

本科毕业论文（设计）

|  |  |
| --- | --- |
| **论文题目（中文）**  **论文题目（英文）** | CALEXText—基于规则的文本时间表达识别系统的设计与实现 |
| CALEXText - A Rule-Based System for Recognizing Temporal Expressions in Text |

学生姓名 赵浩然

指导教师 赵志立

学 院 信息科学与工程学院

专 业 计算机科学与技术

年 级 2019级

兰州大学教务处

**诚信责任书**

本人郑重声明：本人所呈交的毕业论文（设计），是在导师的指导下独立进行研究所取得的成果。毕业论文（设计）中凡引用他人已经发表或未发表的成果、数据、观点等，均已明确注明出处。除文中已经注明引用的内容外，不包含任何其他个人、集体已经发表或未发表的论文。

本声明的法律责任由本人承担。

论文作者签名： 日 期：

**关于毕业论文（设计）使用授权的声明**

本人在导师指导下所完成的论文及相关的职务作品，知识产权归属兰州大学。本人完全了解兰州大学有关保存、使用毕业论文（设计）的规定，同意学校保存或向国家有关部门或机构送交论文的纸质版和电子版，允许论文被查阅和借阅；本人授权兰州大学可以将本毕业论文（设计）的全部或部分内容编入有关数据库进行检索，可以采用任何复制手段保存和汇编本毕业论文（设计）。本人离校后发表、使用毕业论文（设计）或与该毕业论文（设计）直接相关的学术论文或成果时，第一署名单位仍然为兰州大学。

本毕业论文（设计）研究内容：

□可以公开

□不宜公开，已在学位办公室办理保密申请，解密后适用本授权书。

（请在以上选项内选择其中一项打“√”）

论文作者签名： 导师签名：

日 期： 日 期：

**CALEXText—基于规则的文本时间表达识别系统的设计与实现**

中文摘要

从文本中提取和模拟时间信息是构建时间线和轨迹的一个重要因素。然而，识别自然文本中时间相关的表达已经被证明是计算语言学中一个特别具有挑战性且活跃的领域。尽管很多人尝试使用机器学习的方法来解决该问题，但目前最先进的解决方案大多数仍然使用基于规则的系统来规范化时间性表达式，并根据规则为每个时间性表达提供一个确定的时间值。本文设计并实现了一种基于规则的文本时间表达式识别系统——CALEXText，主要工作有：

CALEXText通过识别英语文本中的时间表达，使用CALEX（Calendar Expressions）的注释准则来识别和规范英语文本中的时间表达。

CALEXText系统的设计遵循了一种基于规则的方法，该方法对文本中的时间表达进行分析和识别。CALEXText的规则是基于时间表达的语言特征和语法规则进行制定的，包括日期、时间、星期几等关键词的识别。同时，CALEX还考虑了一些常见的时间表达的变形，如缩写、合并等。识别算法采用了正则表达式匹配，并结合了对规则的分层迭代匹配，使得识别精度高。

实验结果表明，该系统在识别和规范英语文本中的时间表达方面具有较高的准确率和效率。CALEXText系统的识别率高达90%，其中绝大多数的识别错误是由于语料库中缺少相应的规则导致的。最后，本文还对CALEXText系统的未来发展进行了进一步的探讨。

**关键词：**时间表达式识别；文本时间关系解析；规则；正则表达式

**CALEXText - A Rule-Based System for Recognizing Temporal Expressions in Text**

# Abstract

Extracting and modeling temporal information from text is essential in constructing timelines and trajectories. However, recognizing time-related expressions in the natural text has proven to be a particularly challenging and active area of computational linguistics. Although many attempts have been made to address the problem using machine learning approaches, most current state-of-the-art solutions still use rule-based systems to normalize temporal expressions and provide a determined temporal value for each expression according to the rules. This paper aims to introduce a rule-based system for recognizing temporal expressions in the text - CALEXText. CALEXText identifies and normalizes temporal expressions in English text by recognizing them using the annotation criterion CALEX (Calendar Expressions). Time expressions in English texts using the CALEX (Calendar Expressions) annotation guidelines.

The design of the CALEXText system follows a rule-based approach that analyzes and identifies temporal expressions in texts. The rules of CALEXText are formulated based on the linguistic features and grammatical rules of temporal expressions, including identifying keywords such as date, time, and day of the week. At the same time, CALEX also considers some joint deformations of temporal expressions, such as abbreviation and merging. The recognition algorithm employs regular expression matching and combines hierarchical iterative matching of rules, resulting in high recognition accuracy.

The experimental results of the CALEXText system show that the system has high accuracy and efficiency in recognizing and normalizing temporal expressions in English texts. The recognition rate of the CALEXText system is as high as 90%, and most of the recognition errors are caused by the lack of corresponding rules in the corpus. Finally, the future development of the CALEXText system is further discussed in this paper.

**Keywords:** temporal expression recognition; text temporal relationship parsing; rules; regular expression

**目 录**

[中文摘要 I](#_Toc133539039)

[Abstract II](#_Toc133539040)

[第一章 绪论 1](#_Toc133539041)

[1.1 研究背景与目的 1](#_Toc133539042)

[1.2 国内外研究现状 1](#_Toc133539043)

[1.3 本文主要研究内容 3](#_Toc133539044)

[1.4 论文组织结构 3](#_Toc133539045)

[第二章 相关理论知识与技术 4](#_Toc133539046)

[2.1 自然文本时间信息标注问题 4](#_Toc133539047)

[2.1.1 问题定义 4](#_Toc133539048)

[2.1.2 问题关键点 4](#_Toc133539049)

[2.2 TimeML注释模式 5](#_Toc133539050)

[2.2.1 时间（TIME） 5](#_Toc133539051)

[2.2.2 日期（DATE） 5](#_Toc133539052)

[2.2.3 持续时间（DURATION） 6](#_Toc133539053)

[2.2.4 集合（SET） 6](#_Toc133539054)

[2.3 CALEX注释模式 6](#_Toc133539055)

[2.3.1 类型（TYPE） 7](#_Toc133539056)

[2.3.2 元数据（METADATA） 7](#_Toc133539057)

[2.3.3 值（VALUE） 7](#_Toc133539058)

[2.4 基于规则的方法 7](#_Toc133539059)

[2.5 医疗文本信息提取 8](#_Toc133539060)

[2.6 MTSample数据集 8](#_Toc133539061)

[2.7 评价指标 9](#_Toc133539062)

[2.8 本章小结 10](#_Toc133539063)

[第三章 CALEX注释模式有效性的研究 11](#_Toc133539064)

[3.1 数据收集与预处理 11](#_Toc133539065)

[3.1.1 数据来源 11](#_Toc133539066)

[3.1.2 数据描述 11](#_Toc133539067)

[3.1.3 数据预处理 11](#_Toc133539068)

[3.2 注释模式对比 12](#_Toc133539069)

[3.2.1 TIMEX 12](#_Toc133539070)

[3.2.2 CALEX 13](#_Toc133539071)

[3.3 实验结果 14](#_Toc133539072)

[3.4 本章小结 15](#_Toc133539073)

[第四章 基于规则的文本时间表达式识别系统设计与实现 16](#_Toc133539074)

[4.1 系统描述 16](#_Toc133539075)

[4.1.1 问题描述 16](#_Toc133539076)

[4.1.2 环境要求 16](#_Toc133539077)

[4.1.3 系统要求 16](#_Toc133539078)

[4.2 系统设计 17](#_Toc133539079)

[4.2.1 系统架构 17](#_Toc133539080)

[4.2.2 系统流程 18](#_Toc133539081)

[4.2.3 规则设计 19](#_Toc133539082)

[4.3 系统测试与评估 19](#_Toc133539083)

[4.3.1 数据来源 19](#_Toc133539084)

[4.3.2 实验设计 19](#_Toc133539085)

[4.3.3 实验结果 20](#_Toc133539086)

[4.4 本章小结 20](#_Toc133539087)

[第五章 总结与展望 21](#_Toc133539088)

[5.1 论文工作总结 21](#_Toc133539089)

[5.3 未来展望 21](#_Toc133539090)

[参考文献 23](#_Toc133539091)

[致 谢 25](#_Toc133539092)

**图 目 录**

[图 2.1 时间信息表达式 5](#_Toc133538790)

[图 2.2 日期信息表达式 6](#_Toc133538791)

[图 2.3 持续时间信息表达式 6](#_Toc133538792)

[图 2.4 集合信息表达式 6](#_Toc133538793)

[图 2.5 MTSamples报告 9](#_Toc133538794)

[图 3.1 MTSample数据预处理 12](#_Toc133538795)

[图 3.2 示例段落原文 12](#_Toc133538796)

[图 3.3 示例段落中的TIMEX元素 13](#_Toc133538797)

[图 3.4 TIMEX注释模式下的时间线 13](#_Toc133538798)

[图 3.5 示例段落中的CALXE元素 14](#_Toc133538799)

[图 3.6 CALEX注释模式下的时间线 14](#_Toc133538800)

[图 3.7 实验完整度结果 15](#_Toc133538801)

[图 4.1 CALEX系统UML图 17](#_Toc133538802)

[图 4.2 CALEXText系统流程图 18](#_Toc133538803)

[图 4.3 实验召回率结果 20](#_Toc133538804)

**表 目 录**

[表 4.1 系统要求 17](#_Toc133162926)

第一章 绪论

1.1 研究背景与目的

几十年来，企业和组织在不同结构（如结构化、半结构化和非结构化）和格式中创建了大量数据。因此，其中相当一部分信息被以物理或数字文件的形式存储在本地，或者存储在数据仓库、数据湖和网络存储中。以物理或数字形式存储的文件被保留以供参考或日后使用。从这些存储的信息中提取和利用可以为企业和组织提供许多未被挖掘的机会[1]。从文本中提取和模拟时间信息是信息提取的重要任务，也是构建时间线和轨迹的一个重要因素。然而，识别自然文本中时间相关的表达已经被证明是计算语言学中一个特别具有挑战性且活跃的领域，在很多领域都有着广泛的应用，如智能助理、文本挖掘、搜索引擎等。

尽管随着人工智能技术的不断提高，识别自然文本中时间相关的表达的研究也得到了迅速发展，但目前最先进的解决方案大多数仍然使用基于规则的系统来规范化时间性表达式，并根据规则为每个时间性表达提供一个确定的时间值。

TimeML是自2010年以来成为ISO标准的时间信息注释的主要倡议。它旨在将文本的时间分析过程与时间的表示和形式意义联系起来，为时间信息提供模型和注释方案，包括用于表示时间表达式的TIMEX3方案[2]。但对于时间线构建，TIMEX并不具备优势。首先，TimeML中的频率注释对于时间线构建是没有意义的。同时，TimeML的TIMEX规范化步骤并不总是直接用于时间线的重建，因为一些表达式仍然需要不同类型的时间链接来连接到日历信息。

为了解决这些问题，Calendar Expression(CALEX)作为一种新的注释模式问世，在时间表达层面上整合了时间线信息。与TimeML相比，CALEX模型删除了Frequency/Quantifier/Set注释，并引入了新的类型和规范化值[3]。特别是，像日期和持续时间这样的表达式根据它们如何与时间线相联系，被分配了不同的CALEX类型。

本文将设计和开发一个系统，使用基于规则的方法来识别英语文本中的时间表达，并使用CALEX（Calendar Expressions）的注释准则来识别和规范英语文本中的时间表达。

1.2 国内外研究现状

国外对于识别自然文本中时间相关的表达的研究起步较早，有着不同的方法和技术。在上世纪90年代缺少成熟的语料库和注释模式的环境下，研究人员大多只能采用基于规则的简易的事件时间表达关系识别模型来进行简单的分析工作，其使用的规则也多来自于研究人员的经验总结，具有一定的局限性。直到1992年，Lascarides、Asher和Oberlander等人提出了与时序有关的话语结构和两个事件之间关系的观点。他们试图将多种语言环境对应到5种话语关系中，分别是 narration、elaboration、explanation、background和result[4]。至此，通过时序关系与话语结构的对应来确定事件之间的时序关系引起了强烈的关注，并成为了当时语言学的研究重点。之后，研究人员又对时态（tense）、体态（aspect）、事件副词和修饰关系等因素进行了充分的研究与分析，并在2002年提出了识别时序关系的约束语言理论模型CONGEN[5,6]。与此同时，日后最为重要的注释模式TimeML也在一个关于问答系统的AQUAINT项目里首次被提出来的。经过多年的研究与改进，TimeML注释模式本身不仅得到了改进，一些用于后续实验研究的基于TimeML标准创建的语料库也陆续发布。目前，以英文新闻为语料来源的TimeBank是应用最广泛、具有广泛影响力的语料库[7]。在此之后，随着机器学习领域的进步与发展，研究人员逐渐开始思考机器学习技术在识别自然文本中时间相关的表达问题上的应用。为了验证机器学习方法在该问题上的可行性，Mani先后在2003和2006对练决策树分类器模型和最大熵分类器模型进行了尝试与探索，并分别取得了75.4%和62.5%的准确率[8,9]。在此之后，随着TimeML中<TIMEX3>标签的升级，机器学习方法的准确率得到进一步提升。在此之后，Natalia、Hegler和Ariane等人在2019年提出了一个新的可能对时间线的重建有用的注释模式，CALendar EXpression（CALEX）[3]。CALEX更好地利用与时间线上的锚点有关的时间表达属性，使得老方法有了新活力。目前，在西方学术界，开发并使用基于规则的使用CALEX注释准则的系统来识别和规范英语文本中的时间表达已然成为新的研究热点。与此同时，一些研究者则尝试采用深度学习算法，如卷积神经网络，对语料库进行分析，从而识别出时间表达[10]。

国内近年来也有许多基于TimeML标准进行中英文文本时间关系解析的研究，尤其是在中文文本中的应用。但主要关注点在于机器学习的方法和计算语言学的知识构造的结合，而不是基于规则的方法的研究。直到2014年，左亚尧，龙耀发和李杰骏等人提出了在划分时间表达式类别及规范形式基础上，采用正则表达式与Trie树结构相结合的方式构建出时间短语识别树，自动进行中文时间表达式的识别与分类的方法[11]。然而，由于中文的文本时间关系研究起步较晚，且中文与英文语法结构的差异性及汉语现象的繁多性，到现在为止也没有基于TimeML的中文语料库，相关研究以及平台的支持也还在进一步的探索中。

无论是国内还是国外的研究，都在不断提高识别精度和效率的同时，加入了更多的语言特征、语法规则等，并且研究的语料库也不断扩大。同时，随着深度学习技术的发展，基于深度学习的识别系统也在不断改进，从而提高识别精度。然而目前而言，基于规则的系统在识别自然文本中的时间表达任务上仍然有着十分领先的表现。

1.3 本文主要研究内容

目前，人们已经提出了许多使用TimeML中TIMEX注释模式的文本时间表达式识别以及时间线构建方案。本论文将分别使用TIMEX和CALEX两种注释模式分别对特定文章进行实验，将结果进行对比，分析CALEX在文本时间表达式识别以及时间线构建上的优势与劣势，旨在验证CALEX注释模式在时间线构建上的优越性并总结归纳。

同时，本课题设计并开发了一个使用基于规则的方法来识别英语文本中的时间表达，并使用CALEX（Calendar Expressions）的注释准则来标注和规范这些时间表达的文本时间表达式识别系统。该系统将作以一个可本地扩展的Python Package的形式，通过使用可扩展的规则集对用户输入的文本内容进行时间表达识别与处理标注，并将其处理为便于进一步处理的规范化的格式。

1.4 论文组织结构

本文旨在比较TIMEX和CALEX注释模式在识别时间节点并构建时间线的性能，然后提出了一种基于递归分层规则的文本时间表达式识别系统。具体的章节安排如下。

第一章 **绪论**。本章主要介绍了时间线提取问题的研究背景、目的、研究现状和意义，并引出本文的研究思路和主要研究内容。

第二章 **相关理论知识与技术**。本章详细介绍了与自然文本时间信息标注问题相关的内容，主要包括:自然文本时间信息标注、TimeML和CALEX注释模式、基于规则的方法以及评价指标。

第三章 **CALEX注释模式有效性的研究**。通过对比证明了CALEX注释模式的有效性，设计实验、对比结果，并给出了分析与结论。

第四章 **基于规则的文本时间表达式识别系统**。详细介绍基于规则的文本时间表达式识别系统，设计思路、性能测试，验证了系统的有效性。

第五章 **总结与展望**。本章总结了本文的主要工作、CALEX注释模式的有效性和优越性以及基于规则的文本时间表达式识别系统的优势，并对未来工作进行展望。

第二章 相关理论知识与技术

2.1 自然文本时间信息标注问题

2.1.1 问题定义

自然文本中时间相关的表达识别是自然语言处理领域中的一个重要课题，主要指从自然语言文本中识别和提取出时间相关的表达式，并将其转化为计算机可识别的格式的过程。在大量的文本数据中，时间相关的信息是非常有价值的，它可以用来描述事件的时间和状态，为人们提供重要的背景信息。然而，在自然文本中，时间表达的形式非常多样化，有时是显式的，有时是隐式的，并且没有明确的标准化。这导致了识别自然文本中时间相关的表达的困难。

这个问题存在一定难度，因为自然文本中时间表达的形式多种多样，可能是具体的日期和时间，也可能是相对的时间，如“下周”或“明天”。同时，自然语言中的时间表达可能存在缩写、拼写错误、不规则的格式等问题，导致识别困难。

目前主流的许多时间注释方案，它们都关注时间的不同方面：相对事件顺序[12–15]、事件持续时间[16,17]和明确的时间线索，如时间表达式[13,18–20]。

然而，对于大多数事件，现有方案仅提供部分事件时间信息，导致许多事件时间无法确定。缺少完全且有界的注释通常是由于时间信息的隐含性和不确定性而导致，这使得在绝对日历时间轴上定位事件变得非常困难。因此，处理时间不确定性，并将各种类型的时间信息整合到一个单一的方案中也是本问题的难点之一[21]。

因此，识别自然文本中时间相关的表达需要充分考虑自然语言的语言特征和语法规则，并使用合适的算法进行识别，以确保识别准确性。

2.1.2 问题关键点

时间信息标注问题的关键点主要有两个：规则和基于规则的方法。

规则是识别自然文本中时间表达的基础。它们是根据人们通常在日常生活中使用的语法和语义规则来定义的。例如，时间表达中常用的单位有小时、天、周、月和年。这些单位在自然语言中有明确的语法和语义，便于人们理解。此外，一些特殊的时间表达也具有独特的语法和语义，例如“明天”、“后天”等也可以将其看作一种特殊的规则。

基于规则的方法是解决该问题的关键点。它以语言学知识为基础，结合了计算机科学技术，以识别自然文本中的时间表达。基于规则的方法通常使用语法分析、语义分析、正则表达式等技术，通过对文本进行处理，识别其中的时间表达。基于规则的方法的优点在于灵活性和可靠性。它可以根据不同的应用场景，适当地修改和调整规则，以满足识别需求。此外，基于规则的方法可以很好地识别复杂的时间表达，从而解决复杂文本中时间信息的标注问题。

2.2 TimeML注释模式

TimeML注释模式是当前最具代表性且应用最广的时间信息注释模式，其开发是为了能够在一般领域的文本中识别事件和它们的时间顺序[2]。TimeML规范语言提供了一个标准模型，用于标记时间表达式（类型为Date、Time、Duration或Set）、事件（主要是动词或名词短语）及其时间顺序[13]，而且原则上可以应用于任何类型的文本。它是自然语言文本中的事件和时间表达的规范语言，能够在时间标记中捕捉不同的现象，将事件锚定在时间表示的表达上，组成事件-时间关系网络中的节点，并对相对事件表达进行排序，从而实现推理和其他与事件的时间方面有关的任务[22,23]。这不仅解释了句子的含义，更实现了使计算机理解句子逻辑结构这一语义理解的关键步骤[24]。TimeML在时间标记中捕获了三个不同的现象。(1) 在时间的广泛指代和表达上系统地设置了事件的词锚；(2)它相对排序文中的事件表述，既有内在的，又有话语的；(3)允许延迟(不明确)解释部分确定的时间表述[2]。

TimeML主要适用TIMEX3标签对时间信息进行建模，在该标签中，主要捕捉并建模了四种时间信息:时间（TIME），日期（DATE），持续时间（DURATION）和集合（SET）[25]。

2.2.1 时间（TIME）

时间类型的表达式是指一天中的某个时间，即使是以非常不确定的方式。时间和日期的区分主要在于表达式的颗粒度。如果表达式的粒度小于一天，那么该表达式就是一个时间。例如，下面的表达式就属于时间信息。

1. 文本

   描述已自动生成

图 2.1 时间信息表达式

值得注意的是，上边例子中的大多数都不是完全特定的时间表达。也就是说，它们都需要完整规格的上下文。除了**11:00 pm, Jan 1st, 2023**这一表达式外，其他表达式都需要更多的信息来完全表达它们所代表的时间信息。

2.2.2 日期（DATE）

日期类型的表达式可以被认为是指日历时间的任何表达式。日期通常是一天或更大的时间单位。同时，和时间类型一样，日期类型所能够表达的时间信息也经常是不明确的，需要更多的信息来完全表达它们所代表的时间信息。例如，下面的表达式就属于日期信息。

1. 文本

   描述已自动生成

图 2.2 日期信息表达式

2.2.3 持续时间（DURATION）

持续时间类型的表达式可以被认为是指在时间轴上持续的任何表达式。时间和日期通常是一个单独的时间点，而持续时间则更多关注于两个或多个时间点在时间轴上的跨度。例如，下面的表达式就属于持续时间信息。

1. 文本

   描述已自动生成

图 2.3 持续时间信息表达式

2.2.4 集合（SET）

集合时间类型的表达式主要是用于表达事物发生的频率信息。例如，下面的表达式就属于集合信息。

1. 图形用户界面, 文本

   中度可信度描述已自动生成

图 2.4 集合信息表达式

2.3 CALEX注释模式

CALEX指的是一种限于时间表达和概念的时间注解模式，可以（直接或不直接）与绝对时间线相连。该模式的关键创新点是更好地利用与时间线上的锚点相关的时间表达属性，包括引入某些与时间线相关的概念。

与基于TimeML的TIMEX定义对比，CALEX排除了一些内容，因为它们不能直接用于时间线重建:

**频率/集合/数量**:例如twice a day，every 4 weeks等表达

**持续时间**:例如3 days等表达

**时间**:例如last night，3 pm等表达

总的来说，CALEX是TIMEX的一个子集，只包括可以与日历连接的表达式。在CALEX注释模式中，主要有三个主要组成部分:类型（TYPE），元数据（METADATA）和值（VALUE）[22]。

2.3.1 类型（TYPE）

类型元素用于定义日历表达式的类型。在CALEX注释模式中，可能的日历表达式类型主要包含如下几种:

1. CALENDAR: 该类型涵盖了所有不需要任何元数据就能提供最终规范化值的日历表达式，例如：在不同的时间粒度中明确引用日历、时间戳、明确的范围以及不能与时间线建立明确联系的时间段(DURATION)
2. AGE: 任何与年龄相关的时间信息表达式，可以是当前节点上的年龄，也可以是过去某个时间节点的年龄信息。
3. DOMAIN: 要么明确定义特定领域概念的值（2010年6月6日入学），要么是对特定领域概念的引用（入学后第4天）
4. DCT: 文件创立时间，是用于规范其他不明确时间信息的重要锚点。
5. CONTEXT: 指向时间背景的表达式，由最后提到的时间参照或文件中的最新时间参照表示。

2.3.2 元数据（METADATA）

元数据这一特征的使用，为没有明确时间锚定的CALEX提供一种计算上更有效的方式来计算特定的时间参考。这个特征的一个重要方面是，它可以在其定义中包括概念。这些概念也可以在METADATA特征中明确设置，以确保使用原始值，从而使最终的日历表达式标准化。即可以使用doc.DCT，doc.LAST，person.DOB，person.AGE等从文本中获取的概念以及set/add/sub/next/prev等方法对明确时间锚定的CALEX，例如"last Friday"，提供了一种合理的定义方式。

2.3.3 值（VALUE）

该组件给出了一个日历表达式的规范化值，主要遵循之前的TimeML符号，并有一个扩展：范围(RANGE)被用于规范化[begin,end]形式的时间段。

2.4 基于规则的方法

基于规则的方法是所有AI算法的最早类型。实际上，在人们考虑使用机器学习实现一切自动化之前，基于规则的方法是实现该目的的最好办法。基于规则的算法的本质很简单:人为定义规则，之后应用规则，并在这个过程中不断的赋予和调整权重与规则结构，直到最后达到理想的效果。

基于规则的方法通常具有非常高的精度，因为规则是用户定义的，是用户经验与智慧的成功。当人类用户定义规则时，已经可以基本确定规则的正确性与实用性。然而基于规则的方法缺有着非常低的泛化能力。这是因为人类用户永远无法一次性涵盖所有可能的情况。而当用户仅仅将目标关注在某一特定区域时，低召回率的问题也就不再那么重要，人们也可以通过不断的赋予和调整权重与规则结构的方式进一步减少这个问题的影响。

2.5 医疗文本信息提取

临床自然语言处理（Clinical NLP）系统分为两大类：基于规则的和基于统计的。基于规则的自然语言处理系统将专家知识编码为一组结构化规则或模板，应用于非结构化文本时，可产生结构化信息。例如，一条规则可以指定单词、短语或词性的模式，以表明特定类型实体的存在：“如果‘received’后跟着一个名词，后面是‘for’和一个疾病名称，则假定该名词是药物名称[26]。”许多表现最佳的临床自然语言处理系统都是基于规则的：2018年王等人对263篇临床文本挖掘文章进行了综述，其中171篇（65％）使用基于规则的方法[27]。然而，基于规则的系统有两个重要缺点。首先，领域专家通常需要花费大量时间和精力来构建规则。其次，由于它们是针对特定领域的，所以不适用于新问题。例如，用于识别文本中的药物名称的基于规则的系统除了识别文本中的药物名称之外，其他方面都不好。

另一种选择是使用统计自然语言处理系统，该系统使用统计学习（即机器学习）算法构建。例如，如果提供了一个已标注了所有药物名称的文本，该算法将尝试识别表明特定文本范围是药物名称的模式。学习算法本身通常是任务独立的，这是它们的关键优势之一。然而，统计学习算法需要有标注的训练数据，在临床领域中，这种数据通常是有限的或不存在的[28]。此外，隐私问题通常使得在机构之间共享训练数据变得不可能。因此，虽然自然语言处理社区越来越倾向于机器学习而远离基于规则的方法，但临床文本挖掘仍然强调规则[29]。

2.6 MTSample数据集

MTSamples数据集包括上传到社区平台网站的5000份来自不同专业的医疗转录报告样本。该数据集已被广泛地应用于的医学领域的自然语言处理(NLP)研究。

数据集中有40个医疗专业，如"外科"、"咨询-病史和治疗"和"心血管/肺科"。每个专科都包含一些样本报告。这些报告是自由文本，标题根据专科不同而变化。然而，所有的报告都有一个描述字段，这是对报告摘要的一个很好的近似。每份报告的长度因专业不同而有很大差异，报告正文平均为589字，描述为21字。图2.1是MTSamples报告的一个例子。

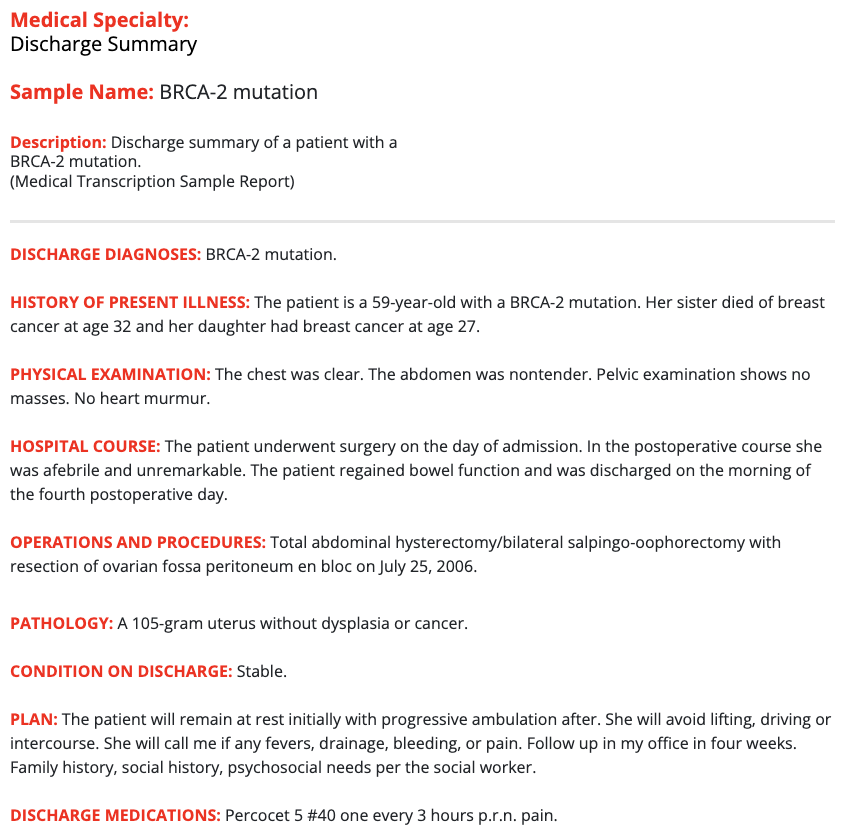


图 2.5 MTSamples报告

2.7 评价指标

召回率（Recall）是用于评估分类模型的性能指标之一，也称为查全率（Sensitivity）。它指的是模型正确检测出所有正样本的能力，即模型在所有正样本中正确检测出的比例，其计算如公式1所示

在某些场景下，正样本的数量可能很少，而模型的主要任务是准确检测出所有正样本，即不漏掉任何一个真正例。例如，在医学影像诊断领域中，模型需要识别肿瘤等病变区域，漏诊可能会导致严重的后果。此时，召回率可以很好地反映模型的性能，帮助判断模型是否适合实际应用。

2.8 本章小结

本章介绍了自然语言处理中关于时间信息抽取的研究现状。首先，本章讨论了时间信息的重要性和应用场景，以及时间信息抽取的目标和挑战。接着，本章概述了常用的时间信息标注模式，包括TimeML和CALEX模式，并详细介绍了这些模式中各自涵盖的时间表达式类型及其语法特点。同时，本章还介绍了基于规则的方法在时间信息抽取中的应用，以及其优缺点。之后，本章介绍了MTSample数据集，该数据集是医疗领域自然语言处理研究中常用的数据集，包含5000份来自不同专业的医疗转录报告样本。在本章的最后，本章引入了时间信息抽取的评价指标——召回率，以及其在不同场景中的应用。总的来说，本章详细介绍了时间信息抽取领域的研究现状和相关技术，为后续的时间信息抽取研究提供了基础和方向。

第三章 CALEX注释模式有效性的研究

本章主要探讨了CALEX注释模式在时间信息抽取中的有效性。为了研究该注释模式的有效性，本文首先收集了医疗转录报告样本，并进行了预处理。接着，本文使用CALEX和TIMEX时间注释模式对这些报告进行了标注，并比较了它们之间的差异。最后，通过对实验结果的分析，展示了CALEX注释模式在时间信息抽取中的优越性。本章的研究结果有望为时间信息抽取领域的研究者提供一种新的思路和方法，同时也为相关领域的应用提供了有力的支持。

3.1 数据收集与预处理

3.1.1 数据来源

在CALEX注释模式有效性的研究中，本文选用MTSample数据集。

3.1.2 数据描述

该数据集是由Kaggle收集的，并进行了更新。该数据集包含了大约5000条病人的记录，以及他们的一般描述，也就是转录的简要描述。数据集包括102份出院总结和232份其他报告，其中包括咨询记录、进展记录、ICU、急诊室访问和身体检查记录。这个语料库包含15089、4814和5518个事件，分别标记为问题、测试和治疗。与i2b2数据集不同的是，这些笔记不包含关于入院或出院时间的明确信息。本文在该数据集中选取了20份笔记用于实验。

3.1.3 数据预处理

在数据预处理阶段，本文主要适用了人工标注的方法，在原本的文本信息中找到CALEX信息，并将其标注出来。同时，在预处理阶段，本文也对易错的区域进行了标注。图3.1是需标注信息的例子，黑色加粗区域为需要的CALEX表达，红色标注区域为不想要但易错的区域。

文本

描述已自动生成

图 3.1 MTSample数据预处理

值得注意的一点是“daily”表达在这里被认为是TIMEX中的FREQUENCY信息，并非CALEX信息。

3.2 注释模式对比

在对比两种注释模式的实验中，本文主要采用人工识别时间表达并构建时间线的方式来对比两种注释模式在构建时间线这一任务上的表现。在接下来的篇章里，本文将使用一段来自MTSample的段落作为示例进行分析。该患者的出生时间是1990年一月七日，原文如图3.2所示。

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

图 3.2 示例段落原文

3.2.1 TIMEX

通过对示例段落的识别，可以发现其中有着四个TimeML注释模式定义的TIMEX元素，如图3.3所示。

图形用户界面, 文本, 应用程序

描述已自动生成

图 3.3 示例段落中的TIMEX元素

在成功提取TIMEX元素后，可以继续根据事件标签创建时间线，如图3.4所示。

日程表

描述已自动生成

图 3.4 TIMEX注释模式下的时间线

通过观察图3.4，可以清晰的发现虽然TIMEX注释模式可以很好的提取给定文本中的明显的时间表达，但很多相对的时间节点则无法进行识别，比如患病时间和幻觉发作时间，这两个时间节点都需要基于其他时间节点进行计算从而得到。除此之外，TIMEX还会识别多余的信息，比如实例中的“twice a day”这样的频率信息，这种信息对于时间线的建立是没有帮助的。因此，可以发现TIMEX注释模式在构建医学时间线时存在自己的缺陷。

3.2.2 CALEX

与TIMEX相比，CALEX注释模式不仅去除了对不需要的频率等信息的识别，还支持了时间的运算，进而支持了相对时间的识别。通过对示例段落的识别，可以发现其中有着五个CALEX注释模式定义的时间表达，如图3.5所示。

图形用户界面, 文本, 应用程序

描述已自动生成

图 3.5 示例段落中的CALXE元素

在成功提取CALEX元素后，可以清楚的发现原本TIMEX中没有识别到的“age of 14”，“On admission”等表达都被计算了出来。接下来本文将继续根据事件标签创建时间线，如图3.6所示。

日程表

低可信度描述已自动生成

图 3.6 CALEX注释模式下的时间线

通过观察图3.6，可以清晰的CALEX很好的完成了时间线构建的工作。毋庸置疑，在面对医学文件时，CALEX有着更强的处理时间线构建任务的能力。

3.3 实验结果

使用时间线的完整度作为度量标准，其计算方式如公式2所示：

本实验选取的20份MTSample样本的完整度结果如图3.7所示。可以观察到，在总体上CALEX有着比TIMEX更高的完整度，这表明了CALEX注释模式在识别医疗文件时间节点上较CALEX注释模式存在着一定的优势。

图表, 条形图

描述已自动生成

图 3.7 实验完整度结果

3.4 本章小结

本章研究了CALEX注释模式在时间信息抽取中的有效性，并使用MTSample数据集进行实验。实验中采用了人工标注的方法对数据进行预处理，并比较了CALEX和TIMEX两种时间注释模式的差异。结果显示，CALEX注释模式具有比TIMEX更高的时间线完整度，且在识别医疗文件时间节点方面具有优势。

具体来说，实验结果表明CALEX注释模式不仅去除了对不需要的频率等信息的识别，还支持了时间的运算，进而支持了相对时间的识别。相比之下，TIMEX注释模式在构建医学时间线时存在一定的缺陷。该研究为时间信息抽取领域的研究者提供了新的思路和方法，也为相关领域的应用提供了有力的支持。

第四章 基于规则的文本时间表达式识别系统设计与实现

在本章节中，本文将深入研究基于规则的文本时间表达式识别系统的设计与开发。时间表达式识别在自然语言处理中具有重要意义，它是许多NLP任务的基础，如信息抽取、问答系统、文本归档和语音识别等。基于规则的系统是一种常见的方法，通过定义规则并对规则进行组合和优化，实现对时间表达式的准确识别。本章节将首先介绍该系统的描述和要求，包括该系统所需完成的任务以及应具备的功能。然后，本文将详细探讨系统的设计与架构，包括规则定义和模块交互等。最后，本文将对该系统进行测试和评估，评估其在给定数据集和实验条件下的性能和准确度。本章节的目标是设计和开发一个可靠、高效和准确的基于规则的时间表达式识别系统。

4.1 系统描述

4.1.1 问题描述

为了更好地帮助医务人员从临床记录中提取有价值的时间信息，本文设计并开发了这个Python模组来帮助他们构建时间线信息，提升效率。此外，这个模组未来还将用于Drexel University 和University of Pennsylvania关于医疗行业的更多合作研究项目。

4.1.2 环境要求

这个模组是用Python编程语言创建的，并使用Datefinder 0.7.3。要导入这个模组，至少需要有64MB的内存和400MHz以上的CPU。另外，该模组至少要使用10兆字节的硬盘空间来存储该模组。

以下列表代表了使用该Python模组通常所需的最低要求。

- 现代操作系统。

- Windows 7或更高

- Mac OS X 10.11或更高，64位

- Linux：RHEL 6/7，64-bit

- x86 64位CPU(英特尔/AMD架构)

- 4GB内存

- 5 GB可用磁盘空间

大多数用户会发现，最近几年购买的任何电脑都能满足（通常超过）这些硬件要求。

4.1.3 系统要求

用户可以输入一个文本或一个文本文件，并给出文件的创建时间。Python软件包会自动应用内置的识别规则，用日历表达式对文本进行注释，并以JSON格式输出。用户还可以将特定格式的识别规则添加到CSV文件中，在验证规则的可用性后，将其应用于CALEX的识别过程。这个软件包CALEX为用户提供了一个简单的方法来完成以下要求：

表 4.1 系统要求

|  |  |
| --- | --- |
| **要求** | **描述** |
| 可扩展的规则 | 以可编辑的形式存储规则，以确保它们可以在任何时候被改变。 |
| 分词处理 | 作为第一步，将应用于要处理的文本进行分词处理。 |
| CALEX识别 | 识别每个给定文本文件中的CALEX表达式，并给出它们在句子中的位置等信息，将它们翻译成机器可处理的格式。 |
| JSON结果格式 | 对文本的识别结果将以JSON文件格式存储。 |

4.2 系统设计

4.2.1 系统架构

在CALEXText Python模组中，总共定义了三个类来实现这些要求。CALEXText类被用于初始文本和最终结果的存储，以及主要功能的实现。datepoint类用于存储包括年、月、日在内的日期点数据，以及上一个数据锚点的信息。daterange类被设计用来使用datepoint存储时间轴上的一段时间。图4.1展示了系统整体架构以及各个类和函数的具体信息。

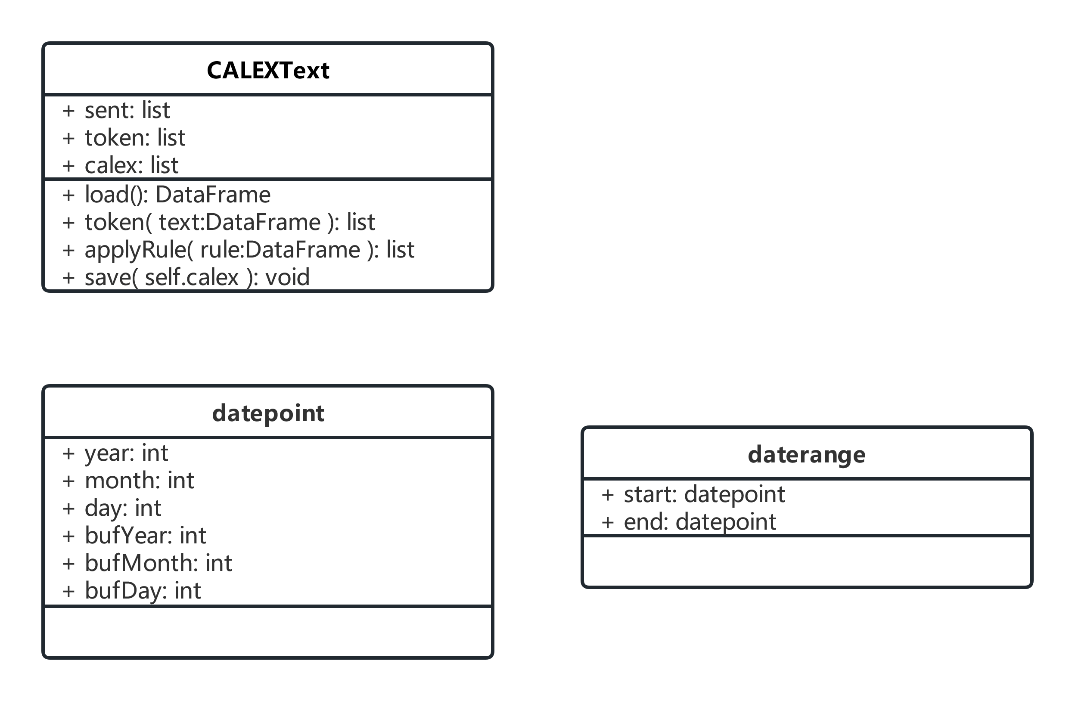


图 4.1 CALEX系统UML图

在CALEXText类中定义了三个属性用于存储本文所关注的文本数据的三个信息：sent-存储原始文本信息，token-存储原始文本的分词信息，calex-存储原始文本中的CALEX表达信息以及位置信息。除此之外，CALEXText类定义了四个方法用以整个系统的运行：

**load()**：用于读取本地规则文件以及文本信息

**token()**：用于对读取的文本信息进行分词操作

**applyRule()**：用于将读取的规则应用到分词后的文本信息上

**save()**：用于将结果信息存储为JSON文件

在datepoint类中，定义了六个属性用于存储时间节点信息，其中year，month，和day属性用于记录当前时间表达的具体信息，bufYear，bufMonth，和bufDay属性用于记录上一高层级时间表达的具体信息，进而确保对于类似“yesterday”等相对时间表达的准确识别。

在daterange中，其存储的信息本质上是一对datepoint对象，分别对应时间范围的起点和终点，两个属性都可以为空值来应对类似“before 1923”等开区间时间范围的情况。

4.2.2 系统流程

图4.2展示了用户使用CALEXText系统获取文本信息中的时间信息的整体流程。

Diagram

Description automatically generated

图 4.2 CALEXText系统流程图

首先用户先调用loadRule模块，读取本地的以CSV形式存储的规则文件，并将规则存储为系统内部的可随时读取的DataFrame对象。之后，调用loadText模组，根据需求调用不同的方法进而读取文本文件或文本文字。之后，为了方便后续的处理和分析，将读取进来的文本数据进行分词操作。之后，在分词操作后的文本数据上调用applyRule模块，读取并应用loadRule模块读取的规则，最终获取存储时间信息，分词信息，位置信息等所需信息的JSON对象。通过save模块，可以将该结果JSON对象导出到JSON文件。

4.2.3 规则设计

在自然语言中，时间表达可以由单个时间单元组成，例如“2 PM”或“July 10th”，也可以由多个时间单元组成，例如“July 10th, 2022 2 PM”。此外，时间表达也可以由时间段或时间范围组成，例如“9 AM to 5 PM”或“from 2022 to 2023”。

因此，时间表达在语言中具有层级结构，其中较小的时间单元可以组成更大的时间单元，更大的时间单元可以组成更大的时间单元，形成一个层次结构。这种层级结构使得时间表达更加灵活和丰富，能够表达不同的时间概念和粒度。

递归的层级结构规则设计可以更好地处理具有层次结构的数据和信息。递归规则是指规则可以调用自身来处理更深层次的嵌套元素。这种设计可以让规则适应不同层次的时间表达，并且可以通过递归来处理任意深度的嵌套层次。

此外，递归的层级结构规则设计还具有灵活性和可扩展性。例如，如果时间表达模式发生变化，只需要修改一些递归规则，而不需要对整个系统进行大规模修改。

因此，本文系统设计有一个五层的逐层递归系统用于处理自然文本的时间信息，将其转化为机器可识别的模式：

**第0级规则**将文本标记作为输入，并将其转换为相应的格式，例如，0将被转换为0，5将被转换为5，最后将被转换为-1。

**第1级规则**用于找到直接的时间表达式，例如从转换后的标记中找到1/1/2022，并将其转换为数据点对象。

**第2级规则**可用于识别时间表达式，例如介词的组合，如"at"和"in"，以及日期点。为了得到简明的时间表达式，这个过程将被反复进行，直到文本中所有的时间表达式都是最简明的格式，也就是日期点对象形式。该规则不会处理像"从"、"到"和"直到"这样可以用来表达时间范围的介词。

**第3级规则**将有一个类似于第2级规则的过程。第3级规则可以迭代识别带有介词如"从"、"到"和"直到"的时间表达，这些表达时间范围的信息，直到最简洁的格式。

**第4级规则**将把结果返回到CALEXText类的属性中。

4.3 系统测试与评估

4.3.1 数据来源

在对CALEXText系统的测试与评估实验中中，本文依旧选择使用MTSample数据集。本文在该数据集中选取了100份笔记用于实验。

4.3.2 实验设计

通过将CALEXText系统应用到给定的100份笔记上进行时间表达信息的识别与提取，可以得到一百份由系统给出的时间表达信息。由于已经对100份笔记进行了人工的识别与处理，得到了正确的时间信息对象列表。通过对比系统的结果与人工识别的结果，可以很好的去评估CALEXText系统在识别MTSample数据集中时间表达的能力。

4.3.3 实验结果

通过实验，可以得到100个对应不同随机的100份MTSample笔记的实验结果。为了更好的对结果进行对比，本文使用召回率作为指标。图4.3展示了召回率的分布信息。

图表, 散点图

描述已自动生成

图 4.3 实验召回率结果

通过对实验结果的观察，可以发现CALEXText系统在给定的100份随机的笔记上进行时间表达信息的识别与提取时平均的召回率为0.90，大多数笔记的召回率主要分布在0.85以上，只有小部分笔记给出了0.7到0.85之间的召回率。

4.4 本章小结

为了更好地帮助医务人员从临床记录中提取有价值的时间信息，本章提出了CALEXText Python模组来帮助他们构建时间线信息，提升效率。同时，本章提出了层次递归的规则设计模式，极大的简化了规则的复杂度的同时提升了规则的适用性。最后，通过随机抽取100份MTSample医疗笔记作为实验数据集对CALEXText系统进行测试，最终获得十分令人满意的实验结果。

第五章 总结与展望

5.1 论文工作总结

提取和计算文本中的时间信息是构建时间线和轨迹的重要组成部分。然而，自然文本中的时间相关表达的识别被证明是计算语言学中一个具有挑战性且活跃的领域。尽管许多学者尝试使用机器学习方法来解决此问题，但目前最先进的解决方案仍然主要使用基于规则的系统来规范化时间性表达式，并为每个时间性表达式提供一个确定的时间值。本文的目的在于比较TIMEX和CALEX两种注释模式在此问题上的表现，证明CALEX的优越性，并介绍一种使用CALEX注释规则的基于规则的文本时间表达式识别系统——CALEXText，主要工作如下：

1. +++++。本文选择了MTSample医疗笔记数据集作为实验数据集，通过随机抽取笔记并分别尝试使用TIMEX和CALEX两种注释模式进行时间节点的识别与时间线的构建，以完整度为评价指标，验证CALEX注释模式在该任务上的优越性。

2. +++++。本文提出了一种基于规则的文本时间表达式识别系统，该系统参照CALEX注释模式，使用分层递归的规则设计，在MTSample医疗笔记数据集上完成文本时间表达式识别任务，验证该系统的有效性并分析其性能。

从实验结果可以看出，

1. CALEX注释模式在识别文本，尤其是医疗文本中的时间节点相较于TIMEX注释模式有着较大的优势，同时CALEX注释模式也更有利于时间线的构建。

2. 本文提出的基于规则的文本时间表达式识别系统是有效的，它可以应对复杂的时间表达，并通过规则的匹配和时间关系的计算对相对时间表达有着很好的准确率，有一定的实际应用价值。

5.3 未来展望

本文所开展的所有实验仅探索了CALEX注释模式和CALEXText系统在医疗文本上进行提取和计算文本中的时间信息任务的有效性，并没有探索其在其他领域的有效性。

CALEXText系统作为一种基于规则的文本时间表达式识别系统，已经在时间信息处理领域取得了一定的进展。未来，CALEXText系统的发展应该着重考虑适应用户特殊的表达习惯和排除易混淆非时间表达式的识别。具体地，可以探索以下几个方向：

首先，CALEXText系统可以通过学习用户的表达习惯，提高时间表达式的识别准确性和效率。这一方向的研究需要建立用户表达习惯的数据集，收集并分析不同用户的表达习惯，并对系统进行相应的改进和优化。例如，对于用户常用的时间表达方式，可以建立相应的规则，加强对其的识别和处理。

其次，CALEXText系统可以针对易混淆的非时间表达式进行排除，提高时间表达式的识别准确性和稳定性。该方向的研究需要对文本中的非时间表达式进行分析和分类，找出易混淆的表达式，并建立相应的规则进行排除。例如，对于表示持续时间的词汇，如“长”、“久”，可以使用上下文信息进行判断，避免其被误识别为时间表达式。

综上所述，CALEXText系统在未来的发展中应该注重用户表达习惯和非时间表达式的识别问题。这些方向的研究可以提高系统的实用性和适用性，进一步推动时间信息处理领域的发展。

参考文献

[1] Abdullah M H A, Aziz N, Abdulkadir S J, et al. Systematic Literature Review of Information Extraction from Textual Data: Recent Methods, Applications, Trends, and Challenges[J]. *IEEE Access*, 2023, 11: 10535-10562. https://doi.org/10.1109/ACCESS.2023.3240898.

[2] Pustejovsky J, Castaño J M, Ingria R, et al. TimeML: Robust Specification of Event and Temporal Expressions in Text[C]. *New Directions in Question Answering*, 2003.

[3] Viani N, Tissot H, Bernardino A, et al. Annotating Temporal Information in Clinical Notes for Timeline Reconstruction: Towards the Definition of Calendar Expressions[C]. *Proceedings of the 18th BioNLP Workshop and Shared Task*, 2019: 201-210. https://doi.org/10.18653/v1/W19-5021.

[4] Lascarides A, Asher N, Oberlander J. Inferring Discourse Relations in Context[C]. *30th Annual Meeting of the Association for Computational Linguistics.* Newark, Delaware, USA: Association for Computational Linguistics, 1992: 1-8. https://aclanthology.org/P92-1001.

[5] Dorr B, Gaasterl T. Constraints on the Generation of Tense, Aspect, and Connecting Words from Temporal Expressions[J]. *UMIACS Technical Reports*, 2002.

[6] Hitzeman J, Moens M, Grover C. Algorithms for Analysing the Temporal Structure of Discourse[J]. *Proceedings of the Annual Meeting of the European Chapter of the Association for Computational Linguistics*, 2002. https://doi.org/10.3115/976973.977009.

[7] Boguraev B, Pustejovsky J, Ando R, et al. TimeBank evolution as a community resource for TimeML parsing[J]. *Language Resources and Evaluation*, 2007, 41: 91-115. https://doi.org/10.1007/s10579-007-9018-8.

[8] Mani I, Schiffman B, Zhang J. Inferring Temporal Ordering of Events in News[C]. *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, 2003. https://doi.org/10.3115/1073483.1073502.

[9] Mani I, Verhagen M, Wellner B, et al. Machine Learning of Temporal Relations[M]. *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, 2006. https://doi.org/10.3115/1220175.1220270.

[10] Meng Y, Rumshisky A. Context-Aware Neural Model for Temporal Information Extraction[C]. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, 2018: 527-536. https://aclanthology.org/P18-1049.

[11] 左亚尧, 龙耀发, 李杰骏. 基于规则的中文时间表达式识别与规范化[J]. 广东工业大学学报, 2014, 31(3): 88-94.

[12] Pustejovsky J, Castano J M, Ingria R, et al. TimeML: Robust specification of event and temporal expressions in text[J]. *New Directions in Question Answering*, 2003, 3: 28-34.

[13] Pustejovsky J, Lee K, Bunt H, et al. ISO-TimeML: An International Standard for Semantic Annotation[C]. *Proceedings of the Seventh International Conference on Language Resources and Evaluation*, 2010: 394-397.

[14] Styler Iv W F, Bethard S, Finan S, et al. Temporal annotation in the clinical domain[J]. *Transactions of the association for computational linguistics*, 2014, 2: 143-154.

[15] Ning Q, Wu H, Roth D. A Multi-Axis Annotation Scheme for Event Temporal Relations[J]. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, 2018: 1318–1328. http://dx.doi.org/10.18653/v1/P18-1122

[16] Pan F, Mulkar R, Hobbs J R. An Annotated Corpus of Typical Durations of Events[C]. *Proceedings of the Fifth International Conference on Language Resources and Evaluation*, 2006: 77-82.

[17] Pan F, Mulkar-Mehta R, Hobbs J R. Annotating and learning event durations in text[J]. *Computational Linguistics*, 2011, 37(4): 727-752.

[18] Ferro L, Gerber L, Mani I, et al. TIDES-2005 Standard for the Annotation of Temporal Expressions[S]. MITRE Corporation, 2007.

[19] Bethard S, Parker J. A semantically compositional annotation scheme for time normalization[C]. *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, 2016: 3779-3786.

[20] Setzer A. Temporal information in newswire articles: an annotation scheme and corpus study[D]. University of Sheffield, 2002.

[21] Leeuwenberg A, Moens M F. Towards Extracting Absolute Event Timelines From English Clinical Reports[J]. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2020, 28: 2710-2719. https://doi.org/10.1109/TASLP.2020.3027201.

[22] Tissot H, Roberts A, Derczynski L, et al. Analysis of Temporal Expressions Annotated in Clinical Notes[C]. *Proceedings of the 11th Joint ACL-ISO Workshop on Interoperable Semantic Annotation*. London, UK: Association for Computational Linguistics, 2015. https://aclanthology.org/W15-0211.

[23] Boguraev B, Ando R. TimeML-Compliant Text Analysis for Temporal Reasoning[C]. *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence*, 2005: 997-1003.

[24] Sanampudi S, G.Vijaya K. Temporal Reasoning in Natural Language Processing: A Survey[J]. *International Journal of Computer Applications*, 2010, 1. https://doi.org/10.5120/100-209.

[25] Pustejovsky J, Knippen R, Moszkowicz J, et al. Temporal and Event Information in Natural Language Text[J]. *Language Resources and Evaluation*, 2005, 39: 123-164. https://doi.org/10.1007/s10579-005-7882-7.

[26] Percha B. Modern Clinical Text Mining: A Guide and Review[J]. *Annual Review of Biomedical Data Science*, 2021, 4(1): 165-187. https://doi.org/10.1146/annurev-biodatasci-030421-030931.

[27] Wang Y, Wang L, Rastegar-Mojarad M, et al. Clinical information extraction applications: A literature review[J]. *Journal of Biomedical Informatics*, 2018, 77: 34-49. https://doi.org/10.1016/j.jbi.2017.11.011.

[28] Spasic I, Nenadic G. Clinical Text Data in Machine Learning: Systematic Review[J]. *JMIR Medical Informatics*, 2020, 8(3): e17984. https://doi.org/10.2196/17984.

[29] Kreimeyer K, Foster M, Pandey A, et al. Natural language processing systems for capturing and standardizing unstructured clinical information: A systematic review[J]. *Journal of Biomedical Informatics*, 2017, 73: 14-29. https://doi.org/10.1016/j.jbi.2017.07.012.

致 谢

在此，我要向许多人表达我最衷心的感谢和敬意。

首先，我要感谢我的导师赵志立老师。感谢您在整个毕业设计过程中对我的支持和指导。您的专业知识、悉心指导和热情帮助，使我受益匪浅。在您的帮助下，我顺利完成了本次毕业设计。感谢您一直以来的耐心指导和关心，让我能够顺利地完成这一挑战性的任务。

其次，我要感谢我的队友们。感谢你们在整个毕业设计过程中的帮助和支持。我们一起并肩作战，共同解决了许多问题。在你们的帮助下，我不仅学到了许多知识，也获得了宝贵的团队合作经验。感谢你们一直以来的支持和鼓励，让我在这个项目中感到了温暖和动力。

最后，我要感谢我的家人和朋友们。感谢你们一直以来的支持和鼓励，让我在学习和生活中充满动力。感谢你们的理解和包容，在我面临困难时给予我温暖的慰藉。我会继续努力，为你们感到骄傲！

在此，再次向以上所有人表示最衷心的感谢和最崇高的敬意！

毕业论文（设计）成绩表

|  |
| --- |
| 导师评语  **建议成绩 指导教师（签字）** |
| **答辩委员会意见**  **答辩委员会负责人（签字）** |
| **成绩 学院（盖章）**  **年 月 日** |