附 录

1 方法

1.1 数据预处理

对专利原始数据进行文本预处理主要包含数据清洗、名词短语提取、去停用词、词形还原以及文本向量化。预处理主体流程见图 1。各项处理工作具体内容如下：

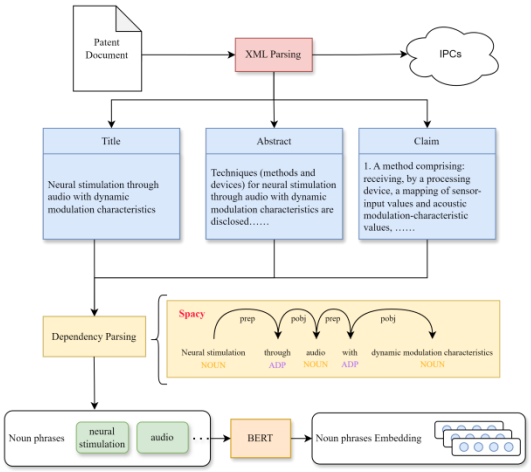


图 1 专利预处理

数据清洗：使用 Python 的 ElementTree 模块解析原始 XML 数据，从中筛选出标题、摘要、权力要求书、IPC 分类号相关的数据。权力要求书仅筛选独立权利要求，因为独立权力要求一般从整体上描述了发明创造的关键技术特征和必要组成部分，而由于收集的专利数据中可能前几项权利要求是“被取消”的状态，因此权利要求书的部分只存储第一项有效的权利要求文本作为独立权利要求。

名词短语提取：使用 spaCy 工具提取文本中的名词短语并去重，避免模型过度关注某些重复输入。

去停用词：名词短语中可能会包含一些停用词，如“a”、“the”等，除此之外，还需要扩展停用词列表，如“method”、“claim”、“system”等词会频繁地出现在专利文本中，且会被识别为名词短语，但其对于专利的表征不具有区分性。

词形还原：词形还原是将单词转化为其原始形式，比如从复数形式转为单数形式，有助于后续对语义的分析。

文本向量化：通过将文本形式的内容映射为一个高维空间的向量，可以让文本参与到模型的运算中。常用的词向量模型有 Word2Vec、GloVe、BERT 等。本章采用 BERT 模型获取一篇专利的表征，将专利提取出的名词短语作为一个批次输入 BERT 模型，获取各个名词短语的嵌入，以该专利所有名词短语嵌入构成一个序列，作为该专利的表征。

对于一份专利，通过预处理得到的名词短语集合记为 P，即 $P=\{p_1,p_2,\dots\dots,p_n\}$，其中 $p_i(i=1,2,\dots\dots,n)$ 代表专利中提取出的各个名词短语。每个名词短语使用 BERT 生成词嵌入 e_i，每个名词短语都是 768 维的嵌入向量，即 $e_i \in \mathbb{R}^{768}$。专利的名词短语嵌入序列记为 $Embed=\{e_1,e_2,\dots\dots,e_i\}$。

1.2 SAO 提取流程

本研究利用文本挖掘工具，从专利的文本中提取 SAO 结构，具体流程如图 2 所示。

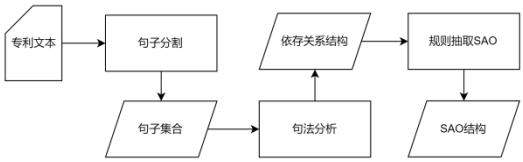


图 2 专利 SAO 结构提取流程图

(1) 句子分割

英文专利的权利要求书格式与中文专利类似，通常分为独立权利要求和从属权利要求。独立权利要求定义了发明的核心特征，一般位于权利要求的开头。这类权利要求常用“A method for…”或“A XX system comprising…”等表达方式，简要概括了发明的基本内容。独立权利要求中的“comprising”一词后面通常跟随多个以逗号或分号分隔的技术要点或组件以进一步定义发明的具体技术特征。从属权利要求通常位于独立权利要求之后，其特点是采用诸如“The XX of claim XX…”或“The XX according to claim XX…”的表述形式，引用独立权利要求并对其中提到的某个技术或组件进行进一步的限定或补充。这种结构使得专利的技术细节和创新点可以得到更加精确的描述。针对专利文本的特点，本研究采取了自定义的句子分割策略。由于专利中的组件描述往往较长，且多个技术要点常常用逗号或分号分隔，这就容易导致标准的 spaCy 句子分割器错误地将一个包含多个组件的句子分割为多个独立句子。因此，需要将这些以逗号或分号分隔的技术组

件与主句连接成一个完整的句子，以避免不必要的误分割。此外，标准的 spaCy 句子分割器在处理原始文本时，可能会错误地将标点符号与单词连接在一起，导致标点符号和单词被误认为是一个整体。这种错误会导致句法分析时将多个句子误解析为一个句子，影响结果的准确性。为了解决这一问题，本研究设计了一个自定义的 spaCy 组件，检查每个 token 是否包含逗号、分号或句号。如果发现 token 中含有这些标点符号，系统会将下一个 token 标记为新的句子开始。这样可以确保在分析过程中，句子的边界得以正确划分，从而提高句法分析的准确性，更好地保留技术要点和组件之间的逻辑关系。

(2) 句法分析

部分词性标记如表 1 所示，部分句法分析标记如表 2 所示。在 SAO 中，主体（Subject）、行为（Action）和客体（Object）分别对应依存句法分析的主语、谓语和宾语，它们通常对应特定的词性和句法依赖标注。主体通常被标记为“nsubj”（名词性主语），行为作为动词通常标注为“ROOT”或“VERB”，而客体则由“dobj”（直接宾语）表示。如图 3，句子“The invention provides a solar panel that automatically adjusts its angle based on sunlight intensity.”经过 spaCy 解析后，“invention”（名词）、“provides”（动词）和“panel”（名词）分别形成“nsubj”和“dobj”依赖关系，精准地对应了 SAO 结构。这些依赖标注在大多数简单句中能有效揭示句子的基本语法框架。在处理更复杂的句式时，仅依赖基础的依存关系可能不足以准确定义主体、行为和客体的角色，它们之间的关系并非总是如此直接

表 1 spaCy **词性标记（部分）**

符号	含义	符号	含义
NOUN	名词	ADP	介词
VERB	动词	CONJ	连词
ADJ	形容词	CCONJ	并列连词
ADV	副词	SCONJ	从属连词
PRON	代词	NUM	数词
DET	限定词	PUNCT	标点符号

表 2 spaCy **句法分析标记（部分）**

符号	含义	符号	含义
ROOT	句子的核心动词（根节点）	amod	名词的形容词修饰语
nsubj	名词性主语	advmod	副词修饰语
csubj	句子性主语	neg	否定修饰语
xcomp	开放补语	advcl	副词性从句修饰语
dobj	直接宾语	conj	并列成分
iobj	间接宾语	cc	并列连词
pobj	介词短语中的宾语	agent	被动结构中的执行者
mark	从句引导标记	prep	介词

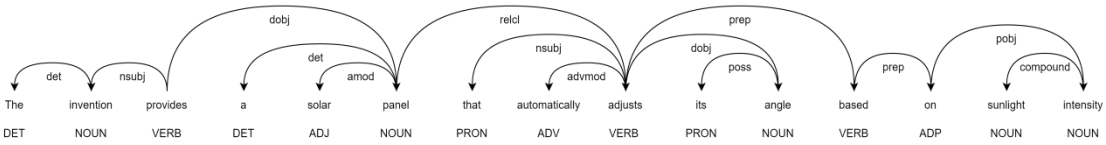


图 3 句法分析示例

(3) SAO 抽取规则

在复杂句式中，句子的结构通常涉及嵌套、修饰和多层次的依赖关系，使得简单的依赖标注无法完全覆盖所有可能的语法现象。例如，修饰性成分（如形容词或副词）可能会嵌入主语或宾语之中，形成更复杂的名词短语，这些修饰词的存在并不意味着它们就是句子的主语或宾语。同时，在带有从句或并列句的复杂句中，句子结构可能会引入多个主谓宾关系。此外，在某些情况下，动词的宾语可能并不直接通过“dobj”关系来标记，而是通过间接的依赖关系或语义推理来确定。这种语法的模糊性导致无法单纯依赖预定义的依赖标注（如“nsubj”、“dobj”等）来识别 SAO 结构，还需要对这些关系进行扩展，结合上下文、修饰成分的影响以及句法结构的复杂性，才能更准确地提取出 SAO 三元组。

为了实现该目标，本研究主要依赖 spaCy 句法树的五个关键属性：dep_、pos_、root、children 和 head。通过这些属性可以明确每个单词在句子中的语法角色和依赖关系。本文提出的 SAO 抽取规则如表 3 所示。

表 3 SAO 抽取规则

规则类型	依赖关系	示例
基础主谓宾	nsubj→VERB←dobj	The sensor detects the signal.
定语从句	VERB.dep_=acl	The device associated with claim 7 improves accuracy.
介宾补足语	VERB.dep_=pcomp	The method for detecting faults involves applying a series of tests.
状语从句	VERB.dep_=advcl	The system according to claim 1 comprising a calculation part adapted to calculate the driving index .
被动语态	nsubjpass→VERB←（prep，pobj）	The encryption algorithm is applied to the data before it is transmitted over the network.

在基础主谓宾结构的提取过程中，本研究采用了正向提取（主语 → 动词 → 宾语）和反向提取（宾语 → 动词 → 主语）两种互补的策略。正向提取的过程首先通过“nsubj”依存关系标识主语名词短语，并沿着 head 指针追溯至支配该主语的谓语动词，最后在谓语动词的子节点中筛选出符合“dobj ”依存关系的宾语。从正向提取的角度来看，主语与谓语动词之间的依赖关系通常较为直接。然而，在句子结构较为复杂的情况下，谓语动词的子节点依赖关系可能变得更加繁杂，这使得动词与宾语之间的依赖关系往往是间接的，从而增加了提取的难度。与之相对，反向提取从宾语出发，沿着 head 指针逐层追溯至谓语动词。这一策略能够避免复杂句子中谓语动词子节点过多导致的提取困难，从而使得提取主语的过程相对更加简单和直接。因此，反向提取发挥了主导作用，而正向提取则作为辅助策略，能够更好地处理不同句型的复杂性。

在本研究中，针对不同句法结构，制定了相应的 SAO 抽取规则。在处理主动语态时，本研究采用了反向提取的方法。首先，对于通过 spaCy 识别的所有名词块，将具有“dobj”（直接宾语）依赖关系的名词块作为候选对象（Object）。然后，从名词块所在的句法树向上遍历，找到最近的动词作为动作（Action）。对于定语从句，由动词引导的从句修饰前面的名词，通过“acl”依赖关系，动词与名词之间建立联系，分别对应主语（Subject）和动作（Action）。对于介宾补足语，动词的“pcomp”和“prep”依赖关系链可以帮助识别主语。对于状语从句，通过动词的“advcl”依赖关系向上追溯，找到句中的主语。针对被动语态，采用了不同于主动语态的思路。被动语态的经典依赖标识是“nsubjpass”，因此首先查找所有存在“nsubjpass”依赖关系的动词，然后将与这些动词存在“nsubjpass”关系的名词块作为候选主语（Subject），最后通过“prep”和“pobj”依赖关系链可以找到宾语（Object）。

在 SAO 提取的过程中，为了进一步优化提取结果，还包括了一些后处理规则，用于对初步提取的 SAO 三元组进行清理和调整，如表 4 所示。

表 4 SAO 后处理规则

规则	关键标识	含义
限定词过滤	DET	去除名词块的限定词，如“a”, “the”等
复合宾语	conj	与宾语并列的名词块逐个替换原 SAO 三元组
助动词扩展	auxpass、prep	合并助动词和相邻介词
助动词替换	is、are 等助动词	替换为 be

2 实验

2.1 基于元学习的 IPC 分类方法模型实验

2.1.1 数据集

本研究所使用的专利数据集是从 USPTO 官网收集而来。表 5 展示了收集的原始数据的统计信息。在图 4 中，BaseSet 的专利 IPC“小类”分布呈现显著的长尾特征。头部 10 个 IPC“小类”其专利数量均突破 10^4 量级，最多的“小类”达 43886 项专利，随后的排名 10 至 100 的“小类”其专利量分布在 10^3-10^4 之间，下降了一个数量级，剩下的 500 余项分布曲线呈指数级衰减趋势，仅占数据集的 20%左右，头部与腰部类别的专利样本数量级差达 10^1，头部与尾部的数量级差达到 10^3。

表 5 数据集统计信息

数据集	样本数	部	大类	小类	平均 IPC 数
BaseSet	368592	8	124	629	1.65
CoreSet	22260	8	118	470	2.20

在高度不平衡的数据集上进行多标签分类时，

样本数量的巨大差异会引发显著的学习偏差。模型在训练过程中会频繁接触头部类别的样本，这种重复性使得模型倾向于优先学习头部类别的特征表示，而尾部类别由于样本数量极其有限，难以获得充分的学习机会，导致模型无法建立有效的特征表示，最终造成尾部类别的分类性能显著劣化。

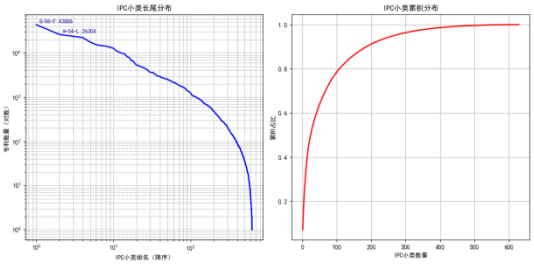


图 4 BaseSet IPC“小类”分布

为了缓解不平衡数据集的影响，本研究基于 BaseSet 的数据构造了 CoreSet。其构造方式是针对各个 IPC“小类”，随机抽样 50 个专利作为其数据，如果小于 50 个样本则剔除该 IPC“小类”。如果一个专利本身属于多个 IPC“小类”且同时被多个 IPC“小类”抽中，在 CoreSet 中仅会保留该专利的一条记录。因而 CoreSet 总样本数不能被 50 整除。CoreSet 的统计信息见表 5，“小类”分布见图 5。

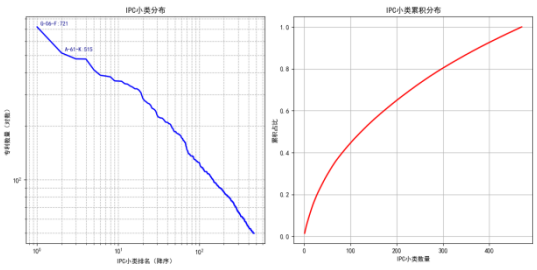


图 5 CoreSet IPC“小类”分布

从 CoreSet“小类”分布图中可见，各个类别的样本数量差异的数量级大大减小，但仍然不是完全均衡的。这种不平衡性主要源于专利的多标签特性，即每个专利样本可能会属于多个不同的 IPC“小类”。例如，在“G06N”采样的一个专利可能还有另一个 IPC 标签“G06F”，导致“G06F”的样本数被动地增加。然而，整体而言，这种程度的差异是可以接受的，因为这种处理方式显著改善了原始数据集中极端不平衡的问题。在 BaseSet 中，头部和尾部类别的样本数量级差异达到 10^3，而 CoreSet 将其缩小到 10^1，大大降低了模型对头部类别的偏好，使尾部类别获得了相对充足的样本支持。其次，适度的数量级差异反映了真实的技术分布，完全平衡的数据集并不现实。在专利分类中，某些技术领域的确比其他领域更活跃，这种差异是技术发展现状的自然体现。

2.1.2 评估指标

在本研究的场景中，每个专利样本往往同时属于多个类别，如表 5 所示，CoreSet 样本的平均“小类”IPC 数为 2.2，因而本研究是一个多标签分类任务，即一个样本同时拥有多个标签，这使得评估模型性能变得比传统的单标签分类更为复杂。为了全面评估模型的性能，本文采用了一系列基于 Top-k 的评估指标。这些指标从不同角度衡量了模型的预测能力，包括预测的准确性、覆盖率以及整体平衡性。

Precision@k 主要用于评估模型预测的前 k 个三级 IPC 标签中正确标签的占比情况，在此过程中，需依据模型预测的概率对标签进行降序排序。对于整个测试集而言，是对各个样本预测的前 k 个标签中正确标签的比例取平均值。Precision@k 的计算公式如下：

$$\text{Precision@}k = \frac{1}{N}\sum_{i=1}^{N}\frac{|\hat{Y}_i^k \cap Y_i|}{k}, \qquad (1)$$

其中，$|\hat{Y}_i^k \cap Y_i|$第 i 个样本预测标签按概率排序后，前 k 个标签中与真实标签集合相交部分的元素数量，即正确标签的数量；k 是设定的截断位置，该值决定了模型每次预测时考虑的标签数量上限。Precision@k 值越高，表明模型预测的前 k 个标签中正确标签的比例越大，模型在预测标签准确性方面的表现越出色。

Recall@k 衡量在模型预测的前 k 个标签中，成功召回的真实标签的比例，此指标重点关注模型对真实标签的覆盖程度。具体计算时，对于每个样本，需计算在前 k 个预测标签中成功找到的真实标签数量与该样本所有真实标签数量的比值。Recall@k 的计算公式如下：

$$\text{Recall@}k = \frac{1}{N}\sum_{i=1}^{N}\frac{|\hat{Y}_i^k \cap Y_i|}{|Y_i|}, \qquad (2)$$

其中，$|Y_i|$表示第 i 个样本的真实标签数量。

F1@k 分数是 Precision@k 和 Recall@k 的调和平均值，综合考量了模型预测的精确性与完整性，避免了仅优化单个指标可能带来的偏差，提供了一个平衡的性能度量。F1@k 的计算公式如下：

$$\text{F1@}k = \frac{2 \times \text{Precision@}k \times \text{Recall@}k}{\text{Precision@}k + \text{Recall@}k}, \qquad (3)$$

F1@k 值越接近 1，表明模型在精确性和召回率两方面的综合表现越优异，能够在保证预测准确性的

同时，尽可能多地覆盖真实标签，更全面、准确地反映模型在多标签分类任务中的性能水平。

2.1.3 实验设置

实验在配备 NVIDIA GeForce RTX 4090 GPU（24GB 显存）和 AMD Ryzen 9 9950X（16 核 32 线程）的服务器上进行，操作系统为 Ubuntu 24.04.2 LTS，使用 Python 3.9 和 PyTorch 2.5.1 框架。模型训练采用 Adam 优化器，元学习外层学习率设置为 1e-3，内层更新学习率设置为 0.1，每个训练批次包含 10 个任务，每个任务的支持集包含 20 个正样本和 20 个负样本，查询集包含 10 个正样本和 10 个负样本。模型在每个任务上进行 20 次内层更新，测试阶段同样进行 20 次内层更新。模型由一个单向 LSTM 和两个线性层组成。模型的总体输入维度为 768，输出维度为 2。

2.1.4 消融实验

(1) 表征方式对比

在专利分类任务中，文本表征方式的选择对模型性能有着至关重要的影响。为了探究不同文本嵌入方法对分类效果的影响，本研究设计了消融实验，对比了四种不同的文本表征方式：1）基于名词短语的 BERT 嵌入，即本研究采用的表征方式；2）基于名词短语的 Word2Vec 嵌入，从专利中提取出的名词短语借助 Word2Vec 词向量取平均，构建出名词短语嵌入序列；3）基于文档级别的 BERT 嵌入，将专利的标题、摘要和权利要求书整合为一个整体，利用 BERT 获取其整体嵌入；4）基于文档级别的 Word2Vec 嵌入，将专利的上述三部分文本合并，对所有单词的词向量取平均。对比的 Word2Vec 类表征方式均执行了去停用词、词形还原等基本的预处理操作。表 6 展示了四种表征方式在 Top@1、Top@3、Top@5 三个不同阈值下的分类性能。

表 6 不同表征方式性能对比

表征方式	Top@1			Top@3			Top@5		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
1	0.4862	0.2305	0.3127	0.2365	0.3482	0.2817	0.1699	0.4187	0.2417
2	0.4639	0.2327	0.3099	0.2288	0.3375	0.2727	0.1427	0.4093	0.2116
3	0.4903	0.2271	0.3104	0.2315	0.3401	0.2754	0.1583	0.4125	0.2289
4	0.4581	0.2265	0.3031	0.2151	0.3158	0.2558	0.1395	0.4052	0.2075

实验结果表明，基于名词短语的 BERT 嵌入方法在整体性能上表现最为优异，尤其是在不同 Top@k 阈值下均取得了最高的 F1 值。这一发现验证了将 BERT 模型应用于专利文本名词短语表征的有效性，表明该方法能够充分捕捉专利文本的局部语义特征，同时也证实了使用名词短语作为专利文档表征的合理性。进一步分析发现，基于 BERT 的表征方法在 F1 值上均显著优于 Word2Vec 类方法，这充分体现了 BERT 模型在处理专利文本复杂语义理解和特征提取方面的优势，其能够更准确地把握专利文本的关键信息。值得注意的是，Word2Vec 类方法在 Top@1 的 F1 值与 BERT 类方法的差距不超过 0.01，这表明传统词向量模型在专利文本表征方面仍具有一定的竞争力。

在精确率指标上，基于文档级别的 BERT 嵌入在 Top@1 评估中表现最佳，达到 0.4903，这一结果说明全局语义信息有助于提升分类的精确性。然而，该方法在召回率指标上表现相对较低，这表明仅依赖文档级别的全局信息可能会遗漏部分重要的局部特征，从而影响分类的全面性。从 Top@k 评估的整体趋势来看，随着 k 值的增大，所有方法的召回率均呈现显著提升，而精确率则呈现下降趋势，整体 F1 值也随之降低。其中，基于名词短语的 BERT 嵌入在 Top@5 评估中仍保持最高的 F1 值（0.2417），这进一步证实了该方法在处理专利多标签分类任务时具有更好的鲁棒性和稳定性。

(2) 专利文本多源信息对比

在本研究中，文本特征的信息来源是标题、摘要和第一权利要求，为了检验专利各部分文本组合对 IPC 分类任务的性能影响，对不同文本组合进行消融实验。共构成 7 项对比组合：1）标题；2）摘要；3）第一权利要求；4）标题+摘要；5）标题+第一权利要求；6）摘要+第一权利要求；7）标题+摘要+第一权利要求。实验结果见表 7。

分析实验数据可知，在单一文本来源的对比中，标题虽然与专利的技术领域存在很大关联，但是由于其长度限制，信息量有限，因而表现最为欠佳，其 F1 值区间为[0.1629,0.2530]，各项指标均明显低

于单一的摘要和第一权利要求。而摘要和第一权利要求因包含更多的技术细节，展现出相对较好的性能，F1 值区间分别为 [0.2248,0.2908] 和 [0.2151,0.2802]。同时，由于摘要通常采用自然的描述性语言重点突出发明的技术要点和创新之处，而第一权利要求由于其法律语言的性质，实质性技术信息密度稍低于摘要，因而第一权利要求的性能表现几乎均略低于摘要。

在文本组合 4、5、6、7 的实验中，可以观察到文本信息的互补。其中，摘要和第一权利要求的组合在实验中表现最突出，相较于单一的摘要和第一权力要求在 Top@1 的 F1 分数分别提升了 1.87 个百分点和 2.93 个百分点，它们分别与标题组合的效果虽有提升，分别为 0.0094 和 0.0129，但显然不如这二者组合的提升幅度明显。完整的三文本组合（标题、摘要和第一权利要求全部组合）的性能是最佳的，F1 值区间为[0.2417,0.3127]，证实了文本组合程度与模型性能呈正相关，这一发现在 Top@1 到 Top@5 的评估中都保持一致。

表 7 专利文本来源组合消融实验

文本组合	Top@1			Top@3			Top@5		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
1	0.3854	0.1883	0.2530	0.1636	0.2493	0.1976	0.1145	0.3241	0.1692
2	0.4526	0.2142	0.2908	0.2215	0.3125	0.2592	0.1587	0.3852	0.2248
3	0.4387	0.2058	0.2802	0.2133	0.3133	0.2538	0.1512	0.3725	0.2151
4	0.4658	0.2215	0.3002	0.2256	0.3285	0.2675	0.1625	0.3985	0.2309
5	0.4525	0.2168	0.2931	0.2185	0.3211	0.2600	0.1547	0.3877	0.2212
6	0.4795	0.2285	0.3095	0.2325	0.3504	0.2795	0.1663	0.4046	0.2357
7	0.4862	0.2305	0.3127	0.2365	0.3482	0.2817	0.1699	0.4187	0.2417

(3) 层级分类对比

层次分类的基础是模型在各层次的各个 IPC 的二分类任务的性能，为了直观展现模型训练以后在各个层级各个二分类任务的性能表现，绘制了各层次二分类性能箱线图，见图 6。

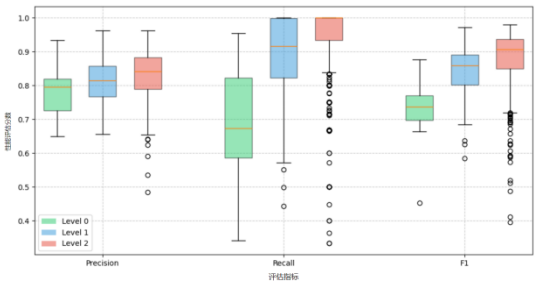


图 6 IPC 各层次二分类性能箱线图

从图中可以观察到从 Level 0 到 Level 2（分别对应 IPC 第一级至第三级）的各项性能指标呈现递增趋势，这一现象可以从任务难度和数据构成两个维度进行深入分析。从任务难度来看，Level 2 在本研究中是最具体的技术类别判断，其负样本来自其他所有三级 IPC 的专利，这些负样本与目标类别的技术领域差异显著，使得分类边界更加清晰；而在 Level 0 层面，需要区分如“物理”和“电学”这样存在大量技术交叉的基础领域，分类边界较为模糊，这种任务本质的差异是性能递增的首要原因。

另一方面，在本研究的数据采集方案中，每个三级 IPC 类别采集约 50 个正样本，这些样本会向上汇聚形成二级和一级类别的数据。当构造元学习任务时，每个任务都采用 N 个正样本和 N 个负样本的平衡配置。对于 Level 2 的任务，从 50 个正样本中抽取 N 个样本，虽然样本总量较少，但这些样本都来自同一个具体的技术领域，特征高度相关且一致。这种高度聚焦的特征分布反而有助于模型学习到更精确的分类边界，这解释了为什么 Level 2 能够获得最好的性能表现。相比之下，Level 0 的一个类别虽然有约 300 个正样本可供选择，看似提供了更大的采样空间，但这些样本来自多个不同的三级类别，技术特征的离散程度较大。当同样的从中抽取 N 个样本时，可能无法很好地代表整个技术领域的特征分布，反而增加了模型学习的难度。

这种任务难度和数据构成的双重影响导致 Level 2 虽然存在一些异常点（可能源于某些“小类”的样本质量问题），但整体性能最优，Level 1 作为中间层级，在特征一致性和任务难度上取得相对平衡，而 Level 0 尽管拥有最多的正样本，但由于特征分布的发散性和任务本身的模糊性，最终展现出相对较低但稳定的性能表现。

为了验证层级分类的有效性，形成了以下三种

实验设置：1）完全不使用层级分类，模型直接在各个三级 IPC 的测试任务中预测测试样本属于该类别的概率，并归纳计算 Top@k 指标；2）模型执行从顶至下的层级分类，但是仅传递标签，不传递模型参数，即对于一个 IPC 类别的某个测试样本，模型判定其为正类，那么在该 IPC 类别的子类中，模型会继续判定该样本属于的类别，否则不必继续判定该样本的类别，同时，在该 IPC 类别适应后的模型不会用作其下级 IPC 预测模型的初始参数；3）完全的层级分类，传递标签且传递模型参数，在某 IPC 类别适应后的模型用作其下级 IPC 预测模型的初始参数。实验结果见表 3.6。

表 8 层级分类消融实验结果

实验设置	Top@1			Top@3			Top@5		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
1	0.4173	0.1877	0.2589	0.1893	0.2644	0.2206	0.1366	0.3226	0.1919
2	0.4546	0.2090	0.2864	0.2123	0.3039	0.2499	0.1537	0.3684	0.2169
3	0.4862	0.2305	0.3127	0.2365	0.3482	0.2817	0.1699	0.4187	0.2417

通过对比表中数据，可以观察到实验设置 1 作为基线方法，在未采用任何层级分类策略的情况下，各项性能指标均处于相对较低的水平。这种方式虽然简化了预测策略，但由于未能利用 IPC 分类体系的层级关系，限制了其分类性能。与实验设置 1 相比，实验设置 2 引入从顶至下的分类策略使其各项性能指标上均呈现出显著提升，Top@1、Top@3、Top@5 的 F1 值相对提升分别为 10.6%，13.3%和 13.0%。这表明即使在不传递模型参数的情况下，层级分类的标签传递机制仍能够在一定程度上利用类别之间的层级关系，提前剔除错误答案，从而提升预测的准确性。实验设置 3 作为完全的层级分类策略，在表中各项评估指标上都取得了最优，其中 Top@1 的 F1 值达到 0.3127，较实验设置 2 进一步提升了约 9.2%。该结果表明通过向下传递模型参数，模型能够有效继承上级类别的学习经验，从而在下级类别中表现出更强的泛化能力。综上所述，层级分类策略在 IPC 分类任务中能够有效提升模型的分类性能，证明了其在 IPC 分类任务中的实用价值。

2.1.5 *参数分析*

(1) 元学习内层迭代数目的影响

在元学习框架中，内层迭代是元模型适应新任务的关键过程，其迭代次数直接影响模型对特定任务的学习效果。为了研究元学习模型在专利分类测试任务中的内层优化特性，本研究设置了一组关于内层迭代次数影响的对照试验，通过改变内层迭代次数（范围从 1 次到 30 次），考察模型在 Top@1、Top@3、Top@5 的 F1 分数变化，并且对数据应用了指数平均移动（Exponential Moving Average，EMA）平滑处理，平滑因子设为 0.3，以便更清晰地观察性能变化趋势，实验结果如图 7 所示。

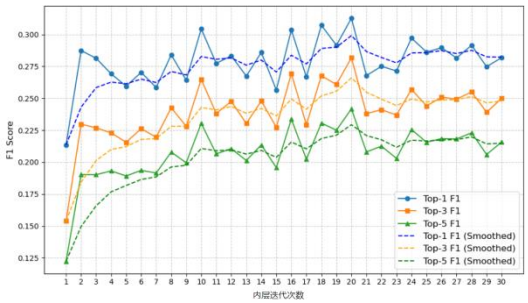


图 7 不同内层迭代次数 Top@k F1 变化

观察图中数据可知，随着内层迭代次数的增加，模型的性能呈现出总体上升但伴随波动的趋势，平滑后的曲线更能体现这一规律。在实验初期，迭代次数 1-5 次时，模型表现出快速学习的特性，F1 分数提升明显，在 5-15 次的中期阶段，提升速度放缓，呈现波动上升态势，超过 15 次后性能趋于稳定，但仍存在一定程度的波动。

值得注意的是，Top@1、Top@3、Top@5 的 F1 分数变化折线呈现出高度相似的变化模式，这种现象反映了模型学习过程的一致性。如果模型在某次迭代中学习到了更好的特征表示或决策边界，这种改进会同时提高模型在 Top@1、Top@3、Top@5 预测的表现。相反，如果模型在某次迭代中遇到了困难样本或优化不稳定，这种负面影响也会同时影响到各个 Top@k 的预测性能。

基于实验结果，本研究选择了 20 次作为最终的内层迭代次数。在这个迭代次数点，Top@1 F1 分数达到约 0.3127，Top@3 达到约 0.2817，Top@5 达

到约 0.2417，都处于较为理想的性能水平。更重要的是，在这个点上，性能增益曲线已经趋于平缓，继续增加迭代次数所带来的收益呈现明显的边际效应。这表明模型已经能够较好地适应任务特征，同时也避免了过度迭代可能带来的计算资源浪费。

(2) 支持集大小的影响

在元学习中，由于模型需要在支持集执行内层迭代适应任务，因而支持集的样本数量直接影响着模型的学习效果和泛化能力。为了在实际应用中更好地平衡模型性能和计算资源，特别是在层级分类这样的复杂任务中，确定合适的支持集大小对于构建高效且可靠的分类系统至关重要。因此，本小节针对不同支持集大小对模型性能的影响展开研究。通过调整 k-shot 学习中的 k 值（即支持集大小）从 1 逐步增加到 30，在 IPC 第三级上评估了模型对各个类别的测试任务的二分类表现。

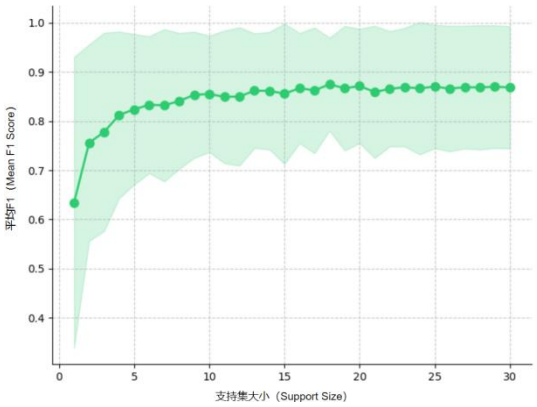


图 8 不同支持集大小平均 F1 变化

图 8 展示了在不同的支持集大小情况下，模型在第三级 IPC 二分类的平均 F1 值的变化情况。在 support size 较小的区间[1,5]内，平均 F1 分数随着支持集样本数量的增加呈现出显著的上升趋势，从初始的约 0.63 快速提升至 0.82 左右。这表明在样本缺乏的情况下，每增加一个支持样本对于模型都是正向增益。当 support size 继续增加到[5,10]的区间时，性能提升速度开始放缓，但仍保持一定的增长。在 support size 达到 10 之后，F1 分数的增长愈加平缓，[10,20]的区间内平均 F1 仅从 0.855 增长到 0.871，并且在[20,30]的区间内在 0.871 附近波动，趋于稳定。这种渐进式的收敛现象表明，过多增加支持集样本数量并不能带来显著的性能提升。

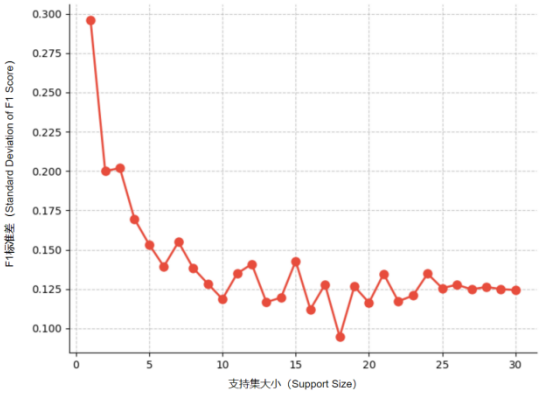


图 9 支持集大小对模型稳定性影响

图 9 展示了不同的支持集大小情况下，图 8 中各 F1 值的标准差变化。从图中可以观察到模型性能稳定性的变化趋势。在 support size 较小时（特别是在[1,5]的范围内），标准差相对较大，最高达到 0.3 左右，表明此时模型在不同的 IPC 类别的二分类性能波动较大，预测结果不够稳定。随着 support size 的增加，标准差呈现出明显的下降趋势，并在 support size 大于 20 后基本稳定在 0.125 左右，这说明较大的支持集，能够提升模型预测的稳定性。

基于两张图的实验结果综合分析，在 support size 位于[10,20]区间时能够达到最优的性能-成本平衡。该范围内的支持集大小不仅确保了模型具有接近 0.85 的优秀 F1 分数表现，同时也实现了约 0.125 的较低标准差，体现出良好的预测稳定性。当 support size 超过 20 后，无论是 F1 分数的提升还是标准差的下降都趋于平缓，继续增加支持样本数量所带来的边际收益已经微乎其微，反而会造成额外的计算开销和资源浪费。

2.2 基于 SAO 的专利相似性判定方法

2.2.1 数据集

本研究的数据来源于 PatentMatch 公开数据集，该数据集构建的基础是欧洲专利局用于文本分析的 EP 全文数据。在这些检索报告里，专利审查员会依据专利申请权利要求的新颖性和创造性判断标准，标记出引用的现有技术文档中的相关段落，这为数据集的构建提供了关键的标注信息。通过对这些数据进行一系列复杂的数据处理，形成了包含 6259703 个样本的数据集，每个样本均由专利申请的权利要求文本、引用的段落文本以及表明引用类型（“X”文档或“A”文档）的标签构成，其中“X”标签表示标记为“X”的专利文档与被审查专利申请的权利要求存在高度匹配的段落，即相关性较高的正样本；“A”标签代表标记为“A”的专利文档不影响被

审查专利申请权利要求的新颖性，仅作为技术背景存在，即负样本。鉴于 PatentMatch 完整数据集的数据规模庞大，为了更高效地开展研究，本研究对其进行了抽样，构建了便于实验的小规模数据集。经过筛选，最终确定训练集包含 34788 个样本对，测试集包含 6956 个样本对。训练集与测试集的样本数量比例约为 5:1。同时，为确保数据的有效性和代表性，训练集和测试集内的正负样本对比例均保持均衡状态，从而为后续的分析和模型训练提供可靠的数据基础。

表 9 数据集样本统计

数据集	正样本数	负样本数	总样本数
train	17394	17394	34788
test	3478	3478	6956

2.2.2 评估指标

在二分类任务的性能评估中，混淆矩阵是一个重要的工具，它能够直观地展示分类模型的预测结果与真实标签之间的对应关系。如图 10 所示，混淆矩阵由四个基本元素构成：真正例（True Positive, TP）、假正例（False Positive, FP）、真反例（True Negative, TN）和假反例（False Negative, FN）。具体而言，真正例表示模型正确预测为正类的样本数量，假正例表示模型错误预测为正类的样本数量，真反例表示模型正确预测为负类的样本数量，假反例表示模型错误预测为负类的样本数量。

预测

实际		Positive	Negative
	Positive	TP	FN
	Negative	FP	TN

图 10 混淆矩阵

基于混淆矩阵，可进一步衍生出多个关键的评估指标。准确率作为最为直观的衡量指标，其含义为模型预测正确的样本在总样本中所占的比例，计算公式为：

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}. \quad (4)$$

精确率衡量的是模型预测为正类的样本中，实际确实属于正类的比例，计算公式为：

$$Precision = \frac{TP}{TP + FP}. \quad (5)$$

召回率则衡量的是实际为正类的样本中被模型正确预测为正类的比例，计算公式为：

$$Recall = \frac{TP}{TP + FN}. \quad (6)$$

除了上述指标外，本节还采用了 F1 作为综合评价指标。该指标综合考虑了精确率与召回率，其计算方式与 2.1.2 小节中定义的 F1 值一致。

2.2.3 实验设置

本实验所用到服务器、操作系统和框架同 2.1.3 节。模型训练采用 Adam 优化器，学习率设置为 1e-4，批量大小（batch size）为 32，训练轮数（epochs）为 100。模型采用孪生神经网络结构，具体而言，每个分支先后由两个线性层、一个多头注意力层组成，最后对两个分支的输出进行平均池化操作。模型的多头注意力机制头数设置为 3，dropout 概率为 0.2。对比损失函数的 margin 值设置为 1.2。

2.2.4 消融实验

本小节首先探讨本研究提出的基于 SAO 的注意力融合孪生神经网络是否有效。图 11 展示了绘制的 AUC-ROC 曲线，用于衡量模型在不同分类阈值下的性能。模型输出的专利对欧氏距离通过设置不同阈值转换为分类结果，欧氏距离小于该阈值的专利对被判定为相似，而欧氏距离大于该阈值的专利对被判定为不相似。根据不同阈值下的分类结果，计算模型的真正率（True Positive Rate，TPR）和假正率（False Positive Rate，FPR），并将其绘制成 ROC 曲线。该图通过两种方式评估模型的分类性能，一方面，曲线接近左上角，表明模型在多数阈值下能够有效地区分正负样本，具有较强的分类能力；另一方面，通过计算 ROC 曲线下的面积（AUC 值），AUC 值越接近 1，表示模型的性能越优越；若 AUC 值接近 0.5，则表示模型的分类效果与随机猜测相似。实验结果显示，本研究提出的模型的 ROC 曲线接近左上角，且 AUC 值达到 0.99，表明模型能够有效地进行专利相似性判定。

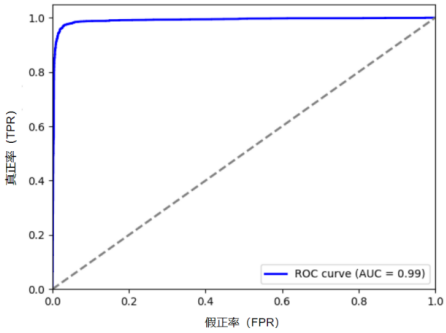
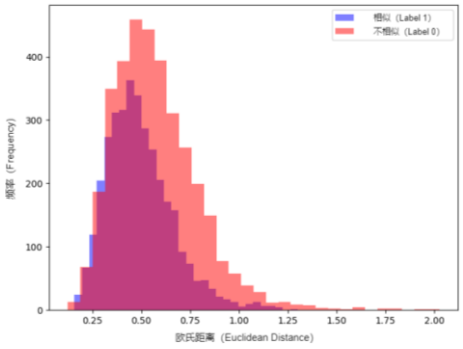
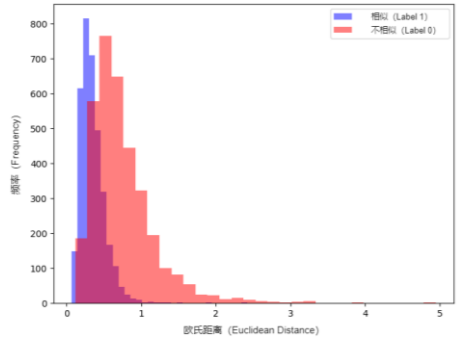


图 11 AUC-ROC 曲线

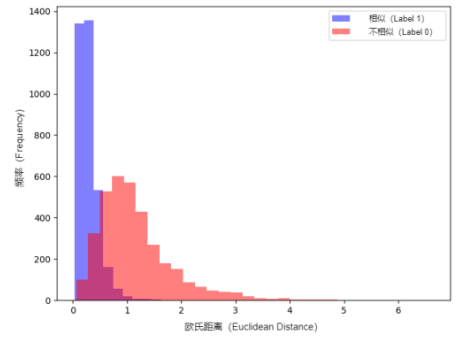
图 12 展示了不同训练轮次下模型对测试集样本输出的欧式距离直方图。欧氏距离的分布用两种不同的颜色区分：蓝色表示相似样本对（标签为 1），红色表示不相似样本对（标签为 0）。横轴表示欧氏距离，纵轴表示每个欧氏距离的样本数量。在 Epoch 1 的直方图中，相似样本的欧氏距离几乎被不相似样本的欧氏距离覆盖，表明模型在训练初期尚未学到明显的区分能力。随着训练的进行，Epoch 5 和 Epoch 11 的直方图显示，相似样本的欧氏距离逐渐集中在较小的值范围，而不相似样本的欧氏距离则开始向较大的值范围迁移，模型的分类能力逐步提高。到了 Epoch 100，模型的欧氏距离分布变得更加明确，相似样本的欧氏距离明显低于不相似样本，表明模型在后期的训练中已经能够很好地将相似和不相似的样本区分开来。



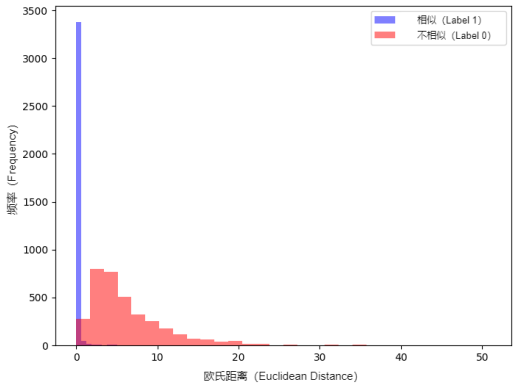
Epoch 1



Epoch 5



Epoch 11



Epoch 100

图 12 欧氏距离直方图

在模型详细设计小节中，介绍了本研究提出的专利相似性判定方法的三部分组成，在上文中已经验证所提出的架构整体的可行性。为了验证 SAO 结构、注意力机制与孪生神经网络各部分在专利相似性判定任务中的有效性，展开以下研究实验：

(1) 嵌入方式对比

为了验证 SAO 结构的有效性，以下将专利的表征方式从 SAO 嵌入替换为其他方式，形成 4 种模型设置：1）模型中将 SAO 替换为名词，使用 BERT 获取这些名词的嵌入；2）在模型中将原本的 SAO 嵌入序列替换为使用 BERT 对专利文本生成一个整体向量表示，由于一份专利的文本仅生成一个整体向量，因而无法再基于这一个向量使用 Attention，孪生神经网络的分支需要从多头注意力切换为 MLP；3）在模型中将 SAO 嵌入方式从 BERT 替换为 Word2Vec；4）模型输入使用 SAO 结构，嵌入使用 BERT，即本研究模型的设置。

表 10 嵌入方式对比

模型编号	嵌入方式	Accuracy	Precision	Recall	F1
1	名词+BERT	0.8933	0.8284	0.9922	0.9029
2	BERT 句子嵌入	0.9618	0.9362	0.9911	0.9629
3	SAO+Word2Vec	0.9616	0.9331	0.9945	0.9628
4	SAO+BERT	0.9645	0.9452	0.9862	0.9652

对比模型 1 和模型 3、4 可见，基于 SAO 三元组的结构化表征相较于孤立名词提取具有显著优势，准确率、精确率和 F1 均有大幅提升。进一步对比模型 3 与模型 4 可见 BERT 的上下文感知能力相较于 Word2Vec 静态嵌入能够带来 1.21 个百分点的精确率增益。模型 2 直接对专利文本生成整体向量的做法，语义捕捉能力较强，但由于其特征来自于专利整段文本，存在许多无关文本干扰，最终在准确率、精确率和 F1 值上略逊于本研究提出的

SAO+BERT 方案。

(2) SAO 提取规则对比

在 1.2 节中介绍了本研究提出的 SAO 提取规则，为了验证各个规则的有效性，形成 5 种提取规则组合：1）仅使用基础的主谓宾提取方式，只包含直接的主谓宾结构；2）基础主谓宾+定语从句；3）基础主谓宾+定于从句+介宾补足语；4）基础主谓宾+定于从句+介宾补足语+状语从句；5）基础主谓宾+定于从句+介宾补足语+状语从句+被动语态，这是完全的 SAO 提取方法，包括前述的所有语法结构并加入了被动语态的处理。

图 13 展示了前述 5 种提取规则组合的性能表现。随着提取规则的增加，Recall 的分数有增有降，整体在 98.5%左右上下浮动，虽然没有明显的提升，但保持在较高的水平，说明规则复杂度增加带来的更多的特征信息并没有干扰模型对于正样本对的判断力。Precision 从基础主谓宾的 91.07% 逐步提升至完全 SAO 提取方式的 94.52%，而由于 Recall 变化较小，相应的 F1 分数也在不断稳步提升，从 94.72%提升至 96.52%，这表明本研究所提出的 5 项提取规则对于模型的判断力是正向增益。另外，将组合 1 与表 4.7 中模型 1 各项指标的对比分析，组合 1 的 Precision 高了近 10 个百分点，说明在专利相似性判定的 0/1 二分类任务里，即便仅采用基础主谓宾的 SAO 表征方式，其展现出的能力也强于专利的名词嵌入，体现了 SAO 表征在该任务中的优势。

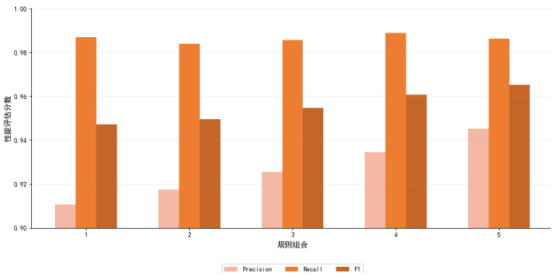


图 13 SAO 抽取规则组合性能对比

(3) 注意力机制消融

为了验证注意力机制的必要性，将从两方面进行对比，一是在专利相似性判定模型中使用 Attention 机制与不使用 Attention 机制对比；二是在专利相似性判定模型中使用自注意力机制和多头注意力机制进行对比。形成三种模型设置：1）模型中不使用 Attention 机制，仅使用 MLP；2）模型使用 Self-Attention 机制；3）模型使用 Multi-head Self-Attention 机制，即本研究模型的设置。

表 11 注意力机制消融对比

模型编号	模型设置	Accuracy	Precision	Recall	F1
1	MLP	0.8781	0.8072	0.9934	0.8907
2	自注意力机制	0.9589	0.9335	0.9882	0.9601
3	多头注意力机制	0.9645	0.9452	0.9862	0.9652

由表可知，不使用注意力机制，仅使用 MLP 时，模型达到 87.81%的准确率和 80.72%精确率，显著低于含注意力的模型，且 F1 值 89.07%最低，这表明 MLP 难以捕捉专利文本间的深层语义关联，其 99.34%的高召回率可能源于对负样本的过度包容，即将部分不相似专利对误判为相关。引入自注意力后，准确率提升 7.08 个百分点，F1 值达到 96.01%。精确率的显著改善说明注意力机制能有效聚焦关键 SAO 结构。进一步采用多头注意力时，模型取得最优能，相较于自注意力，其精确率提升 1.17 个百分点，验证了多头机制通过并行捕捉技术特征，增强了特征表征能力。值得注意的是，随着注意力机制的引入，召回率呈现轻微下降趋势（模型 1→3：99.34%→98.62%），说明模型在提升判别精确性的同时，对潜在正样本的覆盖能力略有妥协，但整体 F1 值的持续增长表明该权衡具有正向收益。

(4) 孪生神经网络消融

为了验证孪生神经网络的有效性，本研究设计了单分支替代方案作为对比模型，该模型摒弃孪生神经网络的双分支结构，将两篇专利的注意力嵌入向量拼接后，通过全连接层映射至二分类，采用交叉熵损失进行分类训练，见表 12 模型 1。

表 12 孪生神经网络消融对比

模型编号	模型设置	Accuracy	Precision	Recall	F1
1	MLP +交叉熵损失	0.9579	0.9633	0.9520	0.9576
2	孪生神经网络 +对比损失	0.9645	0.9452	0.9862	0.9652

从表中数据来看，孪生神经网络在专利相似性判定任务中存在一定优势。从表 12 可以看出，采用孪生神经网络结合对比损失的模型 2 在 Accuracy、Recall 和 F1 指标上均优于单分支的 MLP 模型，其中 Recall 值达到 98.62%，较模型 1 提升了 3.42 个百分点，表明孪生神经网络在捕捉正样本方面具有更强的能力。虽然模型 1 在 Precision 指标上略高于模型 2，但综合考虑各项指标，模型 2 的整体性能更为优异，这验证了孪生神经网络的双分支结构在专利相似度计算中的有效性，其能够更好地捕捉专利文本之间的深层语义关系，从而提升分类性能。

2.2.5 参数分析

本节将深入探究所提模型的各项参数对模型性能产生的影响。

(1) 注意力头数目

在本次实验里，本文针对不同注意力头数目对模型在专利相似性判定任务中的性能影响展开了对比研究。具体而言，着重考察了注意力头数分别为 1、2、3、4、8 和 16 时，模型在 Precision、Recall 和 F1 指标上的表现，实验结果如图 14。

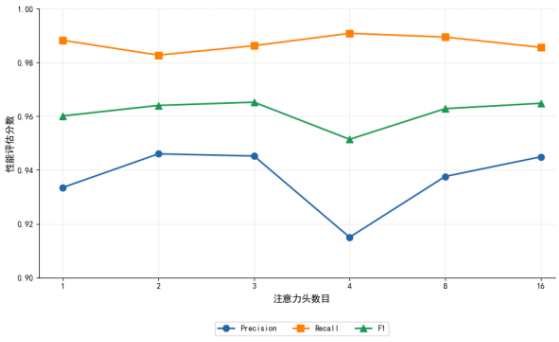


图 14 不同注意力头数目下的模型性能对比

经实验数据分析可知，当注意力头数为 4 时，模型的 Recall 达到了最高值，然而其 Precision 却是所有对比头数中最低的。这一结果表明，在头数为 4 的情况下，模型倾向于将专利对判定为相关，即便部分专利对实际上并不相关。在模型性能综合排名中，头数为 3 和 16 的模型位居前二。尽管头数为 16 的模型在各项评估指标上与头数为 3 的模型仅存在微小差异，但头数为 3 的模型结构更为简单，计算开销更小。因此，本研究最终选择头数为 3 的配置。该配置能够在维持高性能的同时，有效优化计算效率并提升资源利用率。

(2) 对比损失边界参数优化

在本研究中，模型优化采用对比损失函数，而边界参数决定了正负样本对在特征空间中的最小距离，过小的边界可能导致区分性不足，而过大的边界可能导致模型难以收敛。在本次实验里，针对不同的边界参数对模型性能的影响展开了对比研究。着重考察了边界参数为 0.5，1.0，1.2，1.5，2.0，3.0 时的性能指标，详情见图 15，同时绘制了不同参数边界情况下的欧氏距离密度分布图 16 和 ROC 曲线图 17。

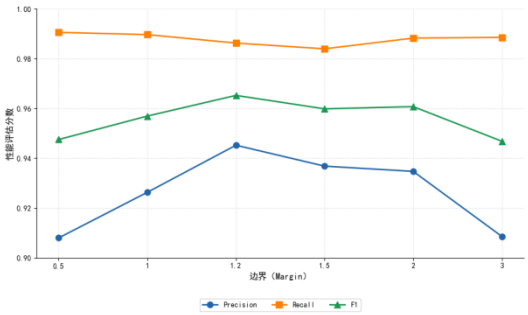


图 15 不同边界参数的模型性能对比

在边界参数的调优过程中，可以观察到模型性能呈现典型的非单调变化特征，见图 15。具体而言，当边界参数在 0.5 至 3.0 的区间内变化时，Precision 指标表现出明显的单峰分布特性，即在边界参数为 1.2 附近达到局部最优值。这种现象可以解释为：当边界参数过小时，模型区分正负样本对的能力有限，导致 Precision 表现欠佳；随着边界参数增大，模型判别能力逐步提升，但当参数超过最优值后，过大的边界约束可能导致模型过度拟合，致使分类精度下降。值得注意的是，不同边界参数设置下 Recall 指标的波动较小，综合考虑 Precision 和 Recall 的平衡性，边界参数为 1.2 时模型取得了最优的 F1 分数。

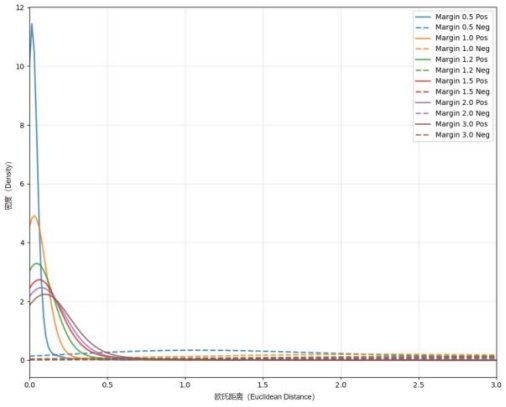


图 16 不同边界参数下欧氏距离低值区域密度分布

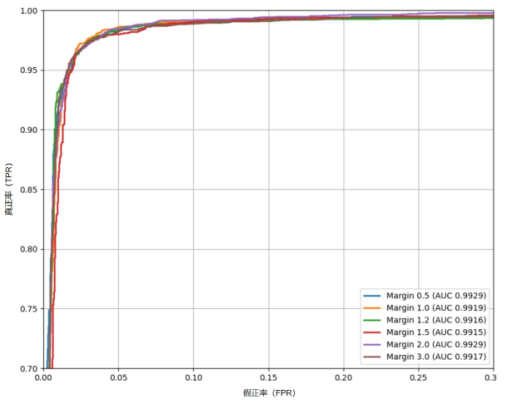


图 17 不同边界参数下的 ROC 曲线对比

如图 16 所示，在边界参数 0.5 至 3.0 的调优范围内，正负样本对的欧氏距离密度分布呈现出显著的特征差异。所有边界参数设置下，欧氏距离的高密度区域均集中在[0,3]范围内，且超过 3 的区域密度呈明显下降趋势，最终趋近于 0。值得注意的是，虽然随着边界参数增大，正样本对的欧氏距离密度峰值呈现右移趋势，但其峰值的横坐标始终在[0,0.5]区间内，且保持明显的单峰分布特征。相比之下，负样本对的密度分布曲线随着边界参数增大逐渐趋于平缓。特别地，在所有边界参数设置下，正负样本对的密度分布均保持明显的区分性，未出现峰值重叠现象，这表明在[0.5,3.0]的参数范围内，模型均具备良好的样本分辨能力。这一结论在图 17 中得到进一步验证，所有边界参数对应的 ROC 曲线均趋近于左上角，且存在显著重叠，AUC 值均维持在 0.99 以上的高水平。尽管边界参数 1.2 的 AUC 值（0.9616）略低于其他四个边界参数，但其 F1 分数是最高的，达到 96.52%。基于上述实验结果与综合分析，本研究最终确定 1.2 作为模型的边界参数设置。