深圳技术大学
SHENZHEN TECHNOLOGY UNIVERSITY

本科毕业论文（设计）

题目：基于文本分类的专利数据分类方法研究

姓　　名	梁浩民
学　　院	大数据与互联网学院
专　　业	计算机科学与技术
学　　号	20200201021 5
指导教师	张博闻
职　　称	助理教授
提交日期	2024 年 5 月 15 日

深圳技术大学本科毕业论文（设计）诚信声明

本人郑重声明：所呈交的毕业论文（设计），题目《基于文本分类的专利数据分类方法研究》 是本人在指导教师的指导下，独立进行研究工作所取得的成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式注明。除此之外，本论文不包含任何其他个人或集体已经发表或撰写过的作品成果。本人完全意识到本声明的法律结果。

毕业论文（设计）作者签名：梁浩民

日期：　2024 年 5 月 15 日

目 录

基于文本分类的专利数据分类方法研究

【摘要】在大数据与人工智能时代，半导体技术的创新和应用成为推动经济和科技进步的关键力量。面对全球半导体产业发展的日新月异，2023 年我国国务院强调了专利数据应用的重要性，注重专利成果的转化和半导体产业的发展，从而增强国家核心竞争力。本研究基于半导体产业专利数据，旨在实现专利数据与半导体产业链上各子领域的精准匹配,从而推动专利成果的转化。主要工作如下：

（1）构建一个中文产业链专利文本数据集（CN-ICPC），该数据集包含 17 万条半导体领域专利文本数据,采用四层结构标签,旨在细化专利数据的分类精度。在数据预处理阶段，我们采用了去停用词和分词技术，并且根据其专有名词训练词向量，以优化数据集的质量和分类效果。

（2）使用多个层次多标签文本分类模型，包括基于层次感知的全局模型、基于信息最大化的全局模型和基于层次感知的标签语义匹配网络模型,并比较了这些模型在不同数据集上的性能。实验结果表明，所有模型在 CN-ICPC 数据集上的 Micro-F1 值均达到 90%以上，超过另外两个经典的数据集的分数，验证了这些模型的有效性和 CN-ICPC 数据集的质量。

【关键词】文本分类；专利分类；层次多标签；产业链；深度学习

Research on Patent Data Classification Method Based on Text Classification

【**Abstract**】In the era of big data and artificial intelligence, the innovation and application of semiconductor technology have become key forces driving economic and technological progress. Faced with the rapid development of the global semiconductor industry, the importance of applying patent data was emphasized by the State Council of China in 2023, focusing on the transformation of patent achievements and the development of the semiconductor industry to enhance the country's core competitiveness. This study, based on semiconductor industry patent data, aims to achieve precise matching between patent data and various subfields of the semiconductor industry chain to promote the transformation of patent achievements. The main work is as follows:

(1) Construct a Chinese Industrial Chain Patent Corpus (CN-ICPC), which contains 178,118 patent text data in the semiconductor field, using a four-layer structure label to refine the classification accuracy of patent data. In the data preprocessing stage, we used stop-word removal and word segmentation techniques, and trained word vectors based on proprietary terms to optimize the quality and classification effects of the dataset.

(2) Employ multiple hierarchical multi-label text classification models, including Hierarchy-Aware Global Model, A Global Model for Hierarchical Text Classification via Information Maximization, and Hierarchy-aware Label Semantics Matching Network for Hierarchical Text Classification, and compared the performance of these models on different datasets. Experimental results show that all models achieved a Micro-F1 value of over 90% on the CN-ICPC dataset, surpassing two classic datasets, verifying the effectiveness of these models and the quality of the CN-ICPC dataset.

【**Key words**】 Text classification; Patent classification; Hierarchical multi-label; Industrial chain; Deep learning

1. 引言

1.1 研究背景和意义

在当前知识爆炸性增长的时代，专利数据作为重要的知识资产，在推动科技创新、产业升级、增强国家实力和企业竞争力等方面发挥着重要作用。随着科技的不断进步，专利数据已成为了科技创新的关键组成部分，它蕴含着丰富的科研成果、技术趋势和产业动向。在 2023 年，我国国务院办公厅印发的《专利转化运用专项行动方案（2023—2025 年）》提出了加强专利数据应用的重要任务和指导意见，明确了推动专利数据高效利用、加强知识产权保护和运用、促进科技成果转化的目标。因此，深入分类和分析专利数据在产业链中的应用现状，具有战略上的重要意义。

然而，目前仍缺少针对中国专利数据对应半导体产业链的细粒度分类，海量的半导体专利数据难以有效利用。在这一背景下，我们迫切需要解决的问题是建立高质量的中文产业链专利文本数据集，大规模覆盖半导体产业各个子领域，并运用人工智能领域的自然语言处理技术对其进行处理和分析。

我们将专利转化运用产业链问题建模为层次多标签文本分类（Hierarchical Multi-label Text Classification，HMTC）问题，致力于研究基于层次多标签文本分类的专利数据分类方法，旨在将专利数据与半导体产业链上下游进行细致匹配，并根据半导体产业细分领域向半导体企业进行精准推送，以实现专利数据的精准应用。具体而言，我们将关注于细粒度、层次多标签文本分类方法的设计与实现，以更好地满足半导体产业领域对专利数据的需求。通过给专利文本标注细粒度的、层次化的标签，构建高效的文本分类模型，不仅可以提高专利数据的利用效率，推动国家和企业的半导体科技创新和半导体产业发展，还能够实现细化匹配和精准推送，有助于加强专利转化运用，促进科技成果向生产力转化，进而推动经济社会的可持续发展。由此，我们基于文本分类的专利数据分类方法研究具有重要的理论和实践意义。

1.2 相关研究现状

1.2.1 层次多标签文本分类

文本分类问题是自然语言处理领域非常经典的研究问题，如果数据包含多个标签类别，对这些标签进行预测就是一个多标签文本分类问题。生活中，存在某些数据对象不仅仅是含有多个标签，而且标签之间还具有一定的层次结构，这种特殊的多标签文本分类问题被称为层次多标签文本分类问题。层次多标签文本分类问题引入了层次结构来组织标签体系，如图 1-1 所示。我们需要预测给定标签层次结构中的多个标签，这些标签通常按照自上而下的方式构建一条或多条路径。细粒度标签能准确描述输入文本，而粗粒度标签则代表着更通用的概念。

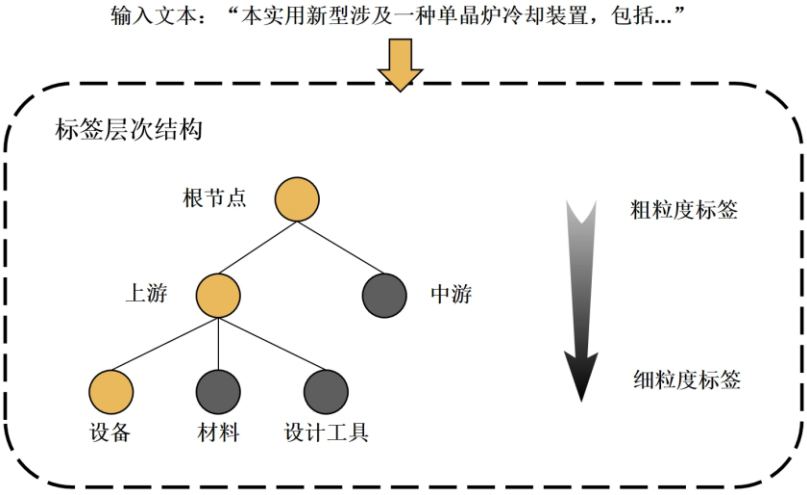


图 1-1 一个从粗粒度标签到细粒度标签的层次文本分类示例

层次多标签问题有 2 类表示方法，树形结构（Tree）或有向无环图结构（DAG）。如图 1-2 所示，假设把一个标签看作是一个子类（Subclass），那么树形结构和有向无环图结构的表示方法区别在于：树形结构的每一个标签节点有且仅有一个父节点，而对于有向无环图结构来说，每一个标签节点可以有多个父节点（如图 1-2（b）所示）。Wu 等[1]定义了层次分类：树形结构的常规概念层次定义在一个偏序集（C，$\prec$），其中 C 表示一个有限集合，$\prec$表示“IS-A”的偏序关系，即$c_x \prec c_y$表示c_x属于c_y，c_y是c_x的父类，c_x是c_y的子类。Wu 等将“IS-A”的关系定义为既是反自反的，又是传递的，我们可以将“IS-A”的关系定义为反对称的、反自反的和传递的：

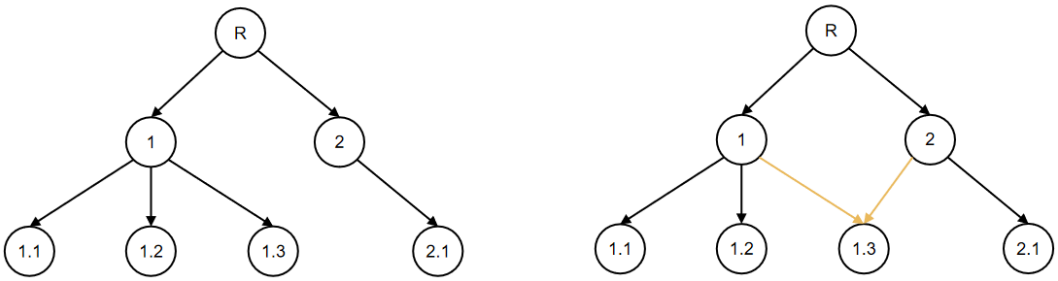
唯一的最大元素“R”是树形结构的根节点（Root）；

- 反对称：$\forall c_x, c_y \in C$, 如果$c_x \prec c_y$, 那么$c_x \not\prec c_y$；
- 反自反：$\forall c_x \in C$, $c_x \not\prec c_x$；
- 传递：$\forall c_x, c_y, c_z \in C$, 如果$c_x \prec c_y$且$c_y \prec c_z$ 那么$c_x \prec c_z$；

将 HMTC 问题抽象为根据一组文档D与对应的标签结构γ，通过对一个分类器或模型Ω进行训练，使其能预测未见过的文档的标签L，可以定义为公式：

$$\Omega(D;\gamma,\theta) \rightarrow L \tag{1-1}$$

其中θ是模型Ω训练得到的参数。



(a) 树形结构　　(b) 有向无环图结构

图 1-2 两种标签层次结构示意图

由于层次多标签文本分类是多标签文本分类的一种特殊情况，因此将其最底层的标签节点进行平铺，进而转换为多标签分类问题，只要正确预测最底层标签，按照层次约束规则，其所有上层的标签节点都被自动归类。如图 1-3，这种平面的方法不仅没有考虑层次标签之间的关系，也没有考虑处于同一层不同标签的关系。如果考虑利用标签关系和层次标签间的关系，有局部的方法和全局的方法。

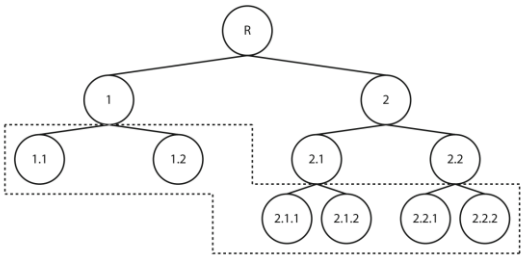


图 1-3 平面方法示意图[3]

局部的方法被称作“自下而上（bottom-up）”的方法，它利用了“分治”的思想，通过解决局部子问题（如逐个标签节点或逐层构建分类器），再将分类结果组成最终全局的分类结果[2]。如图 1-4 所示，每个圆形表示一个标签节点，每个虚线区域表示一个分类器，局部的方法主要可以分 3 种不同的建模策略[3]：每个节点一个局部分类器（a local classifier per node，LCN）、每个父节点一个局部分类器（a local classifier per parent node，LCPN）和每层节点一个局部分类器（a local classifier per level，LCL）。

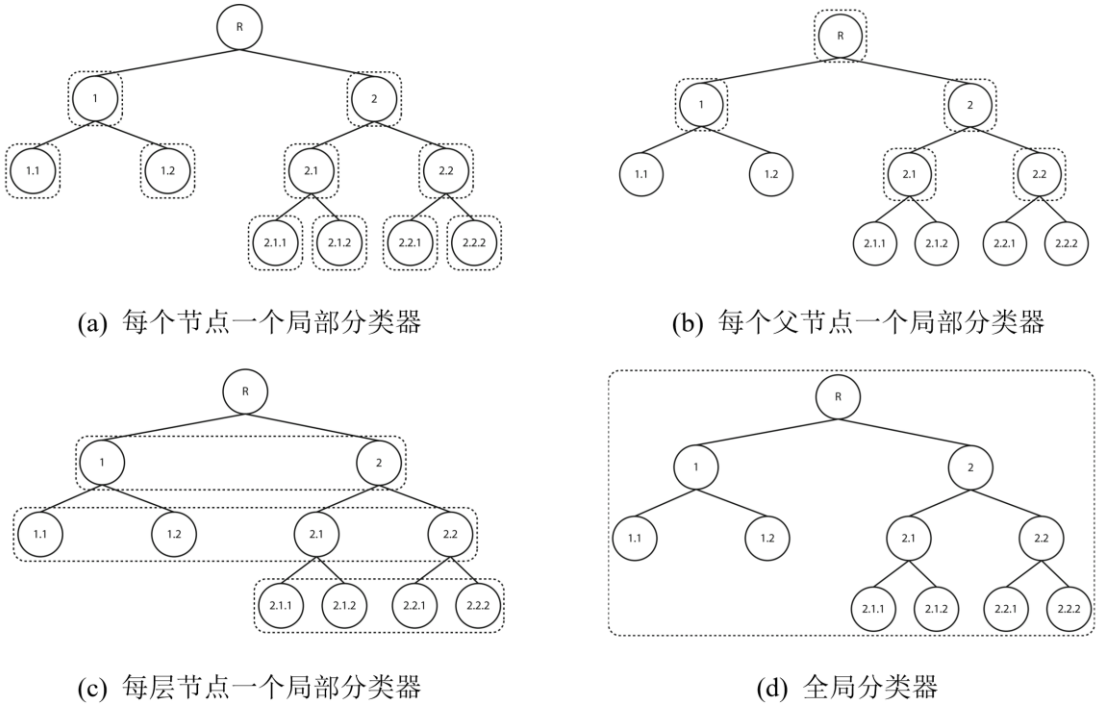


图 1-4 局部方法与全局方法示意图[3]

全局的方法被称作“大爆炸（Big-bang）”的方法，它考虑在层次结构上只构建一个分类器，并且需要同时利用整个标签结构。通常，有些传统的全局方法会参考平面的方法，所以对局部层次信息学习不足。近年来，深度学习模型对层次信息的提取能力比传统方法更强，因此基于深度学习的全局方法成为主流。

1.2.2 深度学习文本分类

在 2006 年，Hinton 等[4]首次提出深度神经网络（DNN），展示了 DNN 在语音与图像识别领域的优越性。相较于传统神经网络，DNN 能更有效地处理大规模数据，并从中提取更丰富的特征。这一重要里程碑标志着深度学习正式的诞生。

在 2011 年，Hinton 等人[5]开始在文本分类应用了循环神经网络（RNN）。然而 RNN 受限于梯度消失和梯度爆炸问题，限制了其在长序列文本数据上的表现。为此，在 2012 年，长短期记忆网络（LSTM）[6]应运而生，其门控机制使得网络能够更好地捕捉长距离依赖关系。2014 年，Kim[7]在文本分类任务中引入了卷积神经网络（CNN），其利用多个卷积核在文本表示上进行卷积操作，通过最大池化操作，使得模型能够有效地捕捉不同长度的局部和全局信息，从而在文本分类任务中取得了优异的表现。CNN 的突破性应用不仅加速了文本分类任务研究的发展，还促进了对于不同神经网络结构的深入探索，如 RNN 与 CNN 的结合[8]。2018 年，Google 公司提出的 BERT（Bidirectional Encoder Representations from Transformers）模型[9]通过大规模无监督训练，在预训练阶段获取了更深度的语言表示，具备了更强的泛化能力与语义理解能力。BERT 的强大表现推动了预训练技术在文本分类任务的广泛应用，是 NLP 领域的重要里程碑。

在多标签文本分类领域的研究中，随着编码技术的不断优化，新的模型能够更有效提取文本特征和精确地建模标签间的关系。2017 年，Chen[15]等研究者发表了 CNN-RNN 模型，其结合了 CNN 与 RNN 各自的优势，有效地提取、分析文本的全局语义与局部语义。模型通过 CNN 去识别关键的视觉特征，再用 RNN 的序列处理进行标签预测，深入解析层次标签间的复杂关系。2018 年，Yang[16]等提出 SGM（Sequence Generation Model for MLC）模型，利用序列生成（Seq2Seq）思想，将 HMTC 问题视为序列生成问题，从而考虑标签之间的关联性。该模型在解码器部分中得到了改进，它不仅可以有效地捕捉不同标签之间的关联性，还能通过注意力机制自动提取输入文本中的关键信息，在预测不同标签时，生成上下文向量，并筛选出与各标签最有贡献度的词。

胶囊网络（Capsule Network）由 Hinton[17]等提出，采用神经元向量去替代传统神经网络中的单一神经元节点，并通过动态路由（Dynamic Routing）的方法进行训练。Aly[18]等进一步将胶囊网络应用于 HMTC 任务，通过对比实验说明浅层胶囊网络结构也比 CNN、LSTM 以及 SVM 能力更强，证明模型在处理具有结构复杂的类别和罕见事件上具有优势，尤其是胶囊网络在整合潜在编码信息方面的能力。

2. 相关技术理论

2.1 文本向量化表示

2.1.1 Word2Vec 模型

Word2Vec（Word to Vector）模型是由 Google 的 Mikolov[19]等人于 2013 年提出的一种训练词嵌入（word embeddings）的方法，Word2Vec 主要用于预训练阶段，使用无监督学习的方式，对给定的语料库使用特定窗口大小扫描每个句子，捕捉窗口内的关键词及其文本上下文，通过计算关键词条件下的文本上下文中的词汇和短语的条件概率，来推算出文本之间的关联度。Word2Vec 模型的词向量映射机制直接与词义相关联，因此 Word2Vec 能够将语义相似的词汇映射至相近的向量空间中，进而有效推断出词汇之间的语义联系。

Word2Vec 的架构包括两种子模型：CBOW 子模型和 Skip-gram 子模型。这两个子模型都包括三层神经网络分别是输入层、隐藏层与输出层。CBOW 模型的输入是独热编码的文本信息，在输出层计算每个节点的输入，进行中间词语的预测任务。而 Skip-gram 模型使用另一种输入方式，在隐藏层中对输入数据进行线性运算后，进行中心词的预测，由此来推断上下文词语。CBOW 和 Skip-gram 模型架构如图 2-1 所示。

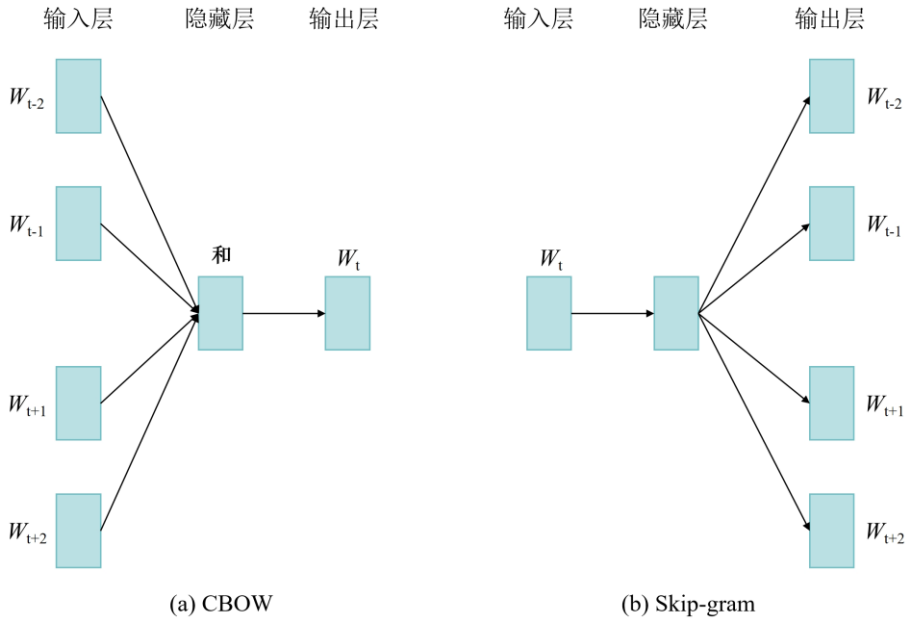


图 2-1 CBOW 和 Skip-gram 模型架构

将一句话假设为：$W_1...W_{t-2}W_{t-1}W_tW_{t+1}W_{t+2}...W_T$，当窗口大小为 2 时，CBOW 用周围词$W_{t-2},W_{t-1},W_{t+1},W_{t+2}$去预测中心词$W_t$，而 Skip-gram 则是用中心词$W_t$去预测周围词$W_{t-2},W_{t-1},W_{t+1},W_{t+2}$。

CBOW 模型的详细推导过程如下：

$$P(context|W_t)=p(W_{t-c},W_{t-c+1},...,W_{t-1},W_{t+1},...,W_{t+c-1},W_{t+c}|W_t) \quad (2\text{-}1)$$

其中，$context$表示上下文，W_t表示中心词，c表示窗口大小范围。

Skip-gram 模型的详细推导过程如下：

$$P(W_t|context)=p(W_t|W_{t-c},W_{t-c+1},...,W_{t-1},W_{t+1},...,W_{t+c-1},W_{t+c}) \quad (2\text{-}2)$$

其中，$context$表示上下文，W_t表示中心词，c表示窗口大小范围。

2.1.2 GloVe 模型

GloVe（Global Vectors for Word Representation）是一种无监督的、词嵌入的模型，它是由斯坦福 NLP 实验室的 Jeffrey Pennington[20]等提出。GloVe 在 Word2Vec 的基础上进行了创新，引入了全局词频的概念，并且将语料库中每一个词汇转换为向量空间中的点，通过这些点之间的空间距离和方向性来体现词与词之间的语义和语法关系。

GloVe 模型是对数据的全局范围中进行统计词与词之间的共现概率，采用矩阵分解技术实现词嵌入。首先需要构建一个“词-词共现矩阵”，矩阵中的元素X_{ij}表示词j在词i所在上下文中出现频次。接着，通过一种特定的加权最小二乘法对该矩阵进行分解，产生低维度的词向量，公式 2-3 为详细计算过程。

$$J=\sum_{i,j=1}^{V} f(X_{ij})\left(w_i^T\tilde{w}_j+b_i+\tilde{b}_j-\log(X_{ij})\right)^2 \quad (2\text{-}3)$$

其中，V表示词汇表，$f(X_{ij})$表示加权函数，w表示词向量，b表示偏置。

2.2 神经网络模型

2.2.1 卷积神经网络

1979 年，日籍研究者福岛邦彦首次提出用于模式识别的神经网络模型：Neocognitron[21]，该模型是首次使用卷积层与下采样层（池化层）的神经网络。1998 年，Yann LeCun 及其团队[22]开发的 LeNet-5 卷积神经网络，成功解决了手写数字识别，并被多家银行采用于取款机的数字识别系统中。

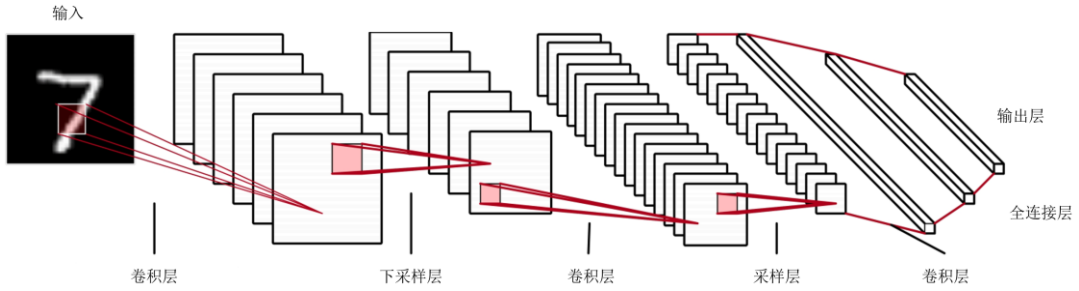


图 2-2 LeNet-5 结构图

卷积神经网络（CNN）是专门用于处理规则性网格数据的一种前馈式神经网络。其特点是用卷积运算，而不是矩阵乘法。卷积运算是一种积分变换的数学方法，可以稀疏交互，输入数据的尺寸大于卷积核的尺寸。在图像处理领域，CNN 在捕捉图像的平衡不变性方面表现出色，即使图像发生扭曲或位移，卷积结构仍能有效提取出相似的特征，增强了网络的鲁棒性。在文本处理领域，CNN 能够对文本进行编码并提取语义特征。通过将文本转换为词向量矩阵，无论采用随机嵌入或其他词嵌入方法，在词向量矩阵上执行卷积操作，不同的卷积核能在不同大小的窗口中提取文本特征。而池化层能够减小模型规模，并筛选特征，提高特征的鲁棒性。

另一方面，CNN 主要捕捉局部特征，由于卷积核的视野有限，难以全面捕捉长文本中的上下文关系。对于长文本，通常采用循环神经网络。

2.2.2 循环神经网络

循环神经网络（RNN）由多个神经元单元构成，这些单元串行处理序列化、可变的数据（如音频、视频、文本等），如图 2-3 所示。

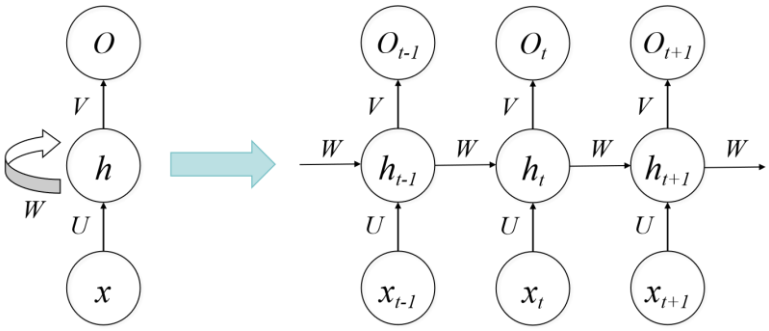


图 2-3 RNN 结构图

与 CNN 相比，RNN 主要优势在于其能够维持短期记忆，能够识别输入与输出序列之间的隐含关系，因此有效地处理序列化数据。RNN 在每个计算步骤中产生的输出与输入数据密切相关，模型的隐藏状态受到当前输入及其前单元状态的影响。RNN 的计算从序列的起始状态开始，一直到终点，其隐藏状态的演化及输出的推导公式如下。

$$h_t = \tanh（Wh_{t-1} + U_{x_t} + b）\tag{2-4}$$

$$O_t = Vh_t + c\tag{2-5}$$

其中，U，V 和W 表示偏移向量，b 和c 为神经元的偏差。

2.2.3 长短时记忆神经网络

长短时记忆神经网络（Long Short-Term Memory, LSTM）[28]基于 RNN 网络，加入了“门”机制去有选择性地对数据信息进行保留和遗弃等。其计算公式如下：

$$i_t = \sigma(W_{ii}x_t + b_{ii} + W_{hi}h_{t-1} + b_{hi})$$

$$f_t = \sigma(W_{if}x_t + b_{if} + W_{hf}h_{t-1} + b_{hf})$$

$$o_t = \sigma(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho})\tag{2-6}$$

$$\tilde{c}_t = \tanh(W_{ic}x_t + b_{ic} + W_{hc}h_{t-1} + b_{hc})$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$$

$$h_t = o_t \odot \tanh(c_t)$$

其中，W 为模型参数，b 为偏置，i_t 为输入门，f_t 为遗忘门，o_t 为输出门，h_t 和 c_t 为某 t 时间隐藏层和记忆细胞的状态，$\odot$ 表示乘积。

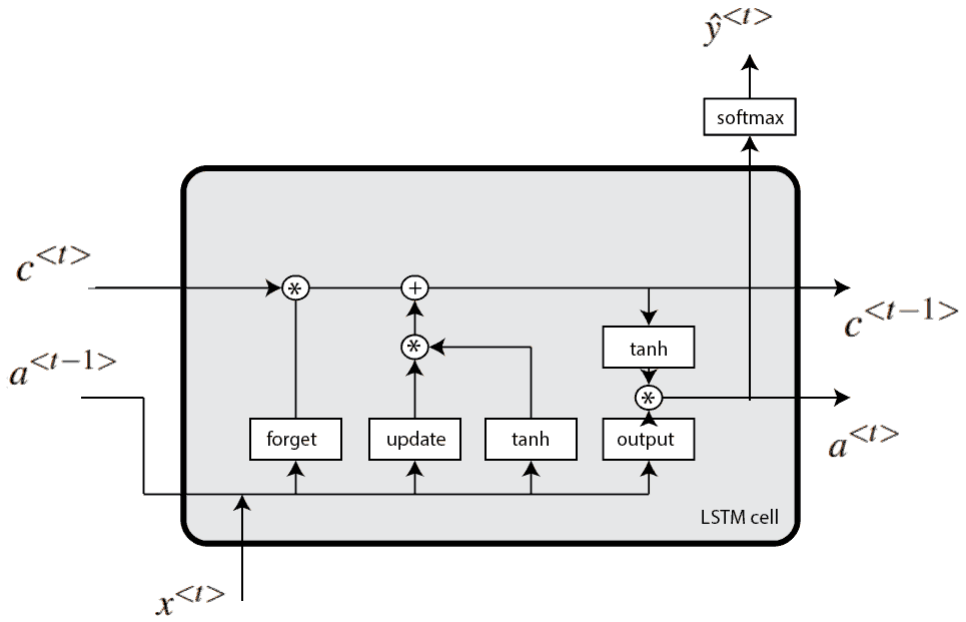


图 2-4 LSTM 示意图[28]

2.2.4 图卷积神经网络

图卷积网络（Graph Convolutional Network，GCN）[29]是针对图形结构数据处理的神经网络。该网络通过在图的各个节点上执行卷积操作，有效地学习节点的特征表示。GCN 专适用于处理数据点间存在复杂关系和依赖性的应用场景，如社交网络分析、知识图谱推理等。

GCN 的基本原理是利用某节点及其邻居的信息来更新节点的特征表示。在每个卷积层中，节点新特征是通过汇聚其自身特征与邻居特征计算得出的。这一汇聚机制允许每个节点去捕获其在图中的局部邻域信息。随着图卷积层数量增加，节点的特征表示会逐渐融合更广泛的邻域信息，进而理解图结构更深层次的特征。

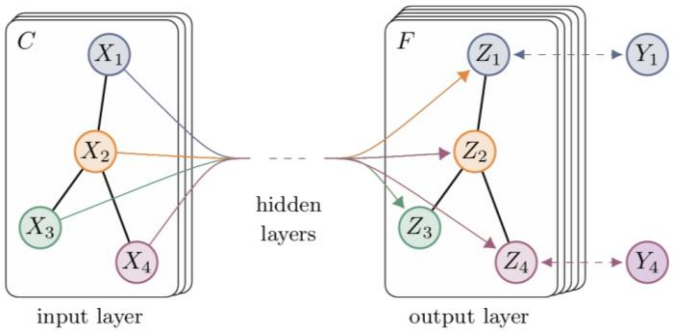


图 2-5 图卷积神经网络示意图[29]

2.3 注意力机制

注意力机制的灵感源自人类生活中处理信息的方式，当人类在处理杂乱的信息时，人脑不会均等分配注意力，而是集中于关键信息。[30]这一机制显著提升了人类处理信息的效率，在计算机视觉和 NLP 领域中有着非常好的效果。

注意力机制的能够形象地简述成：一个查询矩阵Q，表示我们想去理解的词语；键K和值V，表示数据所有词的向量。当计算Q与K之间的相似度后，我们获得相似性得分，这些得分构成了注意力权重。然后，系统根据这些得分对V进行加权求和，以此形成Q在V上的加权表达，计算公式如下。

$$Attention(Q,K,V)=\sum similarity(Q,K)\cdot V \tag{2-7}$$

其中，Q、K和V均为矩阵或向量，函数$similarity(Q,K)$用于计算Q与K之间的相似度

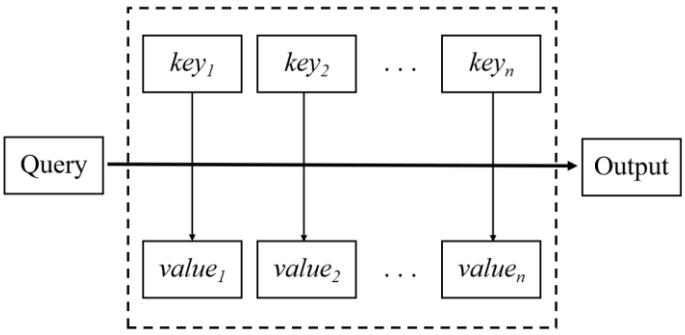


图 2-6 注意力机制原理示意图

3. 中文产业链专利文本数据集的构建

在解决现实世界的问题时，深度学习所需的数据集和模型都是十分关键的。数据集质量和多样性直接影响模型性能，因此，我们需要准确地收集真实、全面的数据。产业链专利层次多标签分类需要大规模的数据，但目前公开的中文专利数据集大多都是单标签的，标签划分主要按照国际专利分类（International Patent Classification, IPC）或中国专利法（只有 3 种类型分别是：发明、实用新型、外观设计），没有出现细粒度的、层次化的、将专利数据与产业链上下游进行细致匹配的数据集。由此，本文决定进行构建一个多标签中文产业链专利文本数据集。

3.1 专利数据集调研

专利分类是 NLP 文本分类任务重要的分支，目前存在多个主流的专利文本数据集，以下是这些经典数据集的基本内容。

- WIPO-alpha

WIPO-alpha[10]数据集收集了出版时间从 1998 年至 2002 年之间的按照澳大利亚专利分类的专利文件，包含了约 4.6 万个训练集和 2.9 万个测试集，从训练集中随机选择 20%的数据点作为验证集。其中在子类别级别，训练集中有 602 个标签，测试集中有 576 个标签。数据集中的 IPC 代码使用 IPC 的第七版定义，每个专利都标有一个主 IPC 代码和一组次要 IPC 代码。

- CLEF-IP (2011)

CLEF-IP (2011)[11]数据集主要包含了欧洲专利局（European Patent Office，EPO）发布的专利文件与世界知识产权组织（WIPO）发布的专利文件，约 150 万专利，350 万 XML 文件。每个专利文件可以是申请文件、搜索报告或已授权的专利文件。该数据集常被用于检索和分类任务。

- USPTO-2M

USPTO-2M[12]的原始专利数据来自美国专利商标局（US Patent and Trademark Office，USPTO）的网站，后缀“2M”的意思是数据量在 200 万条左右，其包含了从 2006 年至 2015 年这 10 年之间的 637 个子类别的实用专利文件。

- BIGPATENT

BIGPATENT[13]数据集是来自 Google 专利公共数据集，包括自 1971 年至 2018 年的 130 万份美国专利文件记录和人工编写的摘要。这些专利文件可以细分到 9 个技术领域，主要构建的是一个专利摘要语料库。相比以往的数据集，这些摘要中包含较少且较短的抽取式短语，并且具有更丰富的话语结构。

- HUPD

HUPD（Harvard USPTO Patent Dataset）[14]数据集是一个大规模、结构好、多用途的语料库，涵盖了 2004 年至 2018 年间提交给美国专利商标局（USPTO）的英语专利申请，包含超过 450 万份专利文档，比同类语料库多出两到三倍。与此前的专利数据集不同，HUPD 所涵盖的是发明人提交的专利申请版本，而非已获得专利的最终版本。以往的专利数据集只包括专利的一两个数据特征（如描述和摘要），但 HUPD 包含了 34 个特征，包括申请日期、细粒度分类代码、审查员信息等。因此，它不仅可以用于文本分类问题，还可以执行新的 NLP 任务。

通过大量调研公开的专利数据集，发现不仅缺少大规模的、高质量的中文专利文本数据集，还缺少针对产业链的层次多标签分类数据集。由此，本文针对中文专利与产业链构建了一个多标签中文产业链专利文本数据集。

3.2 产业链专利数据集简介

3.2.1 半导体产业链

伴随着智能化迅速发展，半导体产业链的越加成熟，其主要由三个部分构成：上游的支撑产业、中游的制造产业和下游的应用产业。产业链可再进一步分多个细分领域，如图 3-1 所示。上游的支撑产业主要是半导体的设备、材料和设计工具，半导体的设备包括晶圆制造设备、硅片制造设备、封装设备等；半导体的原材料包括晶圆制造材料和封装材料；半导体的设计工具包括 EDA 设计工具、ARM 设计工具和 IP 核。中游的制造产业主要是半导体的制造、设计和测封，半导体的制造有模拟芯片制造、光电器件、数字芯片制造等；半导体的设计有存储芯片

设计、通讯芯片设计、微处理器芯片设计等；半导体的测封有封装和测试。而下游的终端应用拉动了整个半导体产业链，如新能源汽车、人工智能相关的新浪潮。

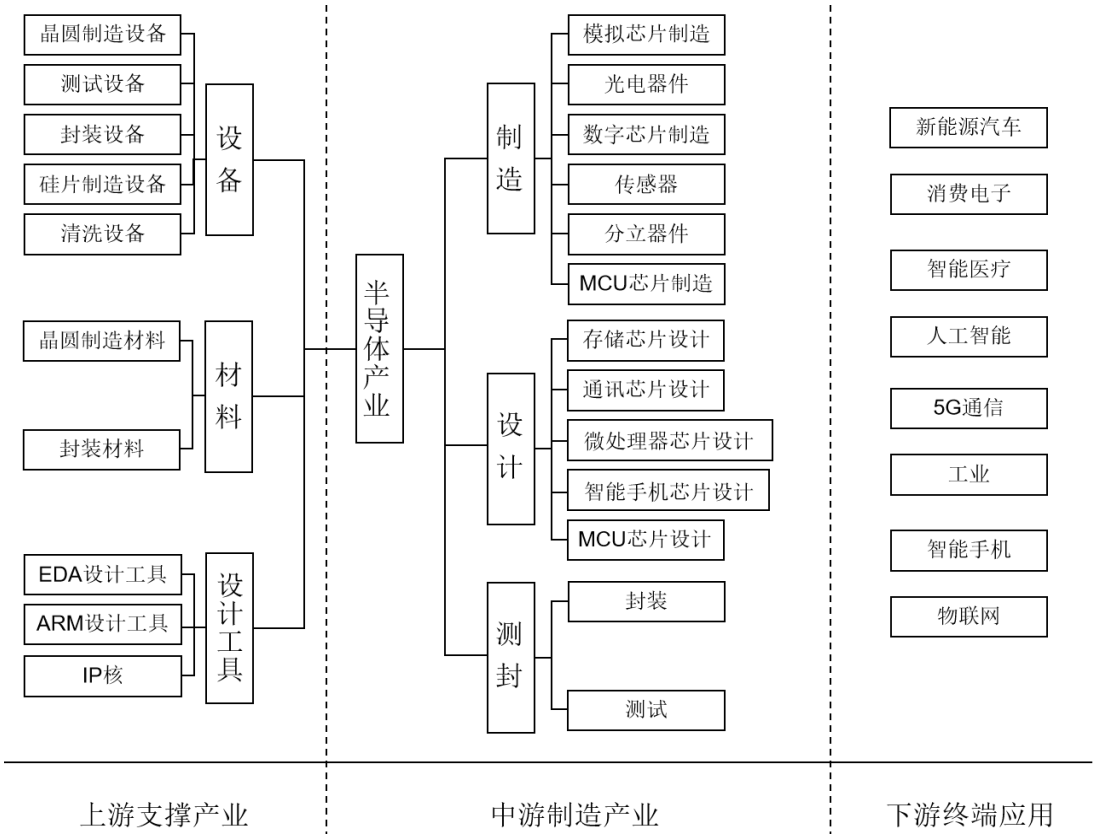


图 3-1 半导体产业链上下游概略图

3.2.2 数据集内容

本文所使用的专利数据是与北京大学（深圳）研究院团队合作收集的一共 906,870 项的关于半导体行业的专利文本数据，包含专利的摘要和与之对应的产业链主题，采用 4 级标签结构，第一级有 2 类标签，第二级有 6 类标签，第三级有 24 类标签，第四级有 91 类标签，标签总数达到 123 类。第一级标签是上游或中游，第二级至第四级标签则是具体细分的半导体产业链领域。表 3-1 是从原始数据集中截取的部分有代表性的数据，数据的属性依次是“一级”、“二级”、“三级”、“四级”、“专利名称”、“申请人”、“专利类型”和“摘要”，基本囊括了一个半导体专利的大部分信息，其中申请人有不少国际知名企业如华为、台积电、韩国三星、荷兰 ASML 公司、日本电工株式会社等。因为是基于中文文本，我们将数据集命名为 CN-ICPC（Chinese Industrial Chain Patent Corpus）。

表 3-1 原始数据集实例

一级	二级	三级	四级	专利名称	申请人	专利类型	摘要
上游	设备	测试设备	激光器	垂直腔面发射激光器 TO 同	中国科学院半导体研究所	发明专利	一种垂直腔面发射激光器的 TO 同…
上游	材料	晶圆制造材料	光刻胶	一种光刻胶的去除方法	京东方科技集团有限公司	发明专利	本发明提供了一种光刻胶的去除方…
中游	制造	模拟芯片制造	射频芯片制造	射频感知方法及相关装置	华为技术有限公司	发明专利	本申请涉及无线通信领域，尤其涉…
中游	设计	存储芯片设计	DRAM 芯片设	具有统一的存取执行时间的	睦塞德技术公司	实用新型专利	一种动态随机存取存储器(DRAM…
中游	制造	模拟芯片制造	射频芯片制造	射频连接器组合	富士康（昆山）有限公司	发明专利	一种射频连接器组合，包括相互对…
上游	设计工具	EDA 设计工具	EDA 设计工具	布线电路基板	日东电工株式会社	发明专利	本发明提供一种布线电路基板，其…
中游	设计	通讯芯片设计	通讯芯片设计	基于正交频分复用的通信系	三星电子株式会社	发明专利	一种基于 OFDM 通信系统中的同…
中游	封测	封装	封装	多芯片半导体封装件	台湾积体电路制造有限公司	发明专利	半导体封装件包括：第一管芯；…

3.3 数据预处理

对于收集到的原始专利文本数据集，其中含大量的噪声（如格式错误，缺失数据，重复数据等），需要对数据进行预处理。数据的预处理流程如图 3-2 所示。

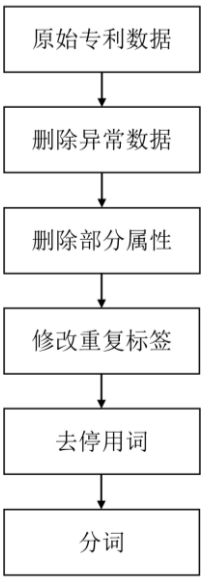


图 3-2 数据预处理的流程图

3.3.1 删除异常数据

原始的专利文本数据中存在一定量的“脏数据”，包括缺失数据、重复项、格式错误等问题。本文对这些“脏数据”进行删除处理后，数据量从 906,870 项变为 178,118 项数据。

3.3.2 删除部分属性

由于 CN-ICPC 数据集的部分属性如“专利名称”存在大量重复，“申请人”和“专利类型”属性收集不完整且对文本分类的作用很小，因此予以删除处理。

3.3.3 修改重复标签

我们对数据的 4 级标签结构进行统计后，发现 2 组前 3 级标签不同而第 4 层级标签名称重复的情况，为了保证标签数据的“互斥”，将每组的其中一类标签归到另一类同名标签下，修改其前 3 级标签。此外，还存在 13 组第 3 级标签和其属下第 4 级标签名称重复的问题。本文将这些父子标签进行添加数字后缀，第 3 级的标签添加后缀数字“1”，第 4 级的标签添加后缀数字“2”，以便区分。

3.3.4 去停用词

停用词是指频繁出现的但无实际意义的功能词，例如连词、介词、代词，语气词、冠词等，这些词在文本分类任务中对文本含义的理解几乎没有贡献，且占用计算和存储资源。本文考虑予以删除操作。我们使用了一份开源的中文停用词表，共 1893 个停用词，其中也包括一些标点符号及特殊符号。

3.3.5 分词

分词是指按照语境和语言规则，将连续的字序列重新组合成有意义的词序列的过程。与英文文本不同的是，中文文本没有按照空格去划分词，而是连续的汉字序列。因此本文决定采取中文分词操作。我们所使用的中文分词库为 jieba 词库，选择了 jieba 工具的精确模式，可以更好识别出专利文本数据中的专有名词。

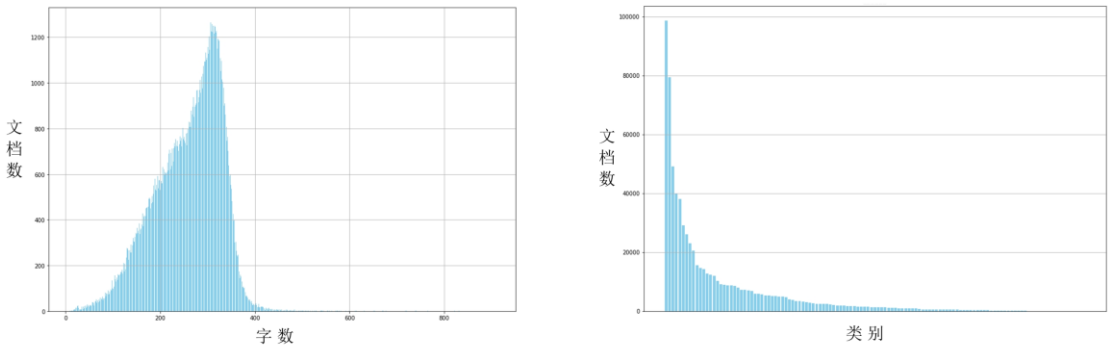
经过以上五步数据预处理后，CN-ICPC 数据集部分实例如表 3-2 所示。

表 3-2 预处理后的 CN-ICPC 数据集实例

一级	二级	三级	四级	摘要
上游	设备	测试 设备	激光器	一种 垂直 腔面 发射 激光器 TO 同轴 封装 测试 夹具 特性 包括 一底件 底件 矩形 结构 底件 …
上游	材料	晶圆 制造 材料	光刻胶	本发明 提供 一种 光刻胶 去除 方法 涉及 显示 技术 领域 解决 现有 光刻胶 去除 干净 一种 …
中游	制造	模拟 芯片 制造	射频 芯片 制造	申请 涉及 无线通信 领域 涉及 一种 射频 感知 方法 相关 装置 方法 包括 第一 设备 发送 …
中游	设计	存储芯片 设计	DRAM 芯片 设计	一种 动态 随机存取 存储器 DRAM 执行 读 写 刷新 操作 DRAM 包括 多个 子 阵列 子 阵列 …
中游	制造	模拟 芯片 制造	射频 芯片 制造	一种 射频 连接器 组合 包括 相互 对接 第一 连接器 连接器 连接器 包括 壳体 导电 端子 组 …
上游	设计 工具	EDA 设计 工具 1	EDA 设计 工具 2	本发明 提供 一种 布线 电路 基板 包括 金属 支承 层 绝缘层 导体 层 绝缘层 接 地层 厚度 …
中游	设计	通讯 芯片 设计 1	通讯 芯片 设计 2	一种 OFDM 通信 系统 扩宽 信号 频带 降低 光电 变换 中 副 载波 间 干涉 接收 信号 失真 …
中游	封测	封装 1	封装 2	半导体 封装 件 包括 第一 管芯 第一 分布 结构 位于 第一 管芯 第一 分布 结构 第一 管芯 …

3.4 数据统计

我们对 CN-ICPC 数据集的摘要文本和所有标签（共 123 类）进行统计和分析，结果如图 3-3 所示，与其他经典的大规模数据集不同，CN-ICPC 数据集的摘要文本长度和各标签数量没有存在分布不均衡的问题，但存在少数极端的样本。经过数据预处理后，数据集平均摘要文本长度仍然有 259.27 个字，保留了大量专业术语和名词短语。



(a) 数据集中的摘要文本长度　　(b) 数据集中各标签数量

图 3-3　CN-ICPC 数据集的统计分析

接着，我们采取旭日图的方式，将经过预处理后 CN-ICPC 数据集的第 2 至 4 级标签类别的层次关系进行可视化，如图 3-4 所示。旭日图的扇区大小代表了该层级或子层级所占比例的大小，因此较大的扇区通常表示更为重要的标签。在图 3-4 中，我们可以明显地看出半导体产业链上游的“设备”是占比最高的，因为半导体设备领域是半导体产业链关键的支撑产业，在上游环节的市场空间开阔，战略价值高。同时，半导体产业链中游的“封测”占比最少，半导体封测产业位于半导体产业链末端，其子领域“封装”和“测试”所需技术相对简单，价值含量低，属于劳动密集型产业。

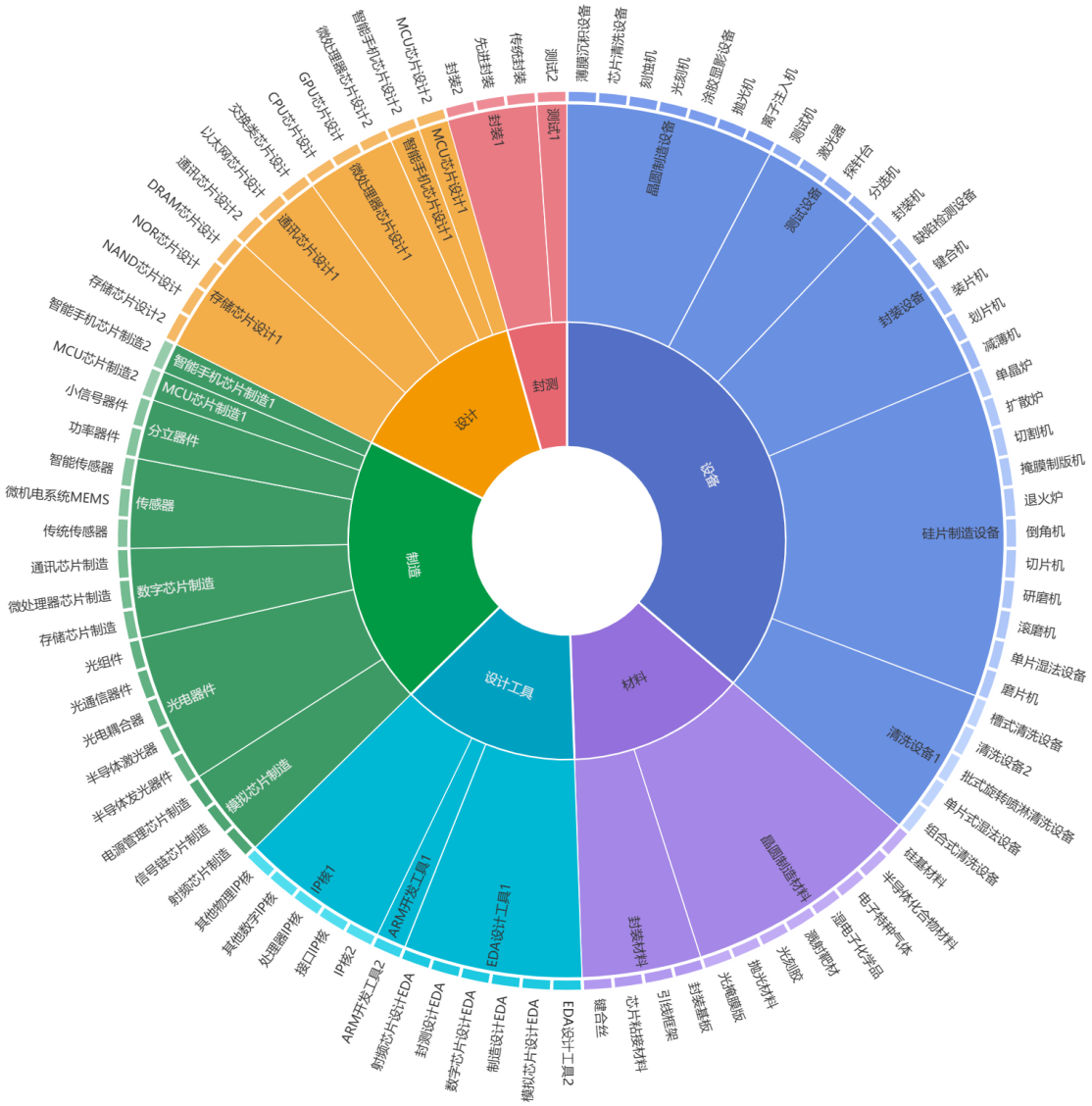


图 3-4　第 2 至 4 级标签旭日图

3.5 数据的向量化表达

目前，常见的预训练 GloVe 模型文件有：

- glove.6B.zip：基于维基百科 2014 年版本和 Gigaword 5 数据集训练，60 亿词条，40 万词汇量，不区分大小写，50/100/200/300 维向量，822MB。
- glove.42B.300d.zip：基于 common crawl 数据集训练，420 亿词条，190 万词汇量，不区分大小写，300 维向量，1.75GB。
- glove.840B.300d.zip：基于 common crawl 数据集训练，8400 亿词条，220 万词汇量，区分大小写，300 维向量，2.03GB。
- glove.twitter.27B.zip：基于 Twitter 数据训练，20 亿推文，270 亿词条，120 万词汇量，不区分大小写，25/50/100/200 维向量，1.42GB。

对于中文文本数据，由于语言差异，不能直接使用在英文文本数据上训练的 GloVe 模型文件，而且专利数据集中存在大量专有名词，因此我们需要训练一个新的中文数据集词向量。本文使用了斯坦福大学（Stanford NLP）的 GloVe 开源技术对 CN-ICPC 数据集的 178,118 项摘要文本进行训练，实验设置词向量维度为 300 维。我们最终得到 48563 个词汇量，大小为 132MB 的词向量文件。

4. 层次多标签文本分类模型

4.1 基于层次感知的全局模型

基于层次感知的全局模型（HiAGM）在 2020 年由 Alibaba NLP 团队[25]提出，它解决了其他层次型神经网络未能有效利用文本特征与标签的问题，当年达到 SOTA（State-Of-The-Art）。模型的思想是以有向图的形式表现层次结构，并提出层次感知的结构编码器以构建标签间的依赖关系，进一步设计了一种全新的端到端层次感知型全局模型。HiAGM 可以按照结构编码器和特征传播机制的不同分出 4 种变体，分别是 LA-GCN，LA-TreeLSTM，TP-GCN 和 TP-TreeLSTM。

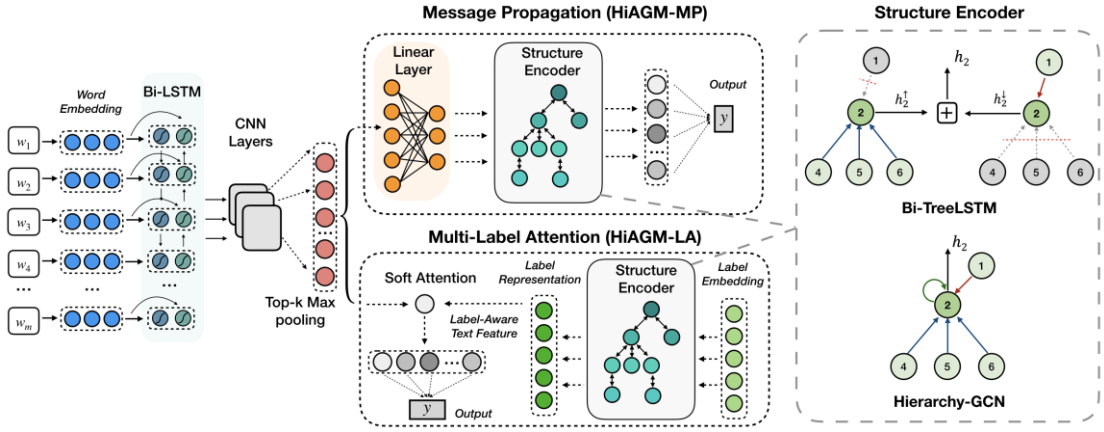


图 4-1　HiAGM 模型架构[25]

如图 4-1 所示，HiAGM 模型架构可以分为 3 个部分，从左至右依次是文本编码器、特征传播机制和结构编码器。文本编码器部分中，输入文本首先经过 Bi-LSTM 处理以学习词汇的表征，之后通过 CNN 抽取 N-gram 特征，最终采用 Top-k Max pooling 保留更多关键特征，形成最终的文本特征向量 *S*。

4.1.1 先验层次信息

模型首次提出一种先验层次信息（Prior Hierarchy Information）的概念，将理解标签 label 节点之间的层次信息转为概率问题。如图 4-2 所示，从父类（Root）到各个子类传递的概率和为 1，而子类到父类传递的概率为 1，这些从训练数据中统计。有了先验层次信息，可以预先了解标签之间的层次概率关系。假设在父

节点v_i与子节点v_j间存在一个层次结构路径$e_{i,j}$，该条边的特征$f(e_{i,j})$可以由先验概率$P(U_j|U_i)$和$P(U_i|U_j)$表示为公式 4-1。

$$\begin{cases} P(U_j|U_i)=\dfrac{P(U_j\cap U_i)}{P(U_i)}=\dfrac{P(U_j)}{P(U_i)}=\dfrac{N_j}{N_i}, \\ P(U_i|U_j)=\dfrac{P(U_i\cap U_j)}{P(U_j)}=\dfrac{P(U_j)}{P(U_j)}=1.0 \end{cases} \tag{4-1}$$

其中，U_k表示v_k存在且$P(U_j|U_i)$是v_j存在时v_i出现的条件概率，$P(U_j\cap U_i)$表示$\{v_j,v_i\}$同时发生的概率，N_k表示U_k在训练集中出现的数量。

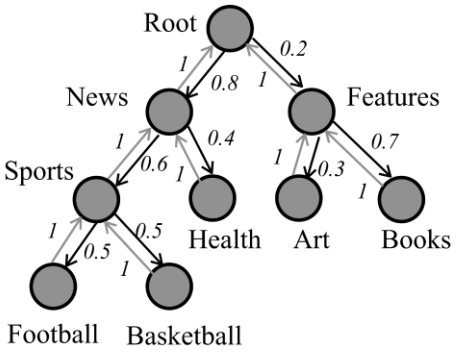


图 4-2 先验层次信息的示例[25]

4.1.2 层次感知的结构编码器

TreeLSTM 和 GCN 是 NLP 领域中常见的处理节点信息的结构编码器。在 HiAGM 中结构编码部分有 2 种编码策略，一种是 Bi-TreeLSTM，另一种是 Hierarchy-GCN，都是在 TreeLSTM 或 GCN 引入先验层次信息即层次感知。

（1）TreeLSTM 是利用 LSTM 对 Tree 结构的数据编码，Bi-TreeLSTM（Bidirectional TreeLSTM）则是双向的 TreeLSTM，双向在模型中指的是父类与子类互相之间的方向，节点编码的公式如下：

$$\begin{aligned} \tilde{h}_k^{\uparrow}&=\sum\nolimits_{j\in child(k)} f_p(e_{k,j})h_j^{\uparrow}, \\ \tilde{h}_k^{\uparrow}&=f_c(e_{k,j})h_p^{\downarrow}, \\ h_k^{bi}&=h_k^{\uparrow}\oplus h_k^{\downarrow}, \end{aligned} \tag{4-2}$$

其中，h_k 表示节点 k 的隐藏状态，$\tilde{h}_k^{\uparrow}$ 和 $\tilde{h}_k^{\downarrow}$ 参与自底向上和自顶向下的 TreeLSTM$\left(\tilde{h}_k\right)$计算过程，$\oplus$表示隐藏层的连接，最后一个隐藏节点 k 表示为h_k^{bi}。

（2）GCN 对标签节点编码时，是利用临近节点进行计算，Hierarchy-GCN 则是利用了自底向上、自顶向下和自循环的路径，根据节点 k 的相关邻域编码其隐藏状态，其计算公式如下：

$$\begin{aligned} u_{k,j} &= a_{k,j} v_j + b_l^k, \\ g_{k,j} &= \sigma(W_g^{d(j,k)} v_k + b_g^k), \\ h_k &= \mathrm{Re}LU\left(\sum\nolimits_{j \in N(k)} g_{k,j} \odot u_{k,j}\right), \end{aligned} \tag{4-3}$$

其中，$d(j,k)$为节点 j 到节点 k 的路径，包括自底向上、自顶向下和自循环三个方向。$a_{k,j}$ 为对于节点的先验层次信息，若 $d(j,k)$ 为自底向上（子节点到父节点）时，$a_{k,j} = N_k / N_j$；若 $d(j,k)$ 为自顶向下（父节点到子节点）时，$a_{k,j} = 1$；若 $d(j,k)$ 自循环，$a_{k,j} = 1$。

4.1.3 混合信息聚合

模型的特征传播机制阶段有 2 种方式，基于层次感知的多标签注意力机制（Hierarchy-Aware Multi-Label Attention）和层次化文本特征传播机制（Hierarchical text feature propagation）。

（1）基于层次感知的多标签注意力机制是通过软注意力机制把输入的文本特征向量 S 和标签体系向量 h 进行交互计算，V 为结果向量，公式可表示为：

$$\alpha_{kj} = \frac{e^{s_j h_k^T}}{\sum_{j=1}^{n} e^{s_j h_k^T}},\ v_k = \sum_{i=1}^{n} \alpha_{ki} s_i \tag{4-4}$$

其中，α_{ki} 表示第 i 个文本特征向量对第 k 个标签的信息量。

（2）层次化文本特征传播机制是将文本特征向量 S 进行全连接，转为向量 V 如公式 4-5 所示，之后进入结构编码器参与计算。

$$V = MS \tag{4-5}$$

4.2 基于信息最大化的全局模型

对于 HiAGM-LA 模型，其结构编码器存在两个问题，一是将每个文本数据与所有标签进行关联，其中包含了不相关的信息；二是未对结构编码器学习得到的标签表示施加统计约束。在 2021 年，Deng 等[26]提出了 HTCInfoMax 模型架构，设计出了一种“信息最大化”的方法，包括文本-标签互信息最大化和标签先验分布这两个模块去解决了 HiAGM-LA 模型存在的问题，其模型架构如图 4-3 所示。

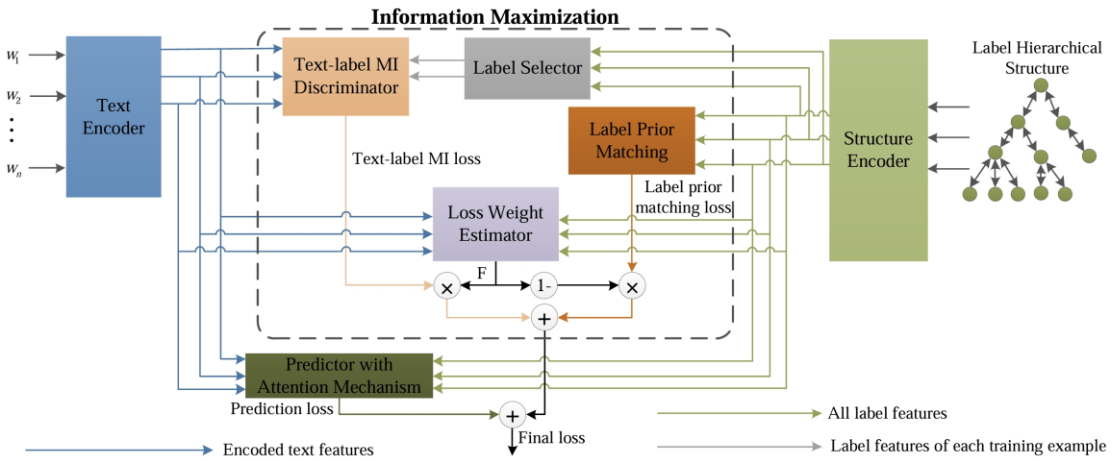


图 4-3　HTCInfoMax 模型架构[26]

4.2.1 文本-标签互信息最大化

文本-标签互信息最大化的这个模块，针对每个文本数据与所有标签进行关联的情况进行改进，使多标签注意力能更准确地提取与文本真实类别对应的特征，减小无关标签信息的干扰，从而提高模型预测的精准度。

4.2.2 标签先验分布

标签先验分布（Label prior matching）为模型提取的标签向量施加了一定的约束条件，以此改善样本中稀疏类别的向量表征。其中引入额外的分类器：前者通过构建一个二分类器来区分真实与虚假的文本标签对，后者则通过构建一个二分类器来区分真实标签向量与来自均匀先验分布的虚假标签向量。这两个分类器的损失函数被并入总体损失函数中，以便一起进行优化。

4.3 基于层次感知的标签语义匹配网络模型

基于层次感知的标签语义匹配网络模型（HiMatch）[27]也是基于层次感知的模型，在 HiAGM 模型的基础上，它提出的思想是将文本和标签分别进行表征学习，根据两表征向量定义不同的优化目标，从而提升层次多标签文本分类的效果。

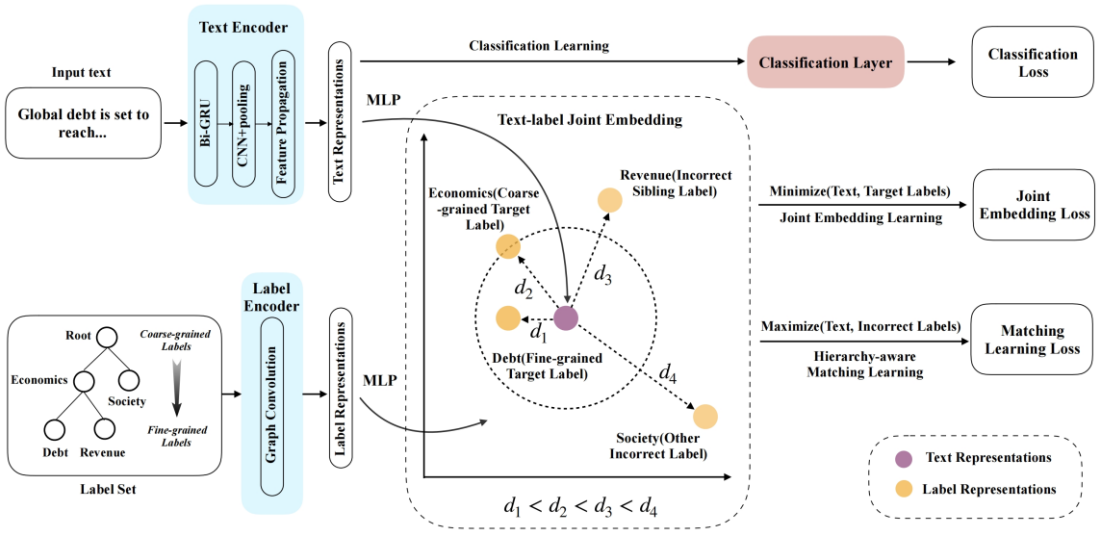


图 4-4　HiMatch 模型架构[27]

如图 4-4 所示，HiMatch 模型架构逻辑：文本编码器和标签编码器分别提取文本和标签的语义信息，将它们映射到共同的嵌入空间。联合嵌入损失的引入鼓励文本语义与目标标签语义相似。如图 4-5 通过引入匹配学习损失，确保细粒度标签（如“Debt”）在语义上最接近文本语义，其次是粗粒度标签（如“Economics”），而其他不正确的标签语义（如“Revenue”）与文本语义的语义距离更远。相对距离顺序为 $d_1 < d_2 < d_3 < d_4$，其中 d 表示联合嵌入中的距离度量。

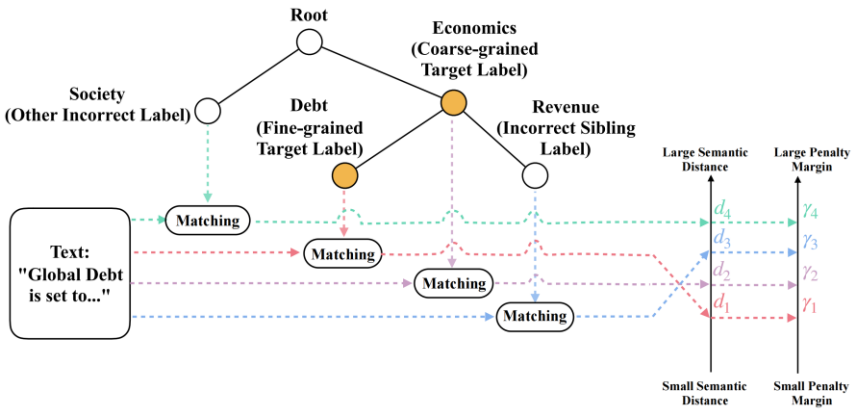


图 4-5 语义匹配示意图[27]

5. 实验与分析

5.1 实验环境及设置

本实验使用 AutoDL 算力云租用平台的服务器，软硬件配置为：64 位 ubuntu20.04 操作系统，内存为 60.0GB，CPU 为 AMD EPYC 9754 128-core processor，GPU 为 NVDIA GeForce RTX4090（24G）与 NVDIA A100-PCIE-40GB，CUDA 版本为 11.3.0，深度学习框架及版本为 Pytorch1.10.0，Python3.8.0。

5.2 实验数据集

我们将 CN-ICPC 数据集中“摘要”特征作为模型的输入文本，“一级”至“四级”作为模型的输出标签。专利数据集的划分，本文按照 8：2 将数据集随机划分成训练集与测试集，接着也按照 8：2 将训练集随机划分成最终的训练集和验证集。

在层次多标签文本分类任务中，有 2 个经典的数据集 WOS 和 RCV1-V2。WOS[23]是由 Web of Science 科学网站的论文摘要所组成，RCV1-V2[24]是一个由路透社新闻组成的文本分类语料库。我们将 CN-ICPC 数据集与这 2 个公开数据集进行统计比较，如表 5-1 所示。

表 5-1 各数据集的统计信息

统计数据	CN-ICPC	WOS	RCV1-V2
总文档数	178,118	46985	804,414
训练集	113,995	30070	20833
验证集	28499	7518	2316
测试集	35624	9397	781,265
第一层标签数	2	7	4
第二层标签数	6	134	55
第三层标签数	24	-	43
第四层标签数	91	-	1
总标签数	123	141	103
标签深度	4	2	4
平均标签深度	4.0	2.0	3.24

5.3 评价指标

实验的评价指标为 micro-F1 和 macro-F1。在二分类模型背景下，若将特定类别视为正类，其他类别则视为负类。根据分类器对数据集的预测结果，可正确或不正确，分为 TP、TN、FP、FN 四种情况：True Positive (TP)表示把正样本成功预测为正；True Negative (TN)表示把负样本成功预测为负；False Positive (FP)表示把负样本错误地预测为正；False Negative (FN)表示把正样本错误的预测为负。在二分类模型中，精确率（Precision），召回率（Recall）和 F1 值定义如下。

$$\text{Precision} = \frac{TP}{TP + FP} \tag{5-1}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{5-2}$$

$$F_1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5-3}$$

F1 值是精确率与召回率的调和平均数。鉴于精确率与召回率是相互矛盾的关系，当精确率高时，召回率相对较低，当召回率高时，精确率相对较低，所以 F1 值可以衡量分类器综合性能。基于此，在多标签场景，我们选用 micro-F1 与 macro-F1 作为评价指标。micro-F1 通过计算所有类别的总精确率与召回率来确定 F1 值；而 macro-F1 则首先计算每个类别的 F1 值，随后计算这些 F1 值的平均值。micro-F1 与 macro-F1 定义如公式 5-4 和 5-5 所示。

$$micro - F1 = F_1\left(\sum_{i=1}^{K} TP_i, \sum_{i=1}^{K} FP_i, \sum_{i=1}^{K} TN_i, \sum_{i=1}^{K} FN_i\right) \tag{5-4}$$

$$macro - F_1 = \frac{1}{K}\sum_{i=1}^{K} F_1\left(TP_i, FP_i, TN_i, FN_i\right) \tag{5-5}$$

其中，F_1 为公式 5-3，K 为类别个数。

根据公式 5-4 和 5-5 可知，micro-F1 为每个样本赋予相同权重，其结果更受样本数量较多的类别影响；相对，macro-F1 为每个类别赋予等同权重，使得结果更受稀有类别的影响。在类别不均衡时，采用 micro-F1 对模型评估更为合理。

5.4 实验参数设置

实验的相关参数设置如表 5-2 所示。

表 5-2 实验参数设置

参数	HiAGM	HTCInfoMax	HiMatch
词向量维度	300	300	300
文本编码器类型	GRU	GRU	GRU
结构编码器类型	GCN/TreeLSTM	TreeLSTM	GCN
训练丢弃率	0.5	0.5	0.5
优化器	Adam	Adam	Adam
学习率衰减	1.0	1.0	1.0
学习率	0.0001	0.0001	0.0001
批样训练样本数	64	64	64
分类损失函数	BCEWithLogitsLoss	BCEWithLogitsLoss	BCEWithLogitsLoss
正则化惩罚系数	0.000001	0.000001	-

5.5 对比模型及结果分析

为了验证 CN-ICPC 专利文本数据集的质量，本文使用了多个基于层次感知的模型进行训练，并对比在 WOS 数据集和 RCV1-V2 数据集的结果[25-27]，实验结果比较如图 5-1 与表 5-3 所示。

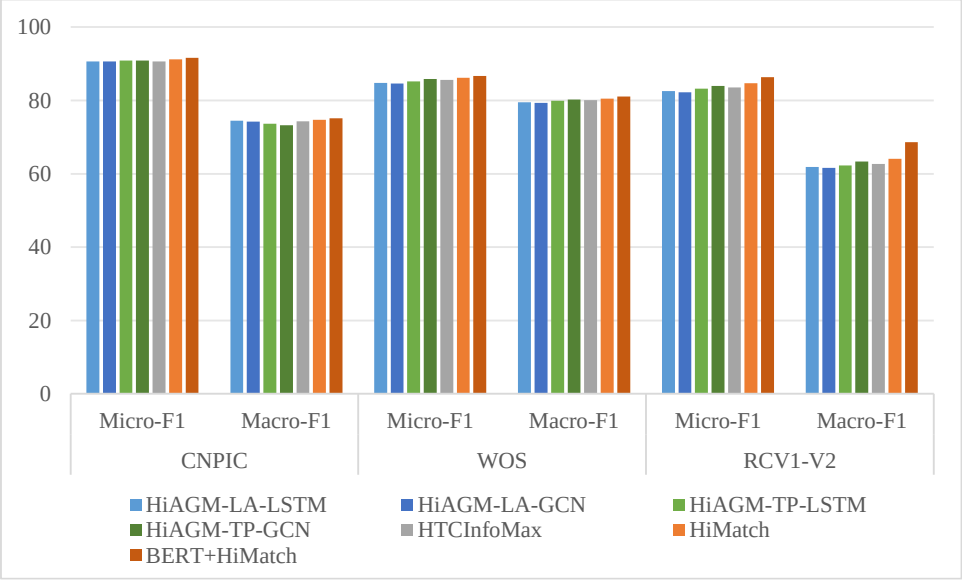


图 5-1 实验结果比较图

表 5-3 实验结果比较表

Model	CN-ICPC		WOS		RCV1-V2	
	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1
HiAGM-LA						
TreeLSTM	90.60	74.52	84.82	79.51	82.54	61.90
GCN	90.66	74.22	84.61	79.37	82.21	61.65
HiAGM-TP						
TreeLSTM	90.90	73.66	85.18	79.95	83.20	62.32
GCN	90.87	73.21	85.82	80.28	83.96	63.35
HTCInfoMax	90.65	74.35	85.58	80.05	83.51	62.71
HiMatch	91.19	74.71	86.20	80.53	84.73	64.11
BERT+HiMatch	**91.62**	**75.14**	**86.70**	**81.06**	**86.33**	**68.66**

我们通过观察和比较各个模型在不同层次数据集上的实验结果，可以得知在 Micro-F1 方面，各类算法在数据集上表现 CN-ICPC＞WOS＞RCV1-V2；在 Macro-F1 方面，在数据集上表现 WOS＞CN-ICPC＞RCV1-V2。各个模型在 CN-ICPC 数据集都表现良好，尤其是 Micro-F1 值均达到 90%以上，说明在 CN-ICPC 数据集中的大部分标签与文本特征之间存在较强的相关性，且各类模型能够有效学习这些特征。而 WOS 数据集在 Macro-F1 值表现均优于 CN-ICPC 数据集，可能是标签的层次过深影响数据的类别平衡方面，导致模型在 CN-ICPC 数据集上处理数量较少的类别能力不足。

此外，我们还可以得知 HiMatch 模型在 3 个数据集上都表现最佳，证明 HiMatch 模型创新性的标签语义匹配网络非常有效。接着，将我们 HiMatch 模型与 BERT 结合起来，我们用 BERT 替换了 HiMatch 的文本编码器，其中 BERT 去预训练标签的表示。相比其他模型，BERT+HiMatch 可以更好地学习细粒度标签与文本语义。

我们通过比较各类层次多标签分类模型在 CN-ICPC 数据集上的表现，不仅验证了各个模型能力的真实性和有效性，而且说明 CN-ICPC 数据集是一个大规模、优质的中文层次文本数据集。

6. 总结

随着全球人工智能(AI)和生成式 AI 迅速发展与激烈竞争，强大的算力资源变得至关重要，而能够提供这些资源的半导体产品正面临着供不应求的局面。全球各国以及众多大型科技公司都在积极推动半导体产业链的发展，认可半导体产业是推动国家经济增长和科技创新的核心动力。在当前大背景下，深入研究和开发半导体行业的专利技术是科技进步的重要方向。基于此，本文构建了一个大规模且细粒度的中文文本语料库 CN-ICPC，该语料库专注于半导体产业链的专利技术。我们根据工业界对半导体产业链的细分领域，对标签进行细粒度划分，运用了多种基于层次感知的先进模型进行深度训练，与同类公开的层次化文本数据集相比，结果表现较好，证明 CN-ICPC 数据集是一个大规模、优质的中文层次文本数据集。通过这种精细化的构造方式，专利文本数据集的研究不仅为半导体产业的未来发展提供了新的视角，而且为半导体技术未来的创新和应用提供了坚实的数据基础。

参考文献

[1] Wu F, Zhang J, Honavar V. Learning classifiers using hierarchically structured class taxonomies[C]//Abstraction, Reformulation and Approximation: 6th International Symposium, SARA 2005, Airth Castle, Scotland, UK, July 26-29, 2005. Proceedings 6. Springer Berlin Heidelberg, 2005: 313-320.

[2] Costa E P, Lorena A C, Carvalho A C, et al. Comparing several approaches for hierarchical classification of proteins with decision trees[C]//Advances in Bioinformatics and Computational Biology: Second Brazilian Symposium on Bioinformatics, BSB 2007, Angra dos Reis, Brazil, August 29-31, 2007. Proceedings 2. Springer Berlin Heidelberg, 2007: 126-137

[3] Silla C N, Freitas A A. A survey of hierarchical classification across different application domains[J]. Data mining and knowledge discovery, 2011, 22: 31-72.

[4] Hinton G E, Osindero S, Teh Y W. A fast learning algorithm for deep belief nets[J]. Neural computation, 2006, 18(7): 1527-1554.

[5] Sutskever I, Martens J, Hinton G E. Generating text with recurrent neural networks[C]//Proceedings of the 28th international conference on machine learning (ICML-11). 2011: 1017-1024.

[6] Graves A, Graves A. Long short-term memory[J]. Supervised sequence labelling with recurrent neural networks, 2012: 37-45.

[7] KIM Y. Convolutional neural networks for sentence classification[J/OL]. CoRR, 2014, abs/1408.5882. http://arxiv.org/abs/1408.5882.

[8] Lai S, Xu L, Liu K, et al. Recurrent convolutional neural networks for text classification[C]//Proceedings of the AAAI conference on artificial intelligence. 2015, 29(1).

[9] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arxiv preprint arxiv:1810.04805, 2018.

[10] Molla D, Seneviratne D. Overview of the 2018 alta shared task: Classifying patent

applications[C]//Proceedings of the Australasian Language Technology Association Workshop 2018. 2018: 84-88.

[11] Piroi F, Lupu M, Hanbury A, et al. CLEF-IP 2011: Retrieval in the Intellectual Property Domain[C]//CLEF (notebook papers/labs/workshop). 2011.

[12] Li S, Hu J, Cui Y, et al. DeepPatent: patent classification with convolutional neural networks and word embedding[J]. Scientometrics, 2018, 117(2): 721-744.

[13] Sharma E, Li C, Wang L. BIGPATENT: A large-scale dataset for abstractive and coherent summarization[J]. arxiv preprint arxiv:1906.03741, 2019.

[14] Suzgun M, Melas-Kyriazi L, Sarkar S, et al. The harvard uspto patent dataset: A large-scale, well-structured, and multi-purpose corpus of patent applications[J]. Advances in Neural Information Processing Systems, 2024, 36.

[15] Chen, Guibin, et al. "Ensemble application of convolutional and recurrent neural networks for multi-label text categorization." 2017 International joint conference on neural networks (IJCNN). IEEE, 2017.

[16] Yang P, Sun X, Li W, et al. SGM: sequence generation model for multi-label classification[J]. arxiv preprint arxiv:1806.04822, 2018.

[17] Sabour S, Frosst N, Hinton G E. Dynamic routing between capsules[J]. Advances in neural information processing systems, 2017, 30.

[18] Aly R, Remus S, Biemann C. Hierarchical multi-label classification of text with capsule networks[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop. 2019: 323-330.

[19] Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[C]//International Conference on Learning Representations. 2013.

[20] Pennington J, Socher R, Manning C D. Glove: Global vectors for word representation[C]//Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014: 1532-1543.

[21] Fukushima K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position[J]. Biological cybernetics, 1980, 36(4): 193-202.

[22] LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.

[23] Kowsari K, Brown D E, Heidarysafa M, et al. Hdltex: Hierarchical deep learning for text classification[C]//2017 16th IEEE international conference on machine learning and applications (ICMLA). IEEE, 2017: 364-371.

[24] Lewis D D, Yang Y, Russell-Rose T, et al. Rcv1: A new benchmark collection for text categorization research[J]. Journal of machine learning research, 2004, 5(Apr): 361-397.

[25] Zhou J, Ma C, Long D, et al. Hierarchy-aware global model for hierarchical text classification[C]//Proceedings of the 58th annual meeting of the association for computational linguistics. 2020: 1106-1117.

[26] Deng Z, Peng H, He D, et al. HTCInfoMax: A global model for hierarchical text classification via information maximization[J]. arxiv preprint arxiv:2104.05220, 2021.

[27] Chen H, Ma Q, Lin Z, et al. Hierarchy-aware label semantics matching network for hierarchical text classification[C]//Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). 2021: 4370-4379.

[28] Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural computation, 1997, 9(8): 1735-1780.

[29] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[J]. arxiv preprint arxiv:1609.02907, 2016.

[30] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate[J]. arxiv preprint arxiv:1409.0473, 2014.

致谢

“斗指东南，维为立夏，万物至此皆长大”。转眼到了立夏，我的本科生涯即将到达尾声。回首四年的点点滴滴，百感交集。

首先，我要向我的导师张博闻博士表达我最诚挚的谢意！我与张老师相识了2年时间，他渊博的学识、严谨的治学态度和高尚的品德深深地影响了我！在求学之路上，他给予我专业、耐心的教导；在生活中，他十分亲和，更像是我的一位好友，对我说了许多发自内心的人生建议。

同时，我还要感谢我的家人、朋友们、班主任、老师们，他们的理解和支持使我在困难时不断奋起，他们的爱是我前行的动力源泉。

“Patience is key in life.”这句话一直鼓励着我向前。在实验过程中，我深刻地体会到了科研道路上的“酸甜苦辣咸”。面对实验中遇到的种种挫折与挑战，我一直抱着一颗乐观向上的心，以持之以恒的精神，克服了大部分难关。

如《道德经》所云：“为之者败之，执之者失之。是以圣人无为也，故无败也；无执也，故无失也。”在未来的生活中，我会继续保持谦逊的态度和开放的思想，不断学习前沿的知识，丰富自己的人生。再一次感谢所有帮助我和支持我的人，是你们让这段旅程变得更加有意义。忏悔，感恩，相信。谢谢大家！