密级：　　保密期限：

北京郵電大學

硕士学位论文

题目：面向司法领域的多标签分类的研究与实现

学　　号：2016110663

姓　　名：杨泽

专　　业：计算机科学与技术

导　　师：张雷

学　　院：计算机学院

2019 年 2 月 22 日

Beijing University of Posts and Telecommunications

Thesis for Master Degree

Student No.:	2016110663
Candidate:	Ze Yang
Subject:	Computer Science and technology
Supervisor:	Lei Zhang
Institute:	School of Computer Science

Feb. 22th, 2019

独创性（或创新性）声明

本人声明所呈交的论文是本人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京邮电大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

申请学位论文与资料若有不实之处，本人承担一切相关责任。

本人签名：_____ 日期：_____

关于论文使用授权的说明

本人完全了解并同意北京邮电大学有关保留、使用学位论文的规定，即：北京邮电大学拥有以下关于学位论文的无偿使用权，具体包括：学校有权保留并向国家有关部门或机构送交论文，有权允许学位论文被查阅和借阅；学校可以公布学位论文的全部或部分内容，有权允许采用影印、缩印或其它复制手段保存、汇编学位论文，将学位论文的全部或部分内容编入有关数据库进行检索。（保密的学位论文在解密后遵守此规定）

本人签名：_____ 日期：_____

导师签名：_____ 日期：_____

面向司法领域的多标签分类的研究与实现

摘 要

中、英文摘要位于声明的次页，摘要应简明表达学位论文的内容要点，体现研究工作的核心思想。重点说明本项科研的目的和意义、研究方法、研究成果、结论，注意突出具有创新性的成果和新见解的部分。

关键词是为文献标引工作而从论文中选取出来的、用以表示全文主题内容信息的术语。关键词排列在摘要内容的左下方，具体关键词之间以均匀间隔分开排列，无需其它符号。

关键词： TEX LATEX xeCJK 模板 排版 论文

ABSTRACT

The Chinese and English abstract should appear after the declaration page. The abstract should present the core of the research work, especially the purpose and importance of the research, the method adopted, the results, and the conclusion.

Key words are terms selected for documentation indexing, which should present the main contributions of the thesis. Key words are aligned at the bottom left side of the abstract content. Key words should be separated by spaces but not any other symbols.

KEY WORDS: TEX LATEX xeCJK template typesetting thesis

目录

第一章　绪论

1.1　研究背景及意义

随着互联网的普及和发展，各行各业都积累了大量的数据，为了应对信息爆炸带来的挑战，迫切需要一些自动化的手段帮助人们解决现实生活中的一些实际问题。如何通过计算机手段，将实际生活中人们的需求转换为计算机可解决的问题，成为我们需要越发关注的问题。

传统的机器学习分类任务中，通常一个实例只包含一种标签，不同类别之间是互斥的，例如：人脸识别中，一个人的头像只能对应一种类别；新闻分类中，一篇新闻文本只对应于一种类型的新闻。而现实生活中的很多问题，往往一个实例对应多种标签，即多标签分类问题，例如：一个歌曲可以被划分为多个不同的流派；一张图片可能包含多种场景对象；生物信息学中，一个未标记的蛋白质含有多重功能；一部电影可能即属于战争又属于历史。然而，在多标签分类研究中经常出现一些问题，比如不同的标签出现的次数可能会相差很多，标签之间存在相互之间的关联，分别称为标签不均衡问题和标签关联性问题。解决这些问题对多标签分类更好地应用于实际生活有重要意义。

在司法领域，根据事实描述判定犯罪者触发了哪些法律是一项非常繁琐的工作，法官通常需要参考一些相关案例来确定具体涉及的法律条文，这项工作是十分耗时并且需要专业知识的。司法判决预测可以根据事实描述进行判决结果预测，这项技术对司法辅助系统是十分有用的，一方面，它可以提供低成本，高质量的司法咨询服务；另一方面，它可以作为法官和律师等专业人士的参考。因此，研究如何自动化地解决判决预测问题是十分重要的。

1.2　国内外研究现状

1.2.1　司法判决预测

作为司法智能化的传统任务，自动判决预测已经被研究了数十年。在早期的研究中，研究者通常将判决文书通过统计分析进行建模[1-5]，这些方法聚焦在如何用数学

的方法对判决文书进行案例分析，而没有将判决预测考虑在内。现有的大多数现有的工作将这个任务当作文本分类问题进行处理，研究者通常通过从文本中抽取有效的特征并利用机器学习方法进行判决预测[6-8]。Liu 等人通过从历史数据中分抽取重要信息，采用 KNN 方法用于决定司法案例的诉讼理由[9]。Latz 等人提出了一个基于树的模型用于预测美国最高法院法官的判决行为[10]。Sulea 等人采用案例描述，规则以及案例时间作为特征信息，开发了一个采用多个 SVM 模型结果进行集成预测的系统[11]。Carvalho 等人针对这个任务提出了一种两步方法，第一步，通过混合 n-gram 模型根据给定事实描述召回若干个法条，接下来，采用机器学习方法决定这些法条中哪些是真正相关的[12]。这些方法需要花费大量的精力进行特征设计，并且很难进行大规模应用。

受到神经网络在各领域成功的启发[13-15]，研究者开始将神经网络技术引入到判决预测任务中。Zhong 等人将任务之间的依赖关系建模成一个有向图，并将这种拓扑结构引入判决预测任务中[16]。由于注意力机制在 NLP 任务中的成功应用，研究者开始将注意力机制引入模型用于处理判决预测，例如，Luo 等人提出一种基于注意力机制的神经网络方法在一个统一的框架下联合建模判决预测任务和相关法条抽取任务[17]。Long 等人利用注意力机制来建模事实描述，辩词，法条之间的复杂语义关系[18]。Hu 等人将几种判别属性引入模型，增强事实描述和罪名之间的关系，提出一种属性-注意力机制罪名预测模型，用于同时预测法条和罪名[19]。

1.2.2 多标签分类

现有的多标签分类算法可以被划分为两个步骤：标签关联利用策略和阈值标定学习。第一个步骤主要用于获取标签之间的关联性，相关工作可以被划分为三类[20]：一阶策略，二阶策略和高阶策略。例如 Boutell 等人将多标签分类问题划分为若干个独立的二分类问题[21]。Brinker 等人提出一种通用扩展方法，克服由于缺乏校准尺度引起的标签排序方法表达能力的限制[22]。Tsoumakas 等人提出了一种针对多标签分类的继承学习方法[23]。在他们的工作中，一种 RAKEL 算法被构造用于考虑标签集合中随机子集之间的影响。Li 和 Guo 提出了一种利用核相关性分析捕获非线性标签相关性并为多标签学习执行非线性标签空间压缩的方法用于多标签分类[24]。Zhai 等人设计了一种以最小排序边界为目标函数的集成学习方法用来构造一个精准的多标签分类器[**Zhai2015A**]。在第二个步骤，一个阈值学习方法被用来决定每个实例的标签集合

大小。例如，Tsoumakas 等人采用固定阈值的方法用来区分每个实例的相关和不相关标签[23]。Yang[25] 和 Fan[26] 分析了在不同条件下几种阈值策略的性能。Elisseeff[27] 和 Zhang[20] 设计了一种线性回归模型用来预测标签集合的大小。

1.2.3 注意力机制

深度学习中的注意力机制模拟人脑在进行图像阅读和文本阅读机制，比如，当我们在看一副画的时候，我们的眼睛聚焦于图像的主体部分，而对背景等信息进行忽略；又比如，我们在阅读新闻的时候，会对新闻中的重点信息进行筛选，而忽略次要信息。这种机制首先在计算机视觉领域被使用[28]。

注意力机制在自然语言处理领域首先被机器翻译任务引入[29]，在他们的工作中，对源语言和目标语言使用注意力机制用于同时进行翻译和文本对齐。Luong 等人扩展了之前的工作，提出了一种全局和局部注意力机制[30]。这些方法都是基于递归神经网络的方法。随着注意力机制的在自然语言处理的广泛应用，研究者开始将注意力机制引入卷积神经网络，Yin 等人在特征图上使用注意力机制以进行后续操作，并取得了良好的效果[31]。

现有的很多研究工作中有很多基于自注意力机制的模型[32, 33]，这种新型注意力机制模型通过自身的交互信息进行注意力加权。Vaswani 等人使用自注意力机制构建了新的模型架构，提出了一种多头注意力机制[34]，得到了很好的效果。

1.2.4 多任务学习

同时学习多个任务的想法是通过利用在不同任务中所包含的不同信息来提高模型泛化性能。该方法广泛应用于各种领域，如计算机视觉[35-37]，自然语言处理[38-42]，基因工程[43, 44]，表达学习[45-47] 等。例如，张等人提出了一种多任务学习架构，它具有四种类型的递归神经层，可以跨多个相关任务融合信息[48]。Sun 等人提出了一种面部识别和通用的联合模型，用于减少人际内部差异，同时扩大人际差异[49]。Wang 等人引入了一个基于多任务学习的框架，用于学习子空间分割的耦合和非平衡表示[50]。提出了一个多任务学习的总体框架，使用递进学习进行句子提取和文档分类[51]。介绍了一种使用多任务学习在 ConvNets 中学习共享表示的基础方法。

1.3 研究内容

本文的研究内容是面向司法领域多标签分类的研究与实现，主要解决在司法领域中挖掘事实描述与涉及的法条之间的关系，采用多标签分类的法条进行法条推荐，为法官裁决和一般民众的法律咨询提供智能化服务的基础。目前，国外已经有一部分针对司法领域犯罪行为与涉及法条之间关系智能化的研究，而国内是司法领域的人工智能研究刚刚开始，还有很高的研究价值。

标签不均衡问题和标签关联性问题是多标签分类中的两个常见问题，它们之间是相互促进又相互影响的，通过标签关联性可以一定程度上缓解标签不均衡问题，标签不均衡问题对标签关联性的获取又有一定程度的阻碍，因此，对这两方面的研究是十分重要的。本文从这两个基本问题出发，主要的研究内容和创新点如下：

1.3.1 基于动态阈值的成对注意力机制模型

司法领域的多标签分类主要面临两个方面的挑战，一个是针对给定相关的法条数量的动态的，我们定义为标签动态问题。另一个是大多数标签很少被命中，我们称为标签不均衡问题。之前的工作通常独立地学习多标签分类模型和标签阈值，并且忽略了标签不均衡问题。为了解决这两个挑战，本文提出了一种通用成对注意力机制模型（**DPAM**）。具体地，**DPAM** 采用多任务学习框架联合学习多标签分类器和阈值预测器，因此 **DPAM** 可以通过学习两个任务的交互信息以提升模型的泛化性能。此外，通过引入基于法条定义的成对注意力模型用来缓解标签不均衡问题。

1.3.2 基于递归注意力机制的模型

法官在进行案件审判的过程中，经常需要来回重复阅读事实描述和法条定义来找到有效信息以便进行正确的匹配（例如，针对给定事实描述，决定与之相关的法条），现有的司法领域多标签分类算法在进行法条推荐的过程中，仅仅分析了相关法条定义的表层语义信息，往往忽略了事实描述和法条之间的重复的交互信息。本文针对这个问题，提出了一种基于递归注意力机制的模型，本模型通过模拟法官交替阅读事实描述和法条定义的过程，引入两者之间的潜在语义信息，从而提升模型性能。

1.4　论文组织结构

本文的组织结构和章节安排如下：

第一章绪论。首先介绍了面向司法领域多标签分类问题的研究背景及研究意义，之后阐述当前国内外司法判决预测及多标签分类的研究现状，接着介绍本文工作的主要研究内容和创新点，最后给出本文的具体章节安排。

第二章相关概念与相关技术。本章节主要介绍本文用到的相关技术以及一些基本概念。首先针对本文研究的问题，详细介绍本文研究问题的定义以及评价指标。确定本文研究问题的输入及输出之后，从三类方法介绍文本分类的基础-文本表示，最后给出本文用到的两种技术，注意力机制以及多任务学习。

第三章是基于动态阈值的成对注意力机制模型。首先介绍算法的动机。之后详细介绍算法的设计，实验数据集，实验结果对比以及实验结果分析。

第四章是基于递归注意力机制的模型。首先介绍算法的实现动机。之后详细阐述算法的设计，实验数据集，实验结果对比以及实验结果分析。

第五章是总结与展望。这一章针对本文的内容进行总结，回归研究内容和研究成果，并对未来工作进行展望。

1.5　本章小结

本章首先介绍了面向司法领域多标签分类问题的研究背景，国内外研究现状。之后根据现有研究的不足，提出了本文的研究内容。为了解决现有算法中存在的问题以及模型的不足，本文提出了一种基于动态阈值的成对注意力机制模型和一种基于递归注意力机制的模型。最后描述了本课题整体章节结构。

第二章　相关概念与相关技术

本章主要针对本课题研究中涉及到的一些主要概念和相关技术进行介绍。包括本课题研究的主要问题定义以及评价指标的介绍，以及自然语言处理领域的三类基础文本表示方法的介绍，还有本课题研究内容中的两个重要的基础知识，注意力机制和多任务学习。

2.1　面向司法领域多标签文本分类核心问题

在司法领域，针对法官拟定的裁判文书，推荐可能需要引用的法条，可以为法官的工作提供便利；针对普通用户输入的案情描述，推荐可能涉及的法条，可以为普通民众提供一定的法律支持。本课题主要研究面向司法领域的多标签分类问题，本节从问题定义和评价指标两个方面进行介绍。

2.1.1　面向司法领域多标签分类的问题定义

给定事实描述，对于法官来说，一种通用的处理方法是他们首先浏览所有的法条，之后根据对案情描述的分析，得到被告人触犯了哪些法条。由于法条数量多而繁杂，因此快速定位相关法条是非常困难的。如表2–1所示，事实描述是一段描述被告人的案情介绍或者事实证据，该被告人触犯了刑法第二百六十四条以及刑法第二百三十二条。本课题面向司法领域的多标签分类问题是以事实描述为输入，预测被告人可能涉及了哪些法条。

2.1.2　面向司法领域多标签分类的评价指标

多标签分类问题与单标签分类问题不同，给定一个样本，其对应的标签可能存在多个，因此在进行多标签分类评估时，需要整体考虑不同标签的影响。本课题采用基于宏平均的方法和杰卡德相似系数两种评价指标进行结果评估。

表 2–1: 判决文书样例

事实描述:	被告人温某 *1* 供称因孙某某（被害人，女，殁年 *47* 岁）借钱不还，遂产生杀害孙某某之念。*2015* 年 *6* 月 *18* 日 *20* 时左右，温某 *1* 携带尖刀前往孙某某位于瓦房店市。期间，温某 *1* 持刀切割孙某某颈部，致其因创伤失血性休克死亡。嗣后，温某 *1* 从孙某某处窃取价值人民币 *1792* 元的金耳环；同时，为掩饰罪行，温某 *1* 将孙某某住处的监控主机拿走，丢弃于一水井内。*2015* 年 *9* 月 *7* 日，温某 *1* 到瓦房店市公安局投案，并如实供述自己的犯罪事实... 综上，根据被告人温某 *1* 的犯罪事实、情节及对社会的危害程度以及其犯罪行为给附带民事诉讼原告人造成的物质损失，依据《中华人民共和国刑法》...
相关法条:	**刑法第二百六十四条**: 盗窃公私财物，数额较大的，或者多次盗窃、入户盗窃、携带凶器盗窃、扒窃的，处三年以下有期徒刑、拘役或者管制，并处或者单处罚金；数额巨大或者有其他严重情节的，处三年以上十年以下有期徒刑，并处罚金；数额特别巨大或者有其他特别严重情节的，处十年以上有期徒刑或者无期徒刑，并处罚金或者没收财产。 **刑法第二百三十二条**: 故意杀人的，处死刑、无期徒刑或者十年以上有期徒刑；情节较轻的，处三年以上十年以下有期徒刑。

2.1.2.1 基于宏平均的方法

现有的二元评价指标以宏平均和微平均为基础，可以将样本根据真实类别和预测类别划分为：

- 真正例（True Positive，TP）：真实类别为正例，预测类别为正例。
- 假正例（False Positive，FP）：真实类别为负例，预测类别为正例。
- 假负例（False Negative，FN）：真实类别为正例，预测类别为负例。
- 真负例（True Negative，TN）：真实类别为负例，预测类别为负例。

定义准确率 (P) 和召回率 (R), 及 $F1$ 如下：

$$P = \frac{TP}{TP + FP}$$

$$R = \frac{TP}{TP + FN}$$

$$F1 = \frac{2 \times P \times R}{P + R}$$

针对一般的二分类问题，可以用以上指标进行评价，而针对多标签分类问题，每个样本包含多个标签，即每个标签都是一个二分类问题，因此，我们需要综合考虑所有标签上的结果，本课题采用宏平均的方式对实验结果进行评价，计算结果如下：

$$Macro\text{-}P = \frac{1}{n}\sum_{i=1}^{n} P_i$$

$$Macro\text{-}R = \frac{1}{n}\sum_{i=1}^{n} R_i$$

$$Macro\text{-}F1 = \frac{1}{n}\sum_{i=1}^{n} \frac{2 \times Macro\text{-}P \times Macro\text{-}R}{Macro\text{-}P + Macro\text{-}R}$$

其中，n 代表多标签分类中标签的总个数，P_i，R_i 分别代表第 i 个标签的准确率，召回率，由每类标签所有样本的真实类别和预测类别进行计算。

2.1.2.2　杰卡德相似系数

杰卡德相似系数是另一种常见的多标签分类评价指标，它用来评价真实标签集合和预测标签集合之间的相似度，它由两个标签集合的交并比定义，其定义如下：

$$Jaccard = \frac{1}{|X|}\sum_{i=1}^{|X|} \frac{|Y^{(i)} \cap Y_{test}^{(i)}|}{|Y^{(i)} \cup Y_{test}^{(i)}|}$$

其中，$|X|$ 代表测试集样本个数，Y 和 Y_{test} 分别代表预测标签集合和真实标签集合，i 代表测试集中的第 i 个样本。

2.2 文本表示方法

文本表示方法作为文本处理的一个核心内容，一直以来都是学者们的研究重点，更有效的文本表示方法对于提高算法性能起到了至关重要的作用。现有的文本表示方法主要分为以下三种类型:

2.2.1 基于空间向量的表示方法

文本表示最直观的方式是空间向量表示法。这类方法将文本表示成文本表示为词组成的向量，向量的每一维代表一个词的词频，向量的维度对应词表的大小，针对“丈夫借名买车，妻子离婚，丈夫偿还。”这句话，根据预定义字典：丈夫:0，买车:1, 借名:2, 偿还:3, 分割:4, 夫妻:5, 妻子:6, 抢劫:7, 支:8, 离婚:9, 可以得到文本向量表示如下：

$$[2, 1, 1, 1, 0, 1, 0, 0, 0, 1]$$

这种方式仅考虑文本中的词频信息，并不考虑不同词在文本中的重要程度，因此无法有效表征文本。为了解决这个问题，之后出现了 $TF-IDF$ 方法，$TF-IDF$ 方法由两部分组成，即 TF 和 IDF：

$$\begin{aligned} TF(w_{ij}, d_j) &= count(w_{jk}, d_j) \\ IDF(w_j k, d_j) &= log(\frac{N}{df_{w_{jk}}}) \\ TF\text{-}IDF(w_{kj,d_j}) &= TF(w_{jk}, d_j) * IDF(w_{jk}, d_j) \end{aligned} \tag{2–1}$$

其中 d_j 代表语料库中的第 k 个文档，w_{jk} 代表第 j 个文档中的第 k 个词，$df_{w_{jk}}$ 代表词 w_{jk} 在整个语料库中出现的文档个数。那么词 w_{jk} 的 $TF\text{-}IDF$ 值即为该词的 TF 值与 IDF 值之积。由上述公式可以看出，TF 值表示某个单词在当前样本中的贡献程度，IDF 值代表某个单词在整个语料库中不同样本的贡献程度，某些词在当前样本中出现的次数很高，即 TF 值很高，同时这些词在不同文档中出现的次数也很多，即 IDF 值很低，那么这些词对不同文本的区分度就没有那么明显，所以 IDF 值能很好弥补 TF 只关注局部的缺点。

$TF\text{-}IDF$ 这类向量空间表示法将各词之间架设为线性无关的，这造成了这类算法无法进行语义相关的判断。此外，向量的维度对应词表的大小，因此，向量维度随着

词表增大而增大，由于样本的中所包含的词远远少于词表，因而造成了向量的高度稀疏。

2.2.2　基于主题模型的表示方法

为了更好地解决语义表示能力，研究者提出了潜在语义分析（LSA）方法[52]，LSA构建文档和词的共现矩阵，通过奇异值分解对原始矩阵降维，可以得到文档向量和词向量，假设 A 是 word-document 矩阵，矩阵每列对应一篇文章，每行是一个单词。$B = A^T A$ 是 Document-Document 矩阵，如果文档 i 和文档 j 有 b 个相同的单词，则 $B[i,j] = b$。$C = AA^T$ 是 Term-Term 矩阵，如果单词 i 和单词 j 同时出现在一篇文档中的频率是 c，则 $C[i,j] = c$。对 A 进行 SVD 分解：

$$A = U \sum V^T \tag{2–2}$$

其中 U 是由矩阵 B 的特征向量构成，V 是由矩阵 C 的特征向量构成，$\sum$ 是由矩阵 B 的特征值的平方根构成的对角矩阵。保留 $\sum$ 的前 K 个特征值，对矩阵 U、V 保留前 K 项：

$$A_{m\times n} = U_{m\times k}\Sigma_{k\times k}V^T_{k\times n} \tag{2–3}$$

根据上式，我们得到 Term-Term 与 Document-Document 在潜在语义空间的相关性矩阵表示：

$$\begin{aligned} A_k A_k^T &= (U_k\Sigma_k V^T)(U_k\Sigma_k V^T)^T = U_k\Sigma_k V^T V_k \Sigma_k^T U_k^T = U_k\Sigma_k\Sigma_k^T U_k^T \\ A_k^T A_k &= (U_k\Sigma_k V^T)^T(U_k\Sigma_k V^T) = V_k\Sigma_k U^T U_k \Sigma_k^T V_k^T = V_k\Sigma_k\Sigma_k^T V_k^T \end{aligned} \tag{2–4}$$

因此，将 $U_k\Sigma_k$ 矩阵的行看作是 term 在语义空间的表示，将 $V_K\Sigma_k^T$ 的行看作是 Document 在语义空间的表示。LSA 算法原理简单，一次奇异值分解就能得到向量空间表示，同时解决了词义相似性的问题。*LSA* 算法通过线性代数中的奇异值分解实现文档映射到低维语义空间里的向量，但是空间中的每个维度没有明确物理含义。

在此之后，还有 pLSA 模型[53] 假设文档具有主题分布，尝试从概率生成角度进行文本表示，这类方法不是本文研究重点，因此此处不进行详细描述。

2.2.3　基于神经网络的表示方法

随着神经网络技术的发展，Bengio 等人尝试使用神经网络进行自然语言建模[54]，在此基础上，Mikolov 等人提出了 Word2Vec 方法[55]，采用 CBOW 和 Skip-gram 方法，

进行语言模型建模，CBOW 使用上下文对目标词进行预测，Skip-gram 通过目标词对上下文进行预测，在此基础上得到的词语义文本表达，在自然语言的各个领域，取得了非常好的效果。fastText[56] 根据 word2vec 思想，对文本进行建模实现分类。其模型如图 2–1所示：

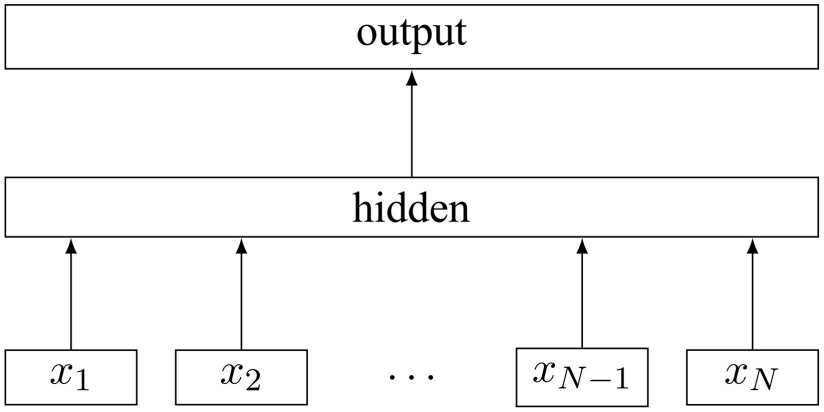

图 2–1: fastText 模型结构图

该模型由输入层，隐藏层和输出层组成，输入层为词向量，隐藏层通过将文本中词向量进行平均得到句子向量的表达，最后通过暑促层线性分类器进行文本分类。该方法训练速度快，在很多文本分类任务中取得了出色的表现。

随着 CNN/RNN 的研究不断增多，越来越多的模型使用这些方法对文本进行建模。Kim 等人提出基于卷积神经网络的文本分类方法[13]，其模型结构如图 2–2所示：

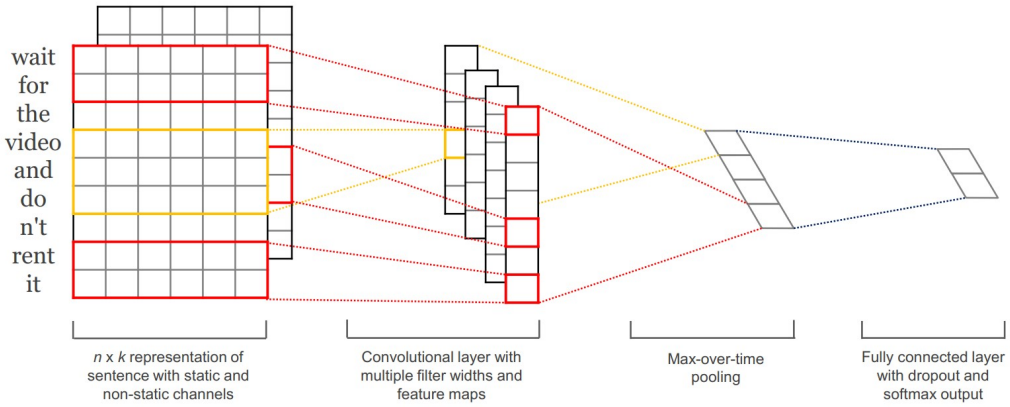

图 2–2: TextCNN 模型结构图

该模型由卷积层，MaxPooling 层，全连接层组成，该模型通过不同大小的卷积核提取不同窗口 N-gram 特征，经过 MaxPooling 将不同窗口特征拼接作为文本表达，

由于 CNN 擅长提取局部特征，仅能获取局部信息，会损失一些语义信息。针对 CNN 的不足之处，研究者们提出了基于 LSTM 的神经网络，用来获取长距离依赖信息，通过将 LSTM 的隐层向量进行平均，得到文本表达。

2.3　注意力机制

注意力机制是一种广泛应用在深度学习的各个领域的机制，其本质是一种特征加权的方式，注意力在自然语言处理领域的应用最早是在机器翻译领域[29]，传统的机器翻译模型通常由编码器-解码器结构组成，编码层即文本表示层，采用文本表示模型对文本进行编码，将输入语句转换成文本表示向量，之后通过解码器对该向量进行解码，将该向量翻译成目标语言。最基本的编码器-解码器结构由 RNN 构成，RNN 由于本身结构的限制，无法得到文本序列的长期依赖，因此，为了解决这个问题，机器翻译模型引入了注意力机制，通过在不同解码阶段不同词重要性的不同，对文本向量进行加权，将语义信息集中在需要的部分。最基本的注意力机制模型如图 2–3所示：

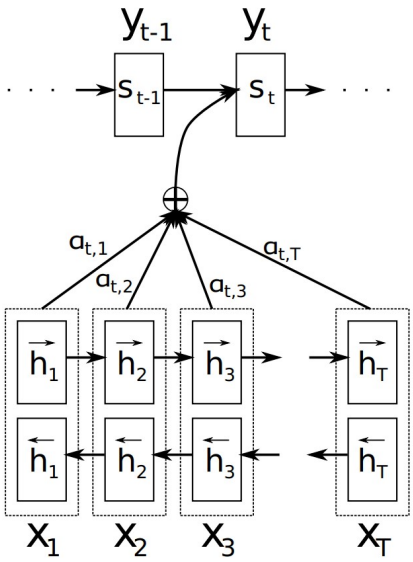

图 2–3: 注意力机制模型结构图

其中，X 代表输入文本，y 代表输出目标，定义如下条件概率：

$$p(y_t|y_1,\ldots,y_{t-1},x) = g(y_{t-1},s_i,c_i) \tag{2–5}$$

其中，s_i 代表第 i 时刻的 RNN 隐层单元，其计算公式如下：

$$s_t = f(s_{t-1}, y_{t-1}, c_i) \tag{2–6}$$

c_i 依赖于来源于输入句子的隐藏序列 $(h_1, \ldots, h_{T_x})$，c_i 由 h_i 加权求和得到：

$$a_{ij} = \frac{e_{ij}}{2\sum_{k=1}^{T_x} exp(e_{ik})} \tag{2–7}$$

其中，$e_{ij} = a(s_{i-1}, h_j)$，由前一时刻 s_{i-1} 和当前时刻隐层状态 h_j 构成。它是一个对其模型，代表输入数据位置 j 周围并且输出在位置 i 的匹配程度。

通过计算注意力机制，可以计算得到编码状态和解码状态之间的关联性权重，从而得到对当前输出位置比较重要的输入位置信息。本课题的研究同样使用了注意力机制用于支持判决预测任务。

2.4 多任务学习

多任务学习已经成功应用于机器学习的各个部分。多任务学习通过训练过程中包含在相关任务中的特定领域信息，提高模型的泛化性能。在自然语言处理深度学习领域，多任务学习通常包含隐层参数硬共享和隐层参数软共享两种。

2.4.1 参数硬共享

在硬参数共享是神经网络中最常用的方法。它通常用来在所有任务之间共享隐层参数，仅仅保持不同任务的输出层不同，其模型结果如图 2–4：

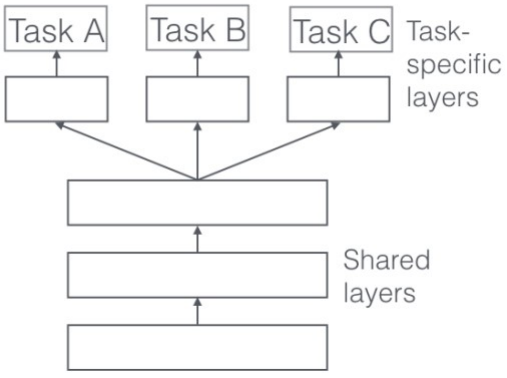

图 2–4: 参数硬共享模型结构图

参数硬攻向通常能很大程度上避免过拟合问题。我们同时学习的任务越多，我们的模型就需要学习到能代表所有任务的特征表达，因此我们的原始任务模型过拟合的几率就会越小。

2.4.2　参数软共享

在参数软共享模型中，每个模型都由自己的参数。不同模型参数之间的距离通过正则化来让共享层参数更加接近，其模型结构如图 2–5：

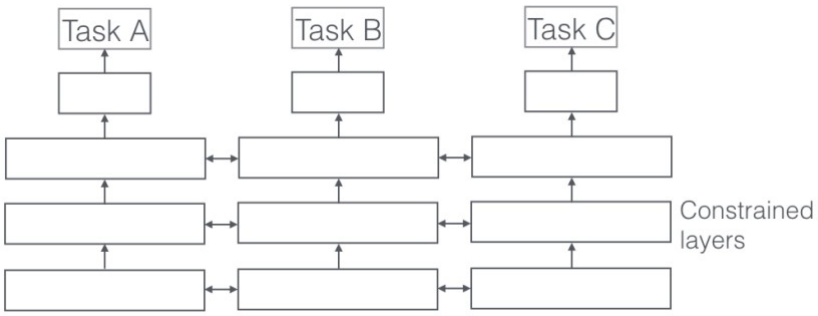

图 2–5: 参数软共享模型结构图

在深度神经网络模型中，软参数共享中的约束层受到正则化的启发，已经用在了很多模型中。

多任务学习在我们训练过程中，有效地增加了训练样本的数量，所有的模型至少有一些噪声信息，当我们针对任务 A 训练模型时，我们的目标是针对任务 A 学习更好的表达，理想状态下忽略了噪声信息和泛化能力。不同的任务拥有不同的噪声模式，一个模型同时学习两个任务，可以学到更多泛化的表达。

2.5　本章小结

在本章节中，主要介绍了与本课题研究内容相关的一些基本概念，包括本课题研究内容问题的定义以及评价方法，还有本课题中用到的一些基本技术，包括文本表示方法，注意力机制与多任务学习。

第三章　基于动态阈值的成对注意力机制模型

本章主要提出了一种基于动态阈值的成对注意力机制模型。首先章节 3.1描述了算法的研究背景；接着，在章节 3.2中介绍了基于动态阈值的成对注意力机制模型结构，包括算法的设计、模型的学习与预测等过程；章节 3.3中介绍实验设置相关内容，对实验数据集，对比方法等进行介绍；章节 3.4 对实验结果进行分析，章节3.5对实验结果进行进一步的分析和讨论，章节3.6对本章内容进行总结。

3.1　引言

在司法领域，基于严格定义的司法法条进行犯罪分类是一项非常繁杂的工作。法官通常需要阅读若干类似的案例针对事实描述给定涉及的法条，这件事是非常耗时并且需要专业知识的。通常来讲，这个任务被当作多标签分类分体，用于提高工作效率和节省人力成本。在本课题中，采用多标签分类处理这个问题，帮助法官快速准确的针对案情描述挑选出合适的法条。

然而，这个问题是非常困难的，面临两个主要问题。一个问题是大多数案情描述涉及的法条数量是不确定的，称为标签动态问题。通过在大规模数据上对 70 个法条进行分析，案列包含的标签数量有很大差异，其结果如图 3–1所示：

另一个挑战是标签不均衡问题。如果一个多标签分类数据集上一部分数据的标签数量远远小于另一部分数据，那么这个多标签分类数据集被认为是不均衡的。针对同一份案例数据进行分析，其结果如图 3–2所示。从图中可以看出，每种法条出现的数量符合长尾分布，这意味着很多法条很少在审判中被引用。大多数传统的多标签分类算法在训练过程中通过最小化整体分类误差来进行优化，这种方式假设所有标签拥有同等的重要性。这种假设使得分类算法在训练过程中偏向于向数量占比多的标签进行学习。虽然法条定义可以体现不同法条之间的一些相关信息用于缓解标签不均衡问题（例如表 3–1所示，刑法第一百九十七条和刑法第一百九十一条是非常相似的。），但是目前在判决预测研究中没有工作考虑这方面的问题。

现有的很多多标签分类工作都引入了标签之间的关联信息，然而，这些工作都将多标签分类和阈值预测器分开学习，并且忽略了标签不均衡问题。为了处理这个问

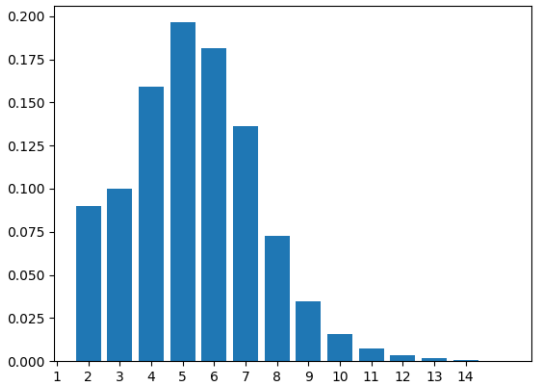

图 3–1: 案例数据上法条集合大小分布于统计。x 轴代表法条集合大小，y 轴代表样本比例占比

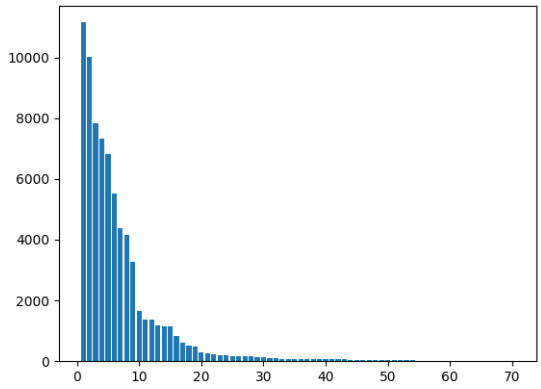

图 3–2: Distribution of article set size over evidences. The x-axis stands for the sorted labels according their frequencies in the dataset, y-axis represents counts of labels.

题，本课题提出了一种多任务学习框架用于联合学习多标签分类模型（简称 **DPAM**）和阈值预测器。针对第二个问题，本课题采用法条描述信息来建模标签之间的成对关系，并针对标签集合构造软注意力机制矩阵，本课题的实验结果证明这种方法能有效缓解标签不均衡问题。

表 3–1: 法条定义中相关法条

刑法第一百九十七条: 使用伪造、变造的国库券或者国家发行的其他有价证券，进行**诈骗**活动，数额较大的，处五年以下有期徒刑或者拘役...
刑法第一百九十一条: 明知是毒品犯罪、黑社会性质的组织犯罪、恐怖活动犯罪、走私犯罪、贪污贿赂犯罪、破坏金融管理秩序犯罪、**金融诈骗**犯罪的所得及其产生的收益，为掩饰、隐瞒其来源和性质，有下列行为之一的，没收实施以上犯罪的所得及其产生的收益...

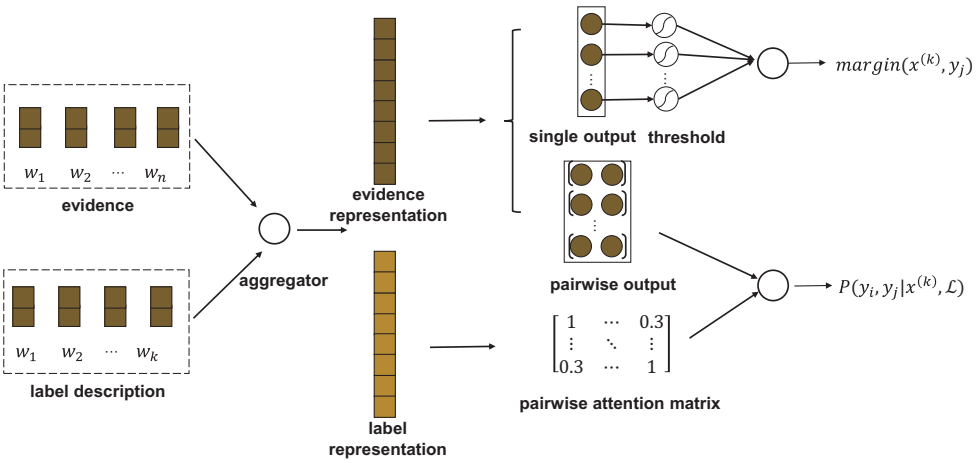

图 3–3: The overall architecture of the proposed Dynamic Pairwise Attention Model (DPAM).

3.2 DPAM 算法设计

在本节中，本课题详细介绍提出的动态成对注意力机制模型。如图 3–3所示是 **DPAM** 模型架构图。具体来讲，本模型包含两部分：一个成对注意力机制模型（简称 **PAM**）用于得到标签关联得分，一个动态阈值预测器得到参考阈值用于决定标签和给定案情描述是否相关。最后，采用多任务学习方法联合学习这两个任务。

3.2.1 成对注意力机制模型

在司法领域，每个案情描述都由一个集合的词组成。本课题采用词袋表示作为输入，并将每个词映射带一个连续的向量空间。之后本课题将所有的词向量进行聚合组成案情表达和标签描述表达。PAM 将标签之间的关联进行考虑。具体来讲，针对每一个训练样本 $(x^{(k)}, Y^{(k)})$，PAM 枚举标签集合 $Y^{(k)}$ 中的成对关系，利用这种关系，本模型可以引入标签关联信息。以 $Y^{(k)} = \{y_1, y_2, y_3\}$ 举例，枚举之后，原始标签集合变为 $\{(y_1, y_2), (y_1, y_3), (y_2, y_3)\}$。

更正式地，用 $\mathbf{V}^I = \{\vec{v}_j^I \in \mathbb{R}^{D_v} | j = 1, \ldots, N\}$ 表示在一个 D_v 维连续空间中的所有词向量。针对每个案情描述和标签定义，本模型采用如下方式将词向量聚合为案情表达和标签描述表达：

$$\vec{v}^{(e,k)} = g(\vec{v}_j^I : j \in x^{(k)})$$
$$\vec{v}^{(l,i)} = g(\vec{v}_j^I : j \in l^{(i)})$$

其中，$g(\cdot)$ 代表聚合函数。在本课题的工作中，使用 TextCNN[13] 用于构造输入向量。给定案情描述 $x^{(k)}$ 和标签定义 $\mathcal{L}$，PAM 考虑成对 $(y_i, y_j) \in Y^{(k)}$ 条件概率如下：

$$P(y_i, y_j | x^{(k)}, \mathcal{L}) = P(y_i, y_j | x^{(k)}) P(l^{(i)}, l^{(j)})$$

其中，$P(y_i, y_j | x^{(k)})$ 和 $P(l^{(i)}, l^{(j)})$ 分别进行计算。

为了解决标签不均衡问题，本课题引入成对标签关联。众所周知，注意力模型在传统的序列建模，例如 LSTM 和 GRU，采用软加权机制对子序列进行加权。为了提高稀疏成对标签的重要性，本课题将传统的注意力机制扩展到成对关系集合中，称为成对注意力机制。给定标签描述表达 $\vec{v}^{(l,i)}$ 和 $\vec{v}^{(l,j)}$，成对注意力矩阵计算如下：

$$P(l^{(i)}, l^{(j)}) = \frac{\vec{v}^{(l,i)} \cdot \vec{v}^{(l,j)}}{\sum_{j=1, j \neq i}^{|C|} \vec{v}^{(l,i)} \cdot \vec{v}^{(l,j)}} \tag{3–1}$$

在本模型中，$P(l^{(i)}, l^{(j)})$ 可以被看做标签集合 $Y^{(k)}$ 中的标签对 (y_i, y_j) 的注意力机制得分。这种机制使得不在样本标签集合中的标签同样影响最后的损失函数，并能增强具有类似描述信息的稀疏成对标签的重要性，这样我们就可以缓解标签不平衡问题。

因此，训练数据中成对标签的后验概率 $P(y_i, y_j|x^{(k)})$ 由 softmax 函数计算如下：

$$P(y_i, y_j|x^{(k)}) = \frac{exp(\vec{v}^{(e,k)}\mathbf{W}\vec{y}^{(i,j)})}{\sum_{\vec{y}^{(i,j)}\in\mathbb{Y}} exp(\vec{v}^{(e,k)}\mathbf{W}\vec{y}^{(i,j)})}$$

其中 $\mathbf{W} = \mathbb{R}^{D_V\times|C|}$ 是交互矩阵，$\vec{y}^{(i,j)}$ 是 $|C|$ 大小的向量，并且 $\vec{y}^{(i,j)}$ 中共现标签位置为 1，其他位置为 0。$\mathbb{Y}$ 是考虑不同标签对 (y_i, y_j) 的所有可能向量。将 PAM 的目标函数定义为所有案情的对数似然函数，其定义如下：

$$\begin{aligned} l_{pam} &= \sum_{x^{(k)}\in X}\sum_{(y_i,y_j)\in E(Y^{(k)})} \log P(y_i, y_j|x^{(k)}, \mathcal{L}) \\ &= \sum_{x^{(k)}\in X}\sum_{(y_i,y_j)\in E(Y^{(k)})} \Big(\log P(y_i, y_j|x^{(k)}) + \log P(l_i, l_j)\Big) \end{aligned} \tag{3–2}$$

其中 $E(Y^{(k)})$ 代表 $Y^{(k)}$ 中成对枚举关系。

最后，PAM 基于学习到的参数 $\mathbf{W}$ 输出针对新实例 $x^{(k)}$ 每个标签 y_i 的概率：

$$P(y_i|x^{(k)}) = \frac{exp(\vec{v}^{(e,k)}\mathbf{W}_{*i})}{\sum_{i=1}^{|C|} exp(\vec{v}^{(e,k)}\mathbf{W}_{*i})}$$

其中，$\mathbf{W}_{*i}$ 代表 $\mathbf{W}$ 的第 i 列。

3.2.2　动态阈值预测器

通过 PAM，每个标签 $P(y_i|x^{(k)})$ 的输出概率通过阈值进行预测最终结果，总的来说，本模型的目标是学习一个决策边界用来决定这个标签是否与给定案情描述相关。直观上来说，如果 $P(y_i|x^{(k)})$ 大于 y_i 的决策边界 t_i，那么标签 y_i 和 $x^{(k)}$ 相关并且 $y_i \in Y^{(k)}$；如果 $P(y_i|x^{(k)})$ 小于 y_i 的决策边界 t_i，标签 y_i 和 $x^{(k)}$ 不相关。特别低，本模型使用下面的函数针对每个案情给定预测标签的置信度：

$$margin(x^{(k)}, y_i) = [P(y_i|x^{(k)}) - t_i)] \cdot Seg(x^{(k)}, y_i) \tag{3–3}$$

其中 $t_i \in \mathbf{T}^{1\times|C|}$ 是针对 y_i 学习到的决策边界. $Seg(x^{(k)}, y_i)$ 是一个分割函数，其定义如下:

$$Seg(x^{(k)}, y_i) = \begin{cases} 1, & y_i \in Y^{(k)} \\ -1, & y_i \notin Y^{(k)} \end{cases}$$

在本模型中，$margin(x^{(k)}, y_i)$ 代表标签 y_i 与案情 $x^{(k)}$ 相关的软边界。$margin(x^{(k)}, y_i) > 0$ 代表 $x^{(k)}$ 被正确分类为 y_i，$margin(x^{(k)}, y_i) < 0$ 代表标签 y_i 和 $x^{(k)}$ 不相关。之后本模型使用下面的函数决定所有样本案例的每个标签：

$$l_{dyn} = \sum_{x^{(k)} \in X} \sum_{y_i \in Y^{(k)}} \log \left[1 + exp(-margin(x^{(k)}, y_i))\right] \tag{3–4}$$

最后，通过结合公式 3–2和公式 3–4，本模型定义多任务学习方法如下：

$$\ell = l_{pam} + l_{dyn} - \lambda \left\| \Theta \right\|_2 \tag{3–5}$$

其中 λ 是正则化系数，Θ 是需要学习的模型参数（例如 $\Theta = \{\mathbf{W}^{D_V \times |C|}, \mathbf{V}^I, \mathbf{T}^{1 \times |C|}\}$）。

3.2.3 模型学习与预测

为了学习 DPAM 模型的参数，本模型采用随机梯度下降进行算法学习。每次迭代，通过公式3–5更新参数。然而，直接通过公式3–2优化由于与 $2^{|C|}$ 成正比的高计算复杂度是非常困难的。因此，本模型采用负采样技术[57] 进行模型优化，新的目标函数定义如下：

$$\begin{aligned} \ell_{NEG} = \sum_{x^{(k)} \in X} \sum_{(y_i, y_j) \in E(Y^{(k)})} \Big(& \log \sigma(\vec{v}^{(e,k)} \mathbf{W} \vec{y}^{(i,j)}) \\ & + n_{neg} \cdot \mathbb{E}_{\vec{y}^{neg} \sim P_y} [\log \sigma(-\vec{v}^{(i)\top} \mathbf{W} \vec{y}^{neg})] + \log P(l^{(i)}, l^{(j)}) \Big) \end{aligned}$$

其中，$\sigma(x)$ 是 sigmoid 函数 $\sigma(x) = \frac{1}{(1+e^{-x})}$，$n_{neg}$ 是负样本个数，$\vec{y}^{neg}$ 是采样向量，它通过所有成对标签组合的经验分布得到。可以看出，带有负采样的 DPAM 模型的目标是增加给定案情的正标签对组合的概率，减少负样本对组合的高铝。之后采用随机梯度下降算法来最大化新的目标函数来进行模型学习。

在训练阶段，本课题发现模型性能的提升不明显。分析得出其原因是由于注意力机制矩阵是由聚合后的词表达进行初始化的。因此在开始的迭代轮次中，注意力机制矩阵变成了模型的噪声。为了获得更好的性能，本课题设计了一种新的学习策略：对前 1000 次迭代，设置 $P(l^{(i)}, l^{(j)}) = 1$，在初始阶段之后，假定已经获得了稳定的词向量表达，之后再通过公式3–1在每次迭代中计算注意力机制矩阵。算法学习的细节如流程图3–1所示：

Algorithm 3–1 Framework of joint learning for our model

Initialize model $\Theta = \{\mathbf{W}^{D_V \times |C|}, \mathbf{V}^I, \mathbf{T}^{1 \times |C|}\}$ randomly
iter = 0
set $n_{burn} = 1000$
repeat
    $iter \leftarrow iter + 1$
    if $iter < n_{burn}$ **then**
        set $P(l^{(i)}, l^{(j)}) = 1$
        for $i = 1, ..., |X|$ **do**
            for instance $x^{(k)}$
            compute the gradient $\nabla(\theta)$ of Equation(3–5)
            update model $\theta \leftarrow \theta + \epsilon\nabla(\theta)$
        end for
    else
        compute $P(l^{(i)}, l^{(j)})$ according Equation(3–1)
    end if
until (Coverage or $t > num$)
return $\{\mathbf{W}^{D_V \times |C|}, \mathbf{V}^I, \mathbf{T}^{1 \times |C|}\}$;

利用学到的参数，判决预测策略如下。针对案情 $x^{(k)}$，标签集合由标签得分是否大于标签阈值决定。预测过程如下：

$$s(Y^{(k)}|x^{(k)}) = \sum_{y_i \in Y^{(k)}} I\Big(\frac{exp(\vec{v}^{(e,k)}\mathbf{W}_{*i})}{\sum_{i=1}^{|C|} exp(\vec{v}^{(e,k)}\mathbf{W}_{*i})} > t_i\Big) \tag{3–6}$$

其中，$I(\cdot)$ 代表索引函数，$s(Y^{(k)}|x^{(k)})$ 是针对案情 $x^{(k)}$，标签集合 $Y^{(k)}$ 中标签的得分。通过公式3–6，针对输入的案情描述，根据前向过程计算每个标签的得分，选取得分大于阈值的作为预测结果。

3.3 实验设置

在本节中，通过实验验证本课题提出的模型的有效性。首先介绍实验设置，之后通过对比试验验证模型效果。

3.3.1 数据集介绍

本课题在两个真实数据集上进行实验验证，数据来源于裁判文书网[1]。裁判文书网是由中华人民共和国政府开放的网站。它记录了中国超过 3000 个法院的裁判文书。在本研究中，本课题收集了 2014 年到 2016 年的 40256 个案由是诈骗和其他少量混合罪名相关的裁判文书。

本课题首先对这些数据进行预处理。本课题删除了一些无效数据，之后从剩余的裁判文书中抽取所有的标签集合和案情描述。经过数据预处理，本课题得到包含 70 个法条标签的 17160 个诈骗类样本，以及包含 30 个法条标签的 4033 个混合数据样本。数据分析如表 3–2所示。最后，本课题将数据以 8 比 2 将数据划分为训练集和测试机两部分。

3.3.2 模型评估

本课题通过和以下模型进行试验对比：

- **POP**: 将出现频次最高的 K 个标签作为测试数据的预测结果（在本模型中，K=5）。

[1]http://wenshu.court.gov.cn/Index

表 3–2: Basic statistics of the two legal case datasets for experiments.

dataset	#evidence	#article	#average evidence length	#average article definition length	#average article set per evidence
Fraud	17160	70	1455	136	4.1
Civil Action	4033	30	2533	123	2.4

- **BSVM**：一种一阶多标签分类模型[27]，在这个模型中，每种标签的预测当作一个二分类问题，之后采用 SVM 作为分类器对每种标签单独进行分类，之后合并每个分类器的结果。
- **ML-KNN**：ML-KNN[58] 是一种常见的一阶多标签分类模型。基于统计的方法针对新的样本，选取与它最近邻的若干个样本的标签作为结果。ML-KNN 采用最大先验原理决定标签集合。
- **BP-MLL**：这种方法[59] 是一种流行的二阶方法。它采用前馈神经网络通过一种新型的成对损失用来进行多标签分类。
- **TextCNN-NLL**：一种二阶多标签分类算法，采用卷积神经网络[13] 作为输入，利用与 BL-MLL 方法一样的损失进行优化。
- **CC**：分类链[60] 是一种新型的链式分类算法，它可以在可接受的计算复杂度范围内建模标签关联信息。

针对 BSVM，ML-KNN，BP-MLL,TEXTCNN-MLL 和 CC，本课题使用公开的 PV 模型[61] 进行案例表示。针对每个模型，本课题在两个数据集上采用维度 $k \in \{64, 128, 192, 256, 320\}$ 进行 20 次重复试验。本课题对实验结果进行平均后与本模型进行对了，实验结果在下面的章节进行分析。

3.4　实验结果

3.4.1　子模型性能验证

首先，本课题在两个数据集上验证模型的有效性。目的是为了验证分别引入注意力矩阵和动态阈值机制是否有效。为了进行比较，本课题采用相同的参数设置，同样使用 320 维特征表达进行实验。

表 3–3: Performance comparison over SPM and SPAM on crimes classification in terms of different evaluation metrics. Improvements of SPAM over SPM on Macro-R, Macro-F1 and Jaccard (when applicable) are significant at $p = 0.05$.

dataset	method	Macro-P	Macro-R	Macro-F1	Jaccard
Fruad	SPM	0.572	0.372	0.430	0.768
	SPAM	0.574	0.390	0.441	0.781
Civil	SPM	0.645	0.322	0.424	0.623
Action	SPAM	0.649	0.340	0.448	0.626

3.4.1.1 注意力矩阵的性能

在本节中，本课题研究注意力矩阵的影响。针对本课题的 DPAM 模型，将动态阈值机制替换为简单的分割点[23, 62] 用来决定给定案例标签集合的大小。本课题将这个模型命名为静态成对注意力机制模型（简称 SPAN）。之后进一步忽略通过标签定义学习到的注意力矩阵的影响（例如，针对注意力矩阵中的元素，设置为固定分数），本课题将消减后的模型称为静态成对模型（简称 SPM）。表 3–3展示了这两种方法的性能对比。从结果中可以得到以下结论：（1）SPAM 几乎在两个数据集上所有的评价指标上好于 SPM，例如，诈骗案由数据集上，Macro-R, Macro-F1 和 Jaccard 的结果分别提升了 1.8%, 1.1%, 和 1.3%。（2）与 SPM 比较，SPAM 在 Macro-P 上的提升非常微小。原因是由于 SPAM 与 SPM 相比可以预测出更多的正确标签，但是它不能恰当的处理阈值问题。在预测阶段，一些置信度不高的标签仍旧有效，因此在 Macro-P 上的性能提升并不明显。

3.4.1.2 动态阈值预测器的性能

本课题进一步分析了模型中动态阈值预测器的性能。针对本课题的 DPA 模型，采用消减策略忽略注意力矩阵中的权重，将新模型命名为动态成对模型（简称 DPM）。本课题将 DPM 与几种常见的阈值策略进行比较，例如，分割点策略和线性机制，实验结果展示在表 3–4中。从实验结果得到一下结论：（1）线性机制[20, 27] 的结果好于 ad-hoc[22, 23] 策略。（2）DPM 性能好于线性机制。以诈骗案由数据集为例，在 Macro-P，Macro-F1 和 Jaccard 上的结果分别提升了 3.1%, 0.5%, and 0.5%。（3）与线性模型相比，

表 3–4: Performance comparisons over Cutting Point, linear model, and DPM on crimes classification in terms of different evaluation metrics.

dataset	method	Macro-P	Macro-R	Macro-F1	Jaccard
Fruad	Cutting Point	0.560	0.371	0.425	0.762
	Linear model	0.573	0.372	0.428	0.767
	DPM	0.604	0.377	0.433	0.772
Civil Action	Cutting Point	0.513	0.201	0.183	0.438
	Linear model	0.393	0.204	0.185	0.435
	DPM	0.653	0.329	0.457	0.613

DPM 在 Macro-R 上性能提升并不明显。原因是动态阈值机制重点在于如何学习一个鲁棒的阈值边界来剔除不置信的标签，因此，它在 Macro-P 上的表现要好于 Macro-R。

3.4.1.3　子模型性能比较

在本节中，本课题将 SPAM 和 DPM 和 DPAM 模型进行对比，以便看到两个子模型的差异。图 3–4展示了三个模型的结果比较。

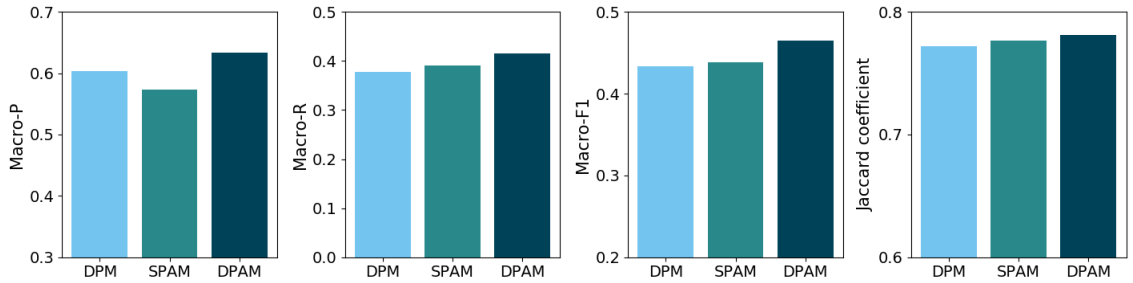

图 3–4: Performance comparison of the final DPAM model with its two sub-variant models DPM and SPAM on Fraud dataset in terms of Marco-P, Macro-R, Macro-F1, and Jaccard.

从实验结果中，可以得到一个有趣的发现，SPAM 在 Macro-R 上比 DPM 获得了更好的结果，DPM 在 Macro-P 上比 SPAM 的结果更好。它意味着 SPAM 通过注意力机制可以缓解标签不均衡问题，DPM 通过调整不同标签的阈值可以更准确地预测标签。最后，通过多任务方式联合学习两个子模型，DPAM 在所有指标上获得了最好的结果。

3.4.1.4 Comaprison against baselines

本课题进一步将 DPAM 模型与基线方法进行对比。在两个数据集上的实验结果如图 3–5。从实验结果中由以下发现：

- 1）毫不惊讶 POP 在所有模型中的结果最差，这体现了判决预测是一项不容易的工作。这是由于司法领域标签集合的分布是不规则的，因此对不同的样本预测相同的标签是不恰当的。
- 2）一阶方法（BSVM,ML-KNN）的结果好于 POP 方法。
- 3）二阶方法的结果要好于一阶方法，这证明了建模标签关联信息可以有效提升模型性能。以诈骗案由数据集为例，BP-MLL 方法相比 BSVM 方法，在 320 维特征下 Macro-F1 上性能提升了 24.4%。
- 4）TEXTCNN-MLL 结果好于 BP-MLL，它体现了通过深度神经网络模型可以比浅层模型得到更好的结果。这个结果与之前研究者的发现一致[13]。
- 5）CC 结果好于 BSVM，但是性能提升有限。这个原因是因为，作为链式方法，CC 容易受到误差传播的影响[63]。例如，当一个分类器误分类了一个像本，错误标签信息将会被传递到下一个分类器。
- 6）最后，当利用多任务方式同时学习阈值预测器和多标签分类，DPAM 在所有指标上获得了最好的结果。例如，与第二名的方法（TEXTCNN-MLL）进行对比，在 Macro-P，Macro-R，Macro-F1 和 Jaccard 的结果分别提升了 2.5%, 4.3%, 3.5% and 2.0%，结果提升明显。

3.5 分析与讨论

3.5.1 训练策略的影响

为了学习提出的 DPA 模型，本课题利用 burn-in 方法进行优化。该方法中参数 n_{burn} 需要进行设置。本课题对 n_{burn} 的影响进行挖掘研究。

具体地，本课题在诈骗案由数据集上尝试了 $n_{burn} \in \{0, 200, 400, 600, 800, 1000, 1200\}$。图 3–6展示了在特征维度为 320 时不同 burn-in 值情况下 Macro-F1 的结果。

从实验结果中，可以得到以下结论：

- 随着 n_{burn} 不断增大，Macro-F1 的结果同时提升。
- 当 n_{burn} 的值持续增大，两次不同值下的性能差异越来越小。例如，当 n_{burn} 从

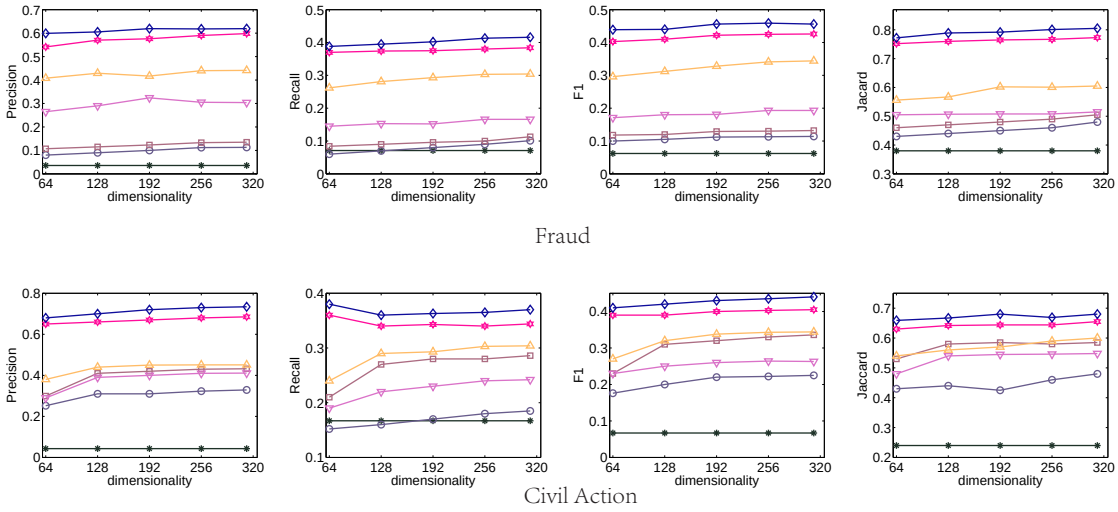

图 3–5: Performance comparison of DPAM among POP, BSVM, ML-KNN, BP-MLL, CC, and TextCNN-MLL over Fraud dataset. The dimensionality is increased from 64 to 320.

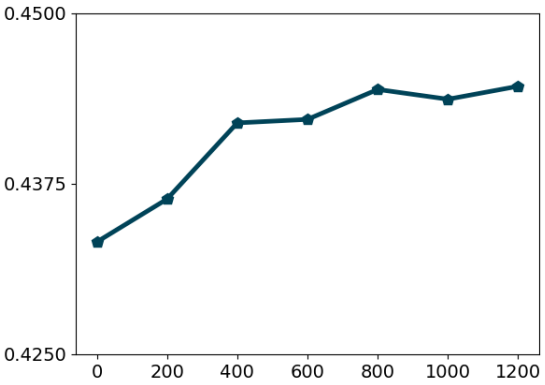

图 3–6: Performance variation in terms of Macro-F1 against the number of burn-in on two datasets. The number of burn-in is increased from 0 to 1200.

800 增加到 1000 时，Macro-F1 上的相对性能提升由 0.3%。它表示在 800 词迭代之后，已经获得了相对稳定的词特征表达，如果继续增加迭代次数，继续提升性能变得更困难。因此，在本模型的对比实验中，将 n_{burn} 设置为 1000。

3.5.2　案例分析

为了更好地理解为什么 DPAM 可以比其他模型获得更好的性能，在本章节中，本课题进行案例分析，比较 DPAM 和第二名的模型 TEXT-CNN 的性能。以诈骗案由数据集为例，首先根据出现次数将 70 个标签进行排序，之后将数据按标签分成 7 组，

每组包含 10 个标签。通过这种方式，第一组数据包含了出现次数最多的 10 个标签，最后一组包含了最稀疏的 10 个标签。

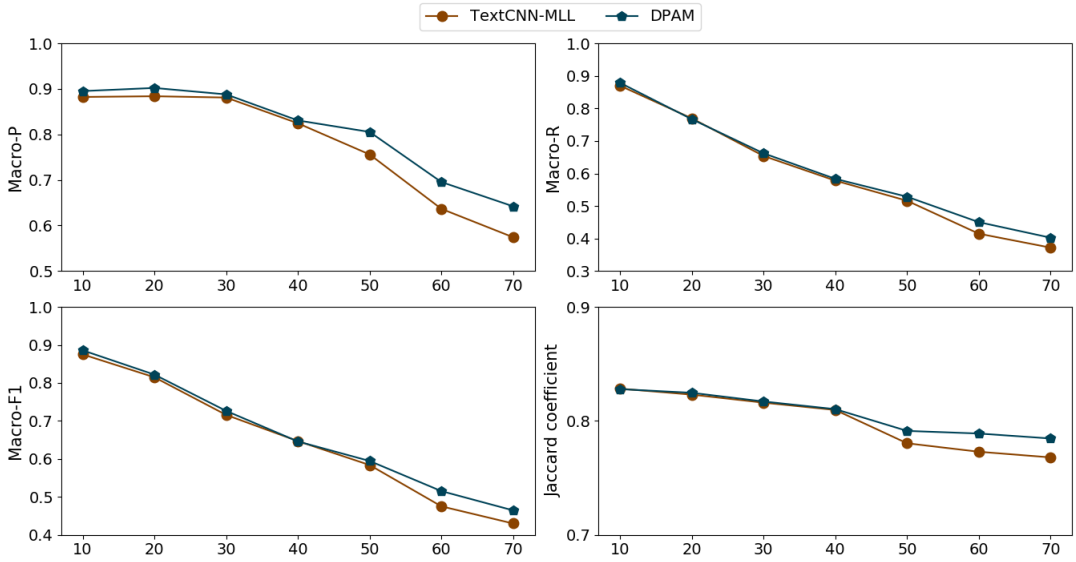

图 3–7: Performance comparison among different label group size. The x-axis represents the label size modeled, y-axis represents the performance in terms of different evaluations metrics.

给定这个条件，本课题比较了在第一组数据上的模型性能，进行 6 次重复实验，每次添加下一组数据进行比较。通过这种方式，本课题希望测试当面对标签不均衡问题时，DPAM 是否有效。结果展示如图 3–7，可以有以下发现：

- DPAM 和 TEXTCNN-MLL 的结果在考虑更多标签时下降，这与期望的结果一致，当考虑更稀疏的标签，会导致模型结果的降低。
- DPAM 在添加到第 5 组数据时，指标全面超越 TEXTCNN-MLL。一个有趣的现象是，当之后添加更多组数据时，性能差距越来越大。这意味着 DPAM 通过引入注意力机制矩阵可以缓解标签不均衡问题。

3.6 本章小结

在本章节中，本课题将司法判决预测问题看作多标签分类问题，提出一种动态成对注意力机制的模型（简称 DPAM）用来进行法条预测。通过引入从标签定义中学到

的注意力机制矩阵，本模型可以缓解标签不均衡问题。本课题之后提出一个动态阈值预测器用来自动学习一个鲁棒的阈值。最后本课题采用多任务框架同时学习多标签分类和阈值预测器，通过利用不同任务之间的信息传递，可以有效提升模型的泛化性能。通过在两个真实数据上的实验验证，证明本课题的方法比之前的方法在不同的评价指标下有明显的提升。

表 4–1: 法条定义中相关法条

刑法第一百九十七条: 使用伪造、变造的国库券或者国家发行的其他有价证券，进行**诈骗**活动，数额较大的，处五年以下有期徒刑或者拘役...
刑法第一百九十一条: 明知是毒品犯罪、黑社会性质的组织犯罪、恐怖活动犯罪、走私犯罪、贪污贿赂犯罪、破坏金融管理秩序犯罪、**金融诈骗**犯罪的所得及其产生的收益，为掩饰、隐瞒其来源和性质，有下列行为之一的，没收实施以上犯罪的所得及其产生的收益...

第四章　基于递归注意力机制的模型

本章主要提出了一种基于递归注意力机制的模型。首先章节 4.1描述了算法的研究背景；接着，在章节 4.2中介绍了基于递归注意力机制的模型结构，包括算法的设计、模型的学习与预测等过程；章节 4.3中介绍实验设置相关内容，对实验数据集，对比方法等进行介绍；章节 4.4 对实验结果进行分析，章节4.5对实验结果进行进一步的分析和讨论，章节4.6对本章内容进行总结。

4.1　引言

法条语义信息（例如，法条的定义）为法官进行正确的决策提供了有效的属性信息。表 **??**展示了两个相似的法条。具体来讲，给定案情描述，对法官来说，一个正常的步骤是法官首先浏览案情，之后浏览所有的法条，挑选与给定案情相关的候选法条（例如，刑法第 263 条和刑法第 264 条都与盗窃罪相关，它们在法条定义中有相似的文本描述）。之后通过对案情和候选法条进行详细的语义分析最终挑选正确的法条。这个过程往往会重复若干次之后才能得到最终的判决结果。

之前的工作，通常忽略了标签语义信息进行预测。此外，案情与法条语义之间的重复迭代信息被忽略，因此之前的算法性能是十分有限的。在本章中，为了解决这些

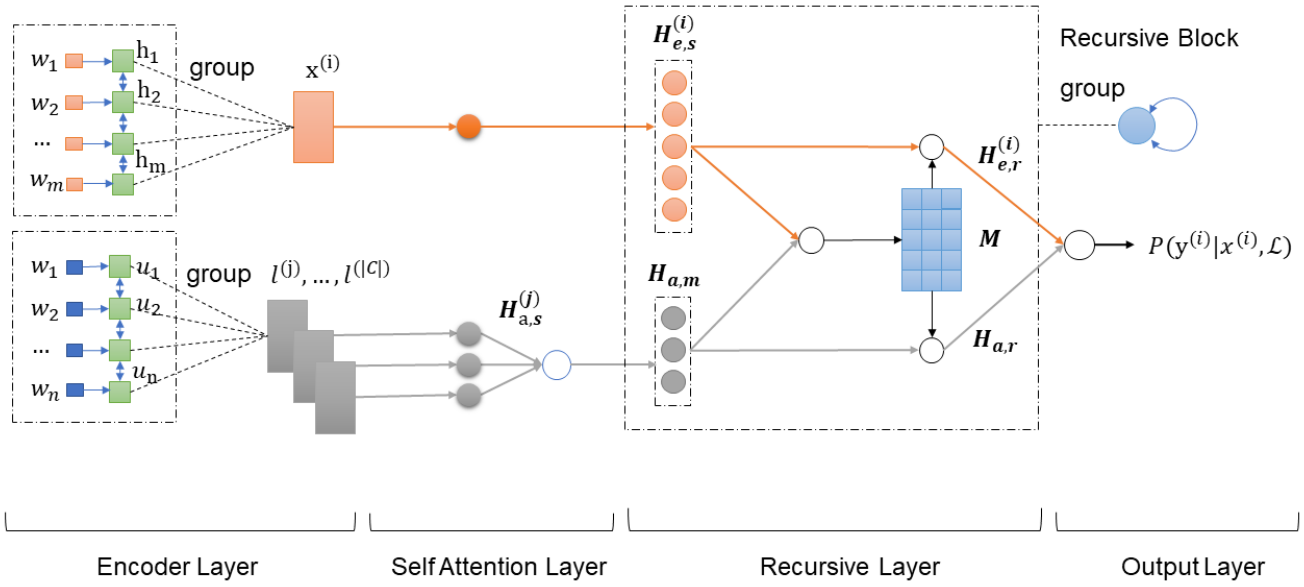

图 4–1: The overall architecture of the proposed Recurrent Attention Model (RAN).

问题，本课题提出了一种递归注意力网络（简称 RNN）。具体来讲，RAN 利用 LSTM 将案情描述和法条定义映射到一个低维空间。自注意力机制用来获取案情和法条自身的内部信息。协同注意力机制用来选择有效的语义信息对案情和法条进行正确的匹配。最后一个递归单元用来建模案情和法条之间的交互信息。

4.2 RAN 算法设计

在本节中，针对提出了 RAN 模型进行详细的介绍，其模型结果如图 4–1，之后对算法的学习和预测过程进行介绍。

4.2.1 编码层

让 $x^{(i)} = \{w_1, w_2, w_3, \ldots, w_m\}$ 表示有 m 个词的案情描述，$l^{(j)} = \{w_1, w_2, w_3, \ldots, w_n\}$ 表示有 n 个词的法条定义，w_i 代表第 i 个词的词袋表示，让 $\mathbf{V}^I = \{\vec{v}_t^I \in \mathbb{R}^{D_v} | t = 1, \ldots, N\}$ 表示所有词向量的连续空间。

针对每个案情描述和法条定义，本课题将词向量聚合作为案情表示和标签表示，一个双向 LSTM(简称 Bi-LSTM) 用来计算每个词在时刻 t 的隐层状态，其计算过程如

公式 4–1：

$$\begin{aligned}\overrightarrow{h_t} &= \overrightarrow{\text{LSTM}}(\overrightarrow{h}_{t-1}, w_t)\\ \overleftarrow{h_t} &= \overleftarrow{\text{LSTM}}(\overleftarrow{h}_{t-1}, w_t)\end{aligned} \tag{4–1}$$

利用 Bi-LSTM，本模型通过拼接两个方向的 LSTM 的隐层状态组成第 i 个词的隐层表示，$\vec{v}_t = [\overrightarrow{h_t}; \overleftarrow{h_t}]$，之后案情序列 $x^{(i)}$ 和法条序列 $l^{(j)}$ 被分别映射到连续的空间 $\mathbf{H}_e^{(i)} = [\vec{v}_e^i(1), \vec{v}_e^i(2), \ldots, \vec{v}_e^i(m)]$，$\mathbf{H}_a^{(i)} = [\vec{v}_a^i(1), \vec{v}_a^i(2), \ldots, \vec{v}_a^i(n)]$。

4.2.2　自注意力层

在案情描述文档中，不同的词拥有不同的重要程度用于支持法官判决，法官需要详细的阅读案情描述来确认其中的重要犯罪情节。相似的，法条定义中不同的词也有不同的重要程度。在刑法中，有许多易混淆的罪名，比如盗窃罪和抢劫罪，故意杀人和过世致人死亡，它们在法条定义中只有一些细微的差别。例如，是否在非法占有过程中使用暴力，犯罪者在非法占有中使用暴力，是否是由犯罪者故意造成受害人死亡。在进行判决的过程中，法官需要根据法条定义中的差异信息来确认被告人违反了哪些法条。

受自注意力机制[34] 的启发，给定案情描述或者法条定义，本课题根据词的重要程度针对文本中的不同词分配不同的权重。这种机制将查询和一系列键值对映射到输出，查询，键，值和输出拥有同样的维度。这本模型中，查询，键，值都是编码层的输出 $\mathbf{H}_e^{(i)}$ and $\mathbf{H}_a^{(j)}$，多头注意力机制允许模型对来自不同子空间的信息在不同位置进行加权。权重计算过程如下：

$$head_i = Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{4–2}$$

$$\begin{aligned}&MultiHead(Q, K, V) =\\ &Concat(head_1, \ldots, head_h)W^O\end{aligned} \tag{4–3}$$

其中，Q, K, V 代表打包的查询，键值，d_k 和 d_v 代表查询和值的维度，投影 $W_i^Q \in \mathbb{R}^{d_{model} \times d_k}, W_i^K \in \mathbb{R}^{d_{model} \times d_k}, W_i^V \in \mathbb{R}^{d_{model} \times d_v}, W^O \in \mathbb{R}^{hd_v \times d_{model}}$ 是参数矩阵。利用自注意力机制，本课题将 $\mathbf{H}_e^i$ 和 $\mathbf{H}_a^j$ 映射到另外一个相当大小的序列空间 $\mathbf{H}_{e,s}^{(i)} = [\vec{v}_{e,s}^{(i)}(1), \vec{v}_{e,s}^{(i)}(2), \ldots, \vec{v}_{a,s}^{(t)}(t)]$. and $\mathbf{H}_{a,s}^{(j)} = [\vec{v}_{a,s}^{(j)}(1), \vec{v}_{a,s}^{(j)}(2), \ldots, \vec{v}_{a,s}^{(j)}(t)]$。

4.2.3 递归层

在司法领域，法官需要仔细阅读案情描述以便获得重要信息，以便找出相关法条作为候选，之后针对案情描述和相关法条进行详细的分析，得到最终的判决结果，这个过程通过需要重复多次来进行最后的决策。与 Cui 等人[64] 提出的不同，不是利用简单的源文本到目标文本之间的交互信息，本课题设计了一种递归注意力机制单元用于建模法官的重复阅读行为。

通过自注意力机制，本课题获得了案情和标签的词级别的表达，分别是 $\mathbf{H}_{e,s}^{(i)}$ 和 $\mathbf{H}_{a,s}^{(j)}$。本课题通过聚合操作获取标签句子级别的表达，其计算过程如下：

$$\begin{aligned} \vec{v}_{a,m}(j) &= \frac{1}{m}\sum_{t}\vec{v}_{a,m}^{(j)}(t) \\ \mathbf{H}_{a,m} &= [\vec{v}_{a,m}(1), \vec{v}_{a,m}(2), \ldots, \vec{v}_{a,m}(|C|)] \end{aligned} \tag{4–4}$$

其中，$\mathbf{H}_{a,m}$ 代表标签全局表达，由每个标签的表达聚合而来。

本课题以 $\vec{v}_{a,m}(j)$ 作为 $\mathbf{H}_{a,m}$ 的第 j 个元素，以 $\vec{v}_{e,s}^{(i)}(k)$ 作为 $\mathbf{H}_{e,s}^{(i)}$ 的第 k 个元素，之后本课题对标签表达和案情词级别的表达构造匹配得分矩阵 $\mathbf{M}^{(i)}$，计算方法如下：

$$\mathbf{M}^{(i)}(j,k) = \vec{v}_{a,m}(j) \cdot \vec{v}_{e,s}^{(i)}(k) \tag{4–5}$$

其中，$M^{(i)} \in \mathbb{R}^{|\mathbf{H}_{a,m}|*|\mathbf{H}_{e,s}^{(i)}|}$，每个元素代表交互得分。

获得匹配得分矩阵 $\mathbf{M}^{(i)}$ 之后，采用列 softmax 函数获得每列的概率分布。其中每列代表独立的注意力值，之后采用 $\alpha(t) \in \mathbb{R}^{|C|}$ 代表每个案情中的词在 t 时刻标签级别的注意力，它可以看作是案情词到标签的权重分布，其计算过程如下：

$$\begin{aligned} \alpha(t) &= softmax(\mathbf{M}^{(i)}(1,t), \mathbf{M}^{(i)}(2,t), \ldots, \mathbf{M}^{(i)}(|\mathbf{H}_{a,m}|,t) \\ \alpha &= [\alpha(1), \alpha(2), \ldots, \alpha(|\mathbf{H}_{e,s}^{(i)}|)] \end{aligned} \tag{4–6}$$

之后对所有的 $\alpha(t)$ 进行平均得到平均标签级别的注意力 $\alpha' \in \mathbb{R}^{|H_{a,m}|}$，其中平均操作不会破坏正则化分布：

$$\alpha' = \frac{1}{|\mathbf{H}_{a,m}|}\sum_{i=1}^{|\mathbf{H}_{a,m}|}\alpha(t) \tag{4–7}$$

通过同样的方式，本课题可以使用行 softmax 获取标签到案情中每个词的注意力值 $\beta' \in \mathbb{R}^{|\mathbf{H}_{e,s}^{(i)}|}$，其计算过程如下：

$$\begin{aligned} \beta(t) &= softmax(\mathbf{M}(t,1), \mathbf{M}(t,2), \ldots, \mathbf{M}(t,|\mathbf{H}_{e,s}^{(i)}|) \\ \beta &= [\beta(1), \beta(2), \ldots, \beta(|\mathbf{H}_{a,m}|)] \end{aligned} \tag{4–8}$$

$$\beta' = \frac{1}{|\mathbf{H}_{e,s}^{(i)}|} \sum_{i=1}^{|\mathbf{H}_{e,s}^{(i)}|} \beta(t) \tag{4–9}$$

目前为止，本课题已经获得了双向的注意力矩阵 α' 和 β'。本课题的研究动机是模拟法官交替阅读案情和法条定义的过程。因此提出了一种递归的结构。直观上来讲，这个操作通过不断进行循环可以学习到重要的交互语义信息。其计算过程如下：

$$\begin{aligned} \mathbf{H}_{e,r}^{(i)} &= r(\mathbf{H}_{e,s}^{i} + \mathbf{H}_{e,s}^{i}\mathbf{W}^{\alpha'}\alpha') \\ \mathbf{H}_{a,r} &= r(\mathbf{H}_{a,m} + \mathbf{H}_{a,m}\mathbf{W}^{\beta'}\beta') \end{aligned} \tag{4–10}$$

其中 $\mathbf{W}^{\alpha'}$ 和 $\mathbf{W}^{\beta'}$ 是维度变换矩阵，r 代表递归操作。如公式 4–10中的描述，本模型重复这个过程若干次，最终通过案情和标签描述之间语义的重复交互获得最终正确的匹配。

4.2.4　输出层

为了整合案情和相关法条定义两部分信息，在输出层中，对于给定实例，本课题同时采用案情和标签两部分特征进行最终结果预测。在所有标签上进行得分预测的过程如下：

$$\begin{aligned} \vec{v}_{e,r}^{(i)} &= g(H_{e,r}^{(i)}) = \frac{1}{m}\sum_{t}\vec{v}_{e,r}^{(i)}(t) \\ \vec{v}_{a,r} &= g(H_{a,r}) = \frac{1}{m}\sum_{t}\vec{v}_{a,r}(t) \\ \vec{v}_{f}^{(i)} &= \vec{v}_{e,r}^{(i)} \oplus \vec{v}_{a,r} \\ \vec{v}_{o}^{(i)} &= \mathbf{W}^{o}\vec{v}_{o}^{(i)} + b^{o} \end{aligned} \tag{4–11}$$

其中，g 代表平均操作，$\vec{v}_{e,r}^{(i)}$ 代表案情文本特征表达，$\vec{v}_{a,r}$ 代表所有法条的平均特征表达，它代表法条标签的全局特征。$\oplus$ 代表拼接操作，$\mathbf{W}^{o}$ 和 $\mathbf{b}^{o}$ 是可学习的参数。

4.2.5 模型学习与预测

为了学习 **RAN** 的模型参数，采用随机梯度下降进行参数更新，在训练过程中采用二元交叉熵作为损失，其计算结果如下：

$$\begin{aligned} l = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{|C|}[Y^{(i)}(j)log(\sigma(\vec{v}_o^{(i)}(j))) \\ + (1-Y^{(i)}(j))log(1-\sigma(\vec{v}_o^{(i)}(j)))] \end{aligned} \tag{4–12}$$

其中 σ 是 sigmoid 函数 $\sigma(x) = \frac{1}{1+e^{-x}}$，$Y^{(i)}(j) \in \{0,1\}$ 是针对实例 $x^{(i)}$ 的第 j 个标签的真实值。

使用学习到的参数，分类策略如算法 4–2所示：

Algorithm 4–2 Framework of joint learning for our model

Initialize model $\Theta = \{\mathbf{W}^{Dv\times|C|}, \mathbf{V}\}$ randomly
iter = 0
repeat
    $iter \leftarrow iter + 1$
    for $i = 1, ..., |X|$ **do**
        for instance $x^{(i)}$
        encode evidence and article $H_e^{(i)}$, H_a
        set $n_{layer} = 3$
        repeat
            $n_{layer} \leftarrow n_{layer} - 1$
            compute the interaction information
        until $n_{layer} < 0$
        compute the gradient $\nabla(\theta)$ of Equation(4–12)
        update model $\theta \leftarrow \theta + \epsilon\nabla(\theta)$
    end for
until (Coverage or $t > num$)
return $\{\mathbf{W}^{Dv\times|C|}, \mathbf{V}^I\}$;

针对每个实例 $x^{(i)}$，可以获得每个标签 $Y^{(i)}(j)$ 的概率分布。结合阈值 t，可以得

到 $x^{(i)}$ 的预测标签集合，其计算过程如下：

$$Y^{(i)}(j) = \begin{cases} 1, & \textbf{if} \quad \mathbb{I}(\sigma(v_o^{(i)}(j))) > t \\ 0, & \textbf{else} \end{cases}$$

其中，$\mathbb{I}$ 是索引函数，概率大于 t 的标签作为与实例 $x^{(i)}$ 相关的标签。

4.3　实验设置

4.3.1　数据集介绍

本模型在三个数据集上进行了实验验证：

- **CJO**：这个数据集包含 114576 个样本，是本课题从中国裁判文书网[1]进行收集的数据，并且每个样本已经被结构化为案情，判决结果等及部分。
- **CAIL samll**：这个数据集是为司法判决竞赛开发的数据集，它由最高人民法院公布 [2]，包含 2043231 个样本。这个数据集包含事实描述，以及对应的相关法条，判决结果，罚金等。
- **CAIL2018**:这个数据集是一个公开的大型数据集[65] 用于司法判决预测,CAIL2018 包含超过 200 万个样本，是现有的同类数据集的若干倍大。

三个数据集的统计分析如表 4–2所示。最后本课题将所有数据以 8 : 2 划分为不重叠的两部分，训练集和测试机。

表 4–2: Basic statistics of the three datasets for experiments.

datasets	number of samples	relevant articles	average fact description length	average article size
CJO	114,576	137	825	1.17
CAIL small	204,231	183	263	1.27
CAIL2018	1,710,856	183	279	1.04

4.3.2　模型评估

本课题采用以下模型进行实验对比：

[1]https://wenshu.court.gov.cn/

[2]http://cail.cipsc.org.cn/

- **ML-KNN**，**BR**，**CC** 均在章节 3.3中进行过详细介绍，此处不再赘述。
- **CNN**：一种二阶多标签分类模型，使用卷积神经网络作为输入特征[13]，之后输入到带有 sigmoid 的全连接模型中在所有标签下得到概率分布。采用多标签软边界损失进行优化。
- **BiLSTM**：BiLSTM[66] 同样是一种二阶方法，该方法是一种常用的用于建模长期依赖的方法，本课题采用于 **CNN** 类似的方法进行继承。
- **DPAM**：DPAM[DPAM] 本文在 三提出的另一种基于神经网络的模型，采用注意力机制获取标签之间的关联关系。

针对所有模型，本课题设置句子最大长度为 500 个词。对于浅层模型来说，这些模型采用词袋和 TF-IDF 特征当作输入，使用卡方检验选取前 5000 个特征[17]。对于其他模型，本课题将案情表达和法条表达设置为 256 维特征。本课题采用一个 100000 大小的词典，词典外的词采用一个特殊的词 *unk* 进行替换。使用 Adam 作为优化器进行损失优化，设置学习率为 0.001，两个动量参数分别设置为 0.9 和 0.999。具体地，

4.4 实验结果

4.4.1 Comparison against Baselines

本课题将提出的 **SAN** 模型与 baseline 进行了对比，在三个数据集上的性能结果如表 **??**所示，MP，MR，MF 和 JS 分别代表 Macro-P，Macro-R，Macro-F1 和杰卡德相关系数（所有数值的百分比符号均被省略）。每行最好的实验结果用下划线进行了标注，最后一列代表与最好的基线模型相关性能提升的多少。本课题将对比的模型分为三个大类别，浅层模型，基于神经网络的模型以及基于注意力机制的模型。从结果中，可以得到以下结论：

- 毫不惊讶，**MLKNN** 和 **BR** 的在所有指标上结果最差，因为这两种方法将多标签分类转换为单标签分类。
- **CC** 方法比 **MLKNN** 和 **BR** 要稍好，这是由于 **CC** 建模了标签关联。
- 基于神经网络的方法性能明显好于大多数浅层模型。以 **TextCNN** 为例，与最好的浅层模型（CC）相比，在 CAIL small 上 MP，MR，MF, JS 分别提升了 36.2%，22.24%，25.77% 和 11.53%。它证明神经网络模型可以有效建模语义信息。
- 基于注意力机制的模型 **DPAM** 的结果好于大多数模型（除过 **RAN**），这表明引入法条语义信息是一种很好的整合案情和法条信息的方式。

表 4–3: Comparison between our methods and all baselines on three datasets.

Dataset	metrics	Shallow Model				Neural Network Based Model		Attention Based Model	Our Mode
		MLKNN	BR	CC	SVM	TextCNN	BiLSTM	DPAM	RAN
CJO	MP	59.49	74.28	72.33	67.68	78.53	78.81	79.39	**81.34**
	MR	32.14	50.84	53.22	51.37	54.16	54.96	55.60	**55.59**
	MF	38.85	57.41	58.60	55.77	61.40	62.17	62.79	**63.28**
	JS	53.25	79.40	82.02	83.55	80.25	80.40	80.76	**80.81**
CAIL small	MP	31.75	41.59	42.12	43.07	78.32	79.93	80.35	**81.23**
	MR	20.11	30.23	32.49	39.66	54.73	57.77	62.03	**64.90**
	MF	22.93	33.57	35.58	40.14	61.35	63.98	67.42	**69.49**
	JS	38.85	59.74	62.59	71.98	74.12	75.09	76.00	**77.42**
CAIL2018	MP	28.88	40.42	38.91	40.82	74.86	75.05	75.26	**77.69**
	MR	16.59	26.95	28.86	31.53	57.43	57.79	58.04	**58.69**
	MF	19.68	30.65	31.59	34.01	62.91	63.01	63.33	**64.88**
	JS	70.28	88.34	90.57	90.92	94.61	94.61	94.39	**94.68**

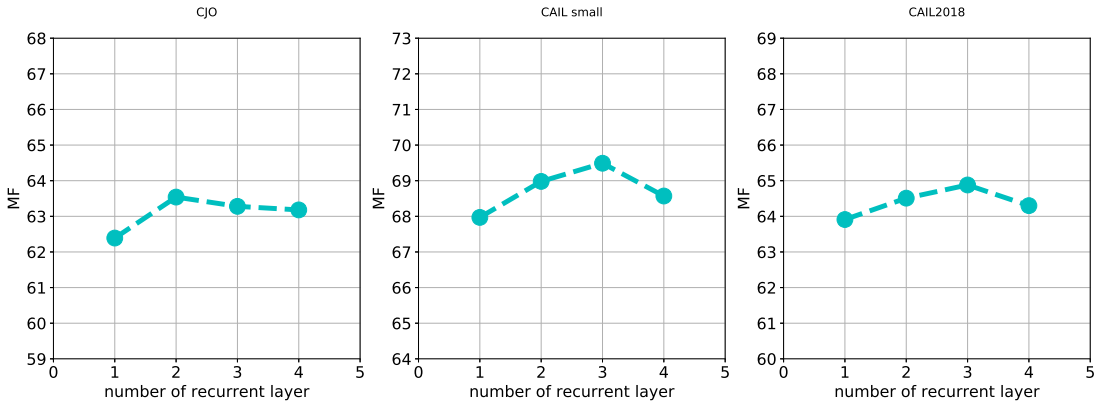

图 4–2: The performance of different number of recurrent layers on three datasets.

- **RAN** 在所有指标上获得了最好的性能。在 CAIL Small 上与 **DPAM** 相比，MP，MR，MF，JS 的结果提升分别是 0.88%, 2.5%, 2.07%, 1.42%。

这个结果能有效支撑本模型的假设，建模案情和法条之间重复的交互语义信息是非常重要和有效的。

4.5 分析与讨论

本课题在模型上进行了进一步的分析。在本节中，本课题首先挖掘了递归层数对结果的影响，之后利用消融实验探索了不同模型层的有效性。

4.5.1 The impact of recurrent layer

本课题在递归交互层数 $n \in \{1, 2, 3, 4\}$ 的情况下在三个数据集上进行实验，实验结果如图 4–2所示，从结果可以得到以下结论：

- 随着递归层数的增多，结果有所提升。
- 当层数增加到一定程度，性能提升变得非常有限，并且需要更多的计算资源。
- 当递归层数增加到 3 层以上，结果开始下降。

4.5.2　Ablation test

为了进一步验证，本课题进行了不同程度的消融实验，将递归层移去（简称 **R-r**），将自注意力层移去（简称 R-s）分别进行实验。实验结果如表 **??**所示：本课题得到以

表 4–4: The ablation experiment on CAIL small.

Dataset	method	MP	MR	MF	JS
CJO	**DPAM**	79.39	55.60	62.79	80.76
	R-s	80.36	55.02	62.85	80.77
	R-r	78.78	54.56	61.97	80.35
	RAN	81.34	55.59	63.28	80.81
CAIL small	**DPAM**	80.35	62.03	67.42	76.00
	R-s	80.2	64.94	68.98	76.58
	R-r	79.37	62.08	66.75	76.93
	RAN	81.23	64.9	69.49	77.42
CAIL2018	**DPAM**	75.26	58.04	63.33	94.39
	R-s	75.3	58.06	63.9	94.64
	R-r	74.33	57.54	62.65	94.41
	RAN	77.69	58.69	64.88	94.68

下结论：

- 仅仅保持自注意力层，实验结果比 **DPAM** 要差。这是由于缺乏标签相关信息无法有效区分易混淆的法条标签。
- 仅仅保存递归层，实验结果稍好于 **DPAM**。这是由于递归层可以有效利用案情和法条标签之间的交互信息。
- 结合自注意力层和递归层，**RAN** 通过自注意力层获得案情和法条独立的重要信息，之后通过递归层获得交互信息，可以获得更有效的信息帮助提高模型性能。

这进一步证明了递归层可以有效建模获取重复的语义信息用于提升判决预测的性能。

4.6 本章小结

在本章节张，本课题提出了一种递归注意力网络，可以模拟法官重复交替阅读法案和法条的过程，这种方法可以有效利用案情描述和法条定义之间的交互语义特征，扩展实验证明提出的模型性能远好于其他模型。进一步的分析证明本模型不仅能够获取标签关联信息，还能通过递归单元获取多层次的注意力信息。

第五章　功能测试

参考文献

[1] Liu C, Chang T. Some Case-Refinement Strategies for Case-Based Criminal Summary Judgments[C]// Foundations of Intelligent Systems, 14th International Symposium, ISMIS 2003, Maebashi City, Japan, October 28-31, 2003, Proceedings. [S.l. : s.n.], 2003: 285-291.

[2] Kort F. Predicting Supreme Court decisions mathematically: A quantitative analysis of the “right to counsel” cases[J]. American Political Science Review, 1957, 51(1): 1-12.

[3] Nagel S S. Applying correlation analysis to case prediction[J]. Tex. L. Rev., 1963, 42: 1006.

[4] Ulmer S S. Quantitative analysis of judicial processes: Some practical and theoretical applications[J]. Law and Contemporary Problems, 1963, 28(1): 164-184.

[5] Keown R. Mathematical Models For Legal Prediction[J]. The John Marshall Journal of Information Technology and Privacy Law, 1980, 2(1): 29.

[6] Kim M, Xu Y, Goebel R. Legal Question Answering Using Ranking SVM and Syntactic/Semantic Similarity[J]., 2014: 244-258.

[7] Aletras N, Tsarapatsanis D, Preotiuc-Pietro D, et al. Predicting judicial decisions of the European Court of Human Rights: a Natural Language Processing perspective[J]. PeerJ Computer Science, 2016, 2: e93.

[8] Liu Y, Chen Y, Ho W. Predicting associated statutes for legal problems[J]. Information Processing and Management, 2015, 51(1): 194-211.

[9] Liu C, Chang C, Ho J. Case Instance Generation and Refinement for Case-Based Criminal Summary Judgments in Chinese[J]. J. Inf. Sci. Eng., 2004, 20(4): 783-800.

[10] Katz D M, II M J B, Blackman J. Predicting the Behavior of the Supreme Court of the United States: A General Approach[J]. CoRR, 2014, abs/1407.6333arXiv: 1407.6333.

[11] Sulea O M, Zampieri M, Malmasi S, et al. Exploring the Use of Text Classification in the Legal Domain[J]., 2017, 2143.

[12] Carvalho D S, Nguyen M, Tran C, et al. Lexical-Morphological Modeling for Legal Text Analysis[J]. International symposium on artificial intelligence, 2015: 295-311.

[13] Kim Y. Convolutional Neural Networks for Sentence Classification[C]// Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL. [S.l. : s.n.], 2014: 1746-1751.

[14] Bordes A, Glorot X, Weston J, et al. Joint Learning of Words and Meaning Representations for Open-Text Semantic Parsing[C]// Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2012, La Palma, Canary Islands, Spain, April 21-23, 2012. [S.l. : s.n.], 2012: 127-135.

[15] Luong T, Sutskever I, Le Q V, et al. Addressing the Rare Word Problem in Neural Machine Translation[C]// Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers. [S.l. : s.n.], 2015: 11-19.

[16] Zhong H, Guo Z, Tu C, et al. Legal Judgment Prediction via Topological Learning[C]// Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018. [S.l. : s.n.], 2018: 3540-3549.

[17] Luo B, Feng Y, Xu J, et al. Learning to Predict Charges for Criminal Cases with Legal Basis.[J]. Empirical methods in natural language processing, 2017: 2727-2736.

[18] Long S, Tu C, Liu Z, et al. Automatic Judgment Prediction via Legal Reading Comprehension[J]. CoRR, 2018, abs/1809.06537.

[19] Hu Z, Li X, Tu C, et al. Few-Shot Charge Prediction with Discriminative Legal Attributes[C/OL]// Proceedings of the 27th International Conference on Computational

Linguistics. Santa Fe, New Mexico, USA: Association for Computational Linguistics, 2018: 487-498. http://aclweb.org/anthology/C18-1041.

[20] Zhang M L, Zhou Z H. A review on multi-label learning algorithms[J]. IEEE transactions on knowledge and data engineering, 2014, 26(8): 1819-1837.

[21] Boutell M R, Luo J, Shen X, et al. Learning multi-label scene classification[J]. Pattern recognition, 2004, 37(9): 1757-1771.

[22] Brinker K. Multilabel classification via calibrated label ranking[J]. Machine Learning, 2008, 73(2): 133-153.

[23] Tsoumakas G, Vlahavas I. Random k-Labelsets: An Ensemble Method for Multilabel Classification[C]// European Conference on Machine Learning. [S.l. : s.n.], 2007: 406-417.

[24] Li X, Guo Y. Multi-label classification with feature-aware non-linear label space transformation[C]// International Conference on Artificial Intelligence. [S.l. : s.n.], 2015: 3635-3642.

[25] Yang Y. A Study of Thresholding Strategies for Text Categorization[C]// Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. New Orleans, Louisiana, USA: ACM, 2001: 137-145.

[26] Fan R E, Lin C J. A study on threshold selection for multi-label classification[J]. Department of Computer Science, National Taiwan University, 2007: 1-23.

[27] Elisseeff A, Weston J. A kernel method for multi-labelled classification[C]// International Conference on Neural Information Processing Systems: Natural and Synthetic. [S.l. : s.n.], 2001: 681-687.

[28] Mnih V, Heess N, Graves A, et al. Recurrent Models of Visual Attention[C]// Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada. [S.l. : s.n.], 2014: 2204-2212.

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

[30] Luong T, Pham H, Manning C D. Effective Approaches to Attention-based Neural Machine Translation[C]// Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015. [S.l. : s.n.], 2015: 1412-1421.

[31] Yin W, Schütze H, Xiang B, et al. ABCNN: Attention-Based Convolutional Neural Network for Modeling Sentence Pairs[J]. TACL, 2016, 4: 259-272.

[32] Lin Z, Feng M, dos Santos C N, et al. A Structured Self-attentive Sentence Embedding[J]. CoRR, 2017, abs/1703.03130arXiv: 1703.03130.

[33] Devlin J, Chang M, Lee K, et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding[J]. CoRR, 2018, abs/1810.04805arXiv: 1810.04805.

[34] Vaswani A, Shazeer N, Parmar N, et al. Attention is All you Need[C]// Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA. [S.l. : s.n.], 2017: 6000-6010.

[35] Torralba A, Murphy K P, Freeman W T. Sharing visual features for multiclass and multiview object detection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29(5): 854-869.

[36] Yim J, Jung H, Yoo B, et al. Rotating your face using multi-task deep neural network[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. [S.l. : s.n.], 2015: 676-684.

[37] Zhang T, Ghanem B, Liu S, et al. Robust Visual Tracking via Structured Multi-Task Sparse Learning[J]. International Journal of Computer Vision, 2013, 101(2): 367-383.

[38] Liu P, Qiu X, Huang X. Adversarial Multi-task Learning for Text Classification[C]// Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers. [S.l. : s.n.], 2017: 1-10.

[39] Glorot X, Bordes A, Bengio Y. Domain adaptation for large-scale sentiment classification: a deep learning approach[C]// International Conference on International Conference on Machine Learning. [S.l. : s.n.], 2011: 513-520.

[40] Collobert R, Weston J. A unified architecture for natural language processing: Deep neural networks with multitask learning[C]// Proceedings of the 25th international conference on Machine learning. [S.l. : s.n.], 2008: 160-167.

[41] Liu X, Gao J, He X, et al. Representation Learning Using Multi-Task Deep Neural Networks for Semantic Classification avvnd Information Retrieval[C]// Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. [S.l. : s.n.], 2015: 912-921.

[42] Luong M T, Le Q V, Sutskever I, et al. Multi-task sequence to sequence learning[J]. ArXiv preprint arXiv:1511.06114, 2015.

[43] Dong Y, Yang Y, Tang J, et al. Inferring user demographics and social strategies in mobile social networks[C]// Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. [S.l. : s.n.], 2014: 15-24.

[44] Zhong E, Tan B, Mo K, et al. User demographics prediction based on mobile data[J]. Pervasive and Mobile Computing, 2013, 9(6): 823-837.

[45] Argyriou A, Evgeniou T, Pontil M. Multi-task feature learning[C]// Advances in neural information processing systems. [S.l. : s.n.], 2007: 41-48.

[46] Kang Z, Grauman K, Sha F. Learning with Whom to Share in Multi-task Feature Learning[C]// International Conference on Machine Learning, ICML 2011, Bellevue, Washington, Usa, June 28 - July. [S.l. : s.n.], 2011: 521-528.

[47] Zhang Y, Yeung D, Xu Q. Probabilistic Multi-Task Feature Selection[J]. Advances in Neural Information Processing Systems, 2010: 2559-2567.

[48] Zhang H, Xiao L, Wang Y, et al. A Generalized Recurrent Neural Architecture for Text Classification with Multi-Task Learning[J]., 2017: 3385-3391.

[49] Sun Y, Wang X, Tang X. Deep Learning Face Representation by Joint Identification-Verification[J]. Advances in Neural Information Processing Systems, 2014, 27: 1988-1996.

[50] Wang Y, Wipf D, Ling Q, et al. Multi-task learning for subspace segmentation[C]// International Conference on International Conference on Machine Learning. [S.l. : s.n.], 2015: 1209-1217.

[51] Isonuma M, Fujino T, Mori J, et al. Extractive Summarization Using Multi-Task Learning with Document Classification[C]// Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017. [S.l. : s.n.], 2017: 2091-2100.

[52] Deerwester S C, Dumais S T, Landauer T K, et al. Indexing by Latent Semantic Analysis[J]. JASIS, 1990, 41(6): 391-407.

[53] Hofmann T. Probabilistic Latent Semantic Analysis[C]// UAI '99: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, Stockholm, Sweden, July 30 - August 1, 1999. [S.l. : s.n.], 1999: 289-296.

[54] Bengio Y, Ducharme R, Vincent P. A Neural Probabilistic Language Model[C]// Advances in Neural Information Processing Systems 13, Papers from Neural Information Processing Systems (NIPS) 2000, Denver, CO, USA. [S.l. : s.n.], 2000: 932-938.

[55] Mikolov T, Chen K, Corrado G, et al. Efficient Estimation of Word Representations in Vector Space[J]. CoRR, 2013, abs/1301.3781arXiv: 1301.3781.

[56] Grave E, Mikolov T, Joulin A, et al. Bag of Tricks for Efficient Text Classification[C]// Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 2: Short Papers. [S.l. : s.n.], 2017: 427-431.

[57] Mikolov T, Chen K, Corrado G, et al. Efficient Estimation of Word Representations in Vector Space[J]. ArXiv: Computation and Language, 2013.

[58] Zhang M L, Zhou Z H. ML-KNN: A lazy learning approach to multi-label learning[J]. Pattern recognition, 2007, 40(7): 2038-2048.

[59] Zhang M L, Zhou Z H. Multilabel Neural Networks with Applications to Functional Genomics and Text Categorization[J]. IEEE Transactions on Knowledge and Data Engineering, 2006, 18(10): 1338-1351.

[60] Read J, Pfahringer B, Holmes G, et al. Classifier chains for multi-label classification[J]. Machine Learning, 2011, 85(3): 333-359.

[61] Le Q V, Mikolov T. Distributed Representations of Sentences and Documents[J]. International conference on machine learning, 2014: 1188-1196.

[62] Clare A, King R D. Knowledge Discovery in Multi-label Phenotype Data[J]. European conference on principles of data mining and knowledge discovery, 2001: 42-53.

[63] Kubat M, Sarinnapakorn K, Dendamrongvit S. Induction in Multi-Label Text Classification Domains[G]// Advances in Machine Learning II, Dedicated to the Memory of Professor Ryszard S. Michalski. [S.l. : s.n.], 2010: 225-244.

[64] Cui Y, Chen Z, Wei S, et al. Attention-over-Attention Neural Networks for Reading Comprehension[C]// Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers. [S.l. : s.n.], 2017: 593-602.

[65] Zhong H, Zhipeng G, Tu C, et al. Legal Judgment Prediction via Topological Learning[C]// Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. [S.l. : s.n.], 2018: 3540-3549.

[66] Graves A, Schmidhuber J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures[J]. Neural Networks, 2005, 18(5-6): 602-610.

致谢

谢谢大家

攻读硕士期间发表的学术论文和专利

[1] **Zhang San**, Newton I, Hawking S W, et al. An extended brief history of time[J]., 2079, 1234(4): 567-890.

[2] McClane J, McClane L, Gennero H, et al. Transcript in Die hard[C]// Proc. HDDD 100th Super Technology Conference (STC 2046). Eta Cygni, Cygnus: [s.n.], 2046: 123-456.

[3] 张三, 李四. 一种进行时空旅行的装置: [P]. 中国. 2046-01-09.