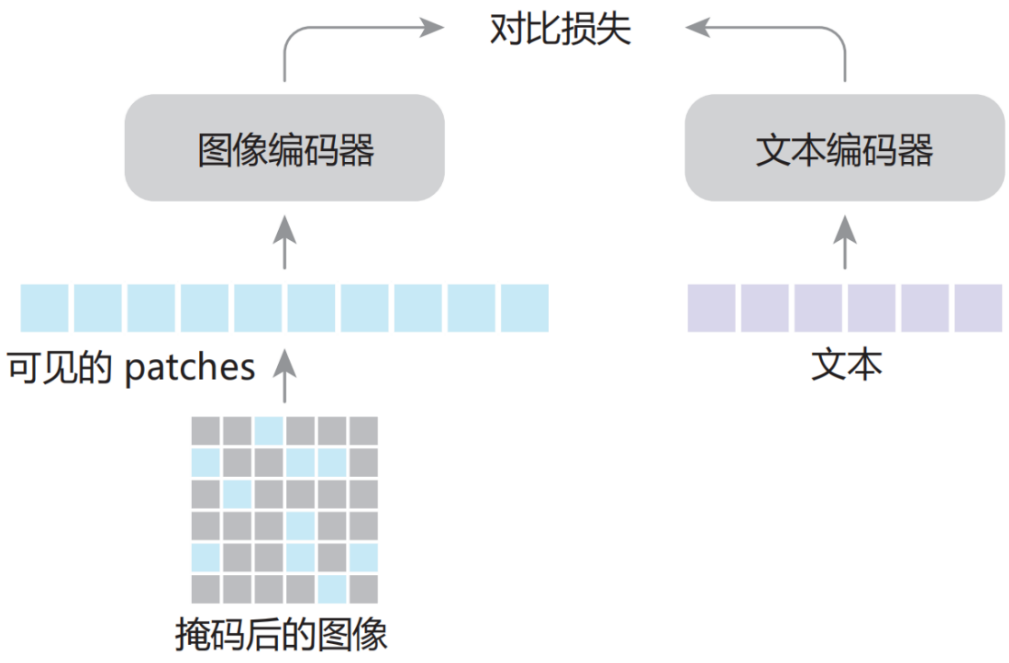

图 8: 掩码特征学习示意图。

(3)　FlashAttention

五、　实验与结果分析

1.　模型预训练

(1)　数据集构建

在预训练阶段，我们首先根据 MUGE Retrieval、Flickr30K-CN、COCO-CN 这三个中文检索方向常用数据集构建了约 800 万样本的数据集。为了确保数据集的质量，我们对数据集进行了彻底的清洗，移除了重复项、低分辨率图像以及包含错误或不完整文本的样本。此外，我们还执行了数据增强，图像包括随机裁剪、翻转、旋转，文本包括同义词替换、翻译互转等操作，以增加数据的多样性并提高模型对不同图像变换的鲁棒性。在数据清洗过程中，需要保持数据的多样性和代表性，确保模型能够学习到丰富的视觉和语言特征，同时使数据分布尽量平衡，防止模型模型出现偏向性。数据增强策略的实施，旨在模拟真实世界条件下图像可能出现的各种情况，以增强模型的泛化性。通过这种方式，能够提升模型对于图像几何变换的适应能力。

(2)　模型预训练

(3)　“零样本”测试

在仅基于公共数据集预训练，**不使用任何比赛数据集相关信息**的情况下，对模型进行性能评估，称为“零样本”测试。具体而言，我们按照章节 5. 中描述的方法，使用基于公共数据集预训练的模型，在比赛数据集的 **I2T 测试集**和 **T2I 测试集**上进行测试。

为了进行更全面的性能评估，我们除了计算题目所要求的 R@5 指标，还分别计算了 R@1、R@10 指标，以及各个指标的均值 MR。

图像检索文本　按照章节 5. 中描述的方法，我们使用基于公共数据集预训练的模型，在比赛数据集的 I2T 测试集上进行测试。

表 2: **“零样本”测试结果——图像检索文本**

回合 \ 评估指标	**R@1**	**R@5**	**R@10**	**MR**
1 epoch	0.3814	0.5800	0.6562	0.5392
2 epochs	0.3814	0.5800	0.6562	0.5392
3 epochs	0.3814	0.5800	0.6562	0.5392
4 epochs	0.3814	0.5800	0.6562	0.5392
5 epochs	0.5156	0.7094	0.7712	0.6654

文本检索图像

2. 模型微调

(1) 数据集构建

我们使用比赛附件 1 中所给的数据构建微调模型所使用的数据集，对数据集进行增强，如对图像进行随机裁剪、翻转、旋转，对文本进行同义词替换、翻译互转，人为地增加了数据的多样性，有助于模型学习到更加鲁棒的特征，从而在面对未见过的样本时能够做出准确的预测。在数据集准备好之后，我们将其划分为训练集、验证集和测试集，以便在训练过程中监控模型的性能，并最终评估其泛化能力。同时我们采用交叉验证的方法，以确保模型在不同的数据子集上都能保持一致的性能。

(2) 模型微调

(3) 微调结果测试

3. 结果分析

(1) 预训练结果

(2) 微调结果

(3) 有效性验证

参考文献

[1] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, “Learning transferable visual models from natural language supervision,” in *International conference on machine learning*, pp. 8748–8763, PMLR, 2021.

[2] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, and J. Tang, “Self-supervised learning: Generative or contrastive,” *IEEE transactions on knowledge and data engineering*, vol. 35, no. 1, pp. 857–876, 2021.

[3] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, “An image is worth 16x16 words: Transformers for image recognition at scale,” in *Proc. Int. Conf. Learn. Represent.*, 2020.

[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017.

[5] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “Roberta: A robustly optimized bert pretraining approach,” *arXiv preprint arXiv:1907.11692*, 2019.

[6] Y. Cui, W. Che, T. Liu, B. Qin, and Z. Yang, “Pre-training with whole word masking for chinese bert,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3504–3514, 2021.

[7] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 248–255, Ieee, 2009.

[9] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, “Loss functions for image restoration with neural networks,” *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 47–57, 2016.

[10] P. K. Diederik, “Adam: A method for stochastic optimization,” *(No Title)*, 2014.

[11] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” *arXiv preprint arXiv:1711.05101*, 2017.

[12] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel, “Handwritten digit recognition with a back-propagation network,” *Proc. Adv. Neural Inf. Process. Syst.*, vol. 2, 1989.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[14] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[15] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 770–778, 2016.

[16] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.