**中国证券业协会2018年重点课题**

**研究报告**

**课题名称： 基于知识图谱与深度学习事件驱动的证券公司风险监测预警机制研究**

**课题研究单位: 中泰证券股份有限公司**

**联合研究单位: 北京邮电大学**

**课题负责人: 李恒第 赵成林**

**课题组成员: 肖晓超 王健 许方敏 仇超**

**中国证券业协会**

**2018年8月**

摘要

近年来随着科学技术的不断进步和金融业务的日益发展，金融科技融合了金融和科技两方面的内容正在加速迭代风靡全球。大数据、互联网、云计算、人工智能、区块链等新兴技术接踵而至，逐步从概念走向成熟，金融科技已经逐渐形成新兴产业，带动和引领全球新一轮的生产和服务效率的变革浪潮，将进一步深刻改变全球金融业态的未来格局。大数据时代的开启为海量数据资源的融合与智能解析提供了完整的新路径。面向不同数据源、不同数据结构和不同数据类型的企业风险信息的集成、融合及其深度挖掘，可以使已知数据中的隐含知识得以最大化的释放，从而更好地认识与识别证券公司风险。知识图谱，即语义互联，是一种基于图的数据结构，将多元异构信息连接在一起得到的关系网络，提供从“关系”角度分析、推理问题的能力。目前，知识图谱在国内的总体应用还处于初始阶段，但在公安、电商等多个行业中已得到了较为广泛的应用。人工智能与知识图谱作为认知智能的关键技术，正成为智能金融中新的研究热点。这两种技术在金融领域的应用场景有高度的重合，往往是互为依托、互为补充。

金融科技的蓬勃发展不仅是实体经济发展的重要支撑，更是国家战略层面的必然选择。2017年由人民政协报财经周刊主办的第16期财经智库沙龙提出“防控系统性金融风险金融科技当作为”。2018中国金融科技产业峰会的主要议题为“利用金融科技，强化系统性金融分析的防范能力”，讨论了金融科技在银行、保险、证券和互联网金融等多个金融领域的风险管理应用。无论是国家战略层面，还是大型公司的发展布局来看，深度学习无疑是目前学习领域最为火热的研究方向，例如谷歌的大脑计划，百度的深度学习实验室等等。深度学习通过对人类大脑的神经连接结构进行模拟而建模，通过对连续多个变换阶段进行分层抽象，进而对数据的特征进行模拟描述。近年来深度学习在语音识别、图片识别、人工智能等众多领域中取得突破性进展。随着人工智能、大数据等人工智能新兴技术的兴起，“知识图谱”的概念也应运而生。“知识图谱”就像摩尔摩斯破案一样，利用大数据采集散步在各个角落的碎片化信息并按标准化思考方式整理，再将各个相关联的信息关联起来，并挖掘相互之间的关联关系与影响程度。

随着证券行业监管日益严格，证券公司已经把公司风险的防范与把控提升到非常重要的高度，如何利用深度学习、自然语言处理、数据挖掘等人工智能技术结合知识图谱对证券公司风险进行科学的分析、预测，实现公司内部风险信息监测预警，为公司提供科学的投融资决策及促进业务发展，已经成为证券公司研究的热点问题。

基于此，本课题拟对基于知识图谱与深度学习事件驱动的证券公司风险监测预警机制展开研究。拟建立证券公司自营以及融资融券、股票质押等业务的持仓主体以及关联方的公司主体知识图谱与风险事件图谱，并结合深度学习模型对与证券公司相关的公司主体进行风险监测、预警，从而对证券公司总体的风险程度进行监测、预警，从而防范证券公司风险，避免由于证券公司风险引起系统性风险的事件发生。

本课题涉及的创新内容主要包括三个方面：

1. 公司主体知识图谱的创建。公司知识图谱主要是描述上市公司的固定属性以及各种关联关系，基本属性包括公司介绍、高管、集团、股东、债务、行业、上游、下游、概念、主营业务等属性。上市公司的信息来源较为复杂，多数是半结构化、非结构化的文本信息。在公司知识图谱的构建过程中，最重要的三个环节就是知识单元的抽取、知识单元间关系的识别以及知识图谱的绘制。而其中尤其以知识单元抽取、知识单元间关系的识别最为关键。
2. 风险事件知识图谱的创建。相对于上市公司图谱的构建，事件图谱的构建中其基本信息数据更加关注宏观政策、行业新闻、网络舆情、行政处罚情况、经营情况、公司公告、全网新闻、新闻事件等与股市相关的互联网数据。
3. 利用深度学习结合知识图谱对证券公司风险进行监测、预警。利用爬虫系统爬取全网新闻、风险事件并将事件输入到训练好的深度学习模型中，并关联公司主体图谱与风险事件图谱，预测公司主体的健康度，进而监测、预警证券公司的风险程度。

基于深度学习与知识图谱的前沿理论与方法，通过深入研究事件驱动对公司自营以及融资融券、股票质押等业务的风险传导机制以及国内外股票市场的运行规律，构建科学合理、操作性强的适合券商内部风险监测预警机制，提升公司的风险管理水平，促进公司业务的健康、快速发展。

关键词：金融科技；知识图谱；深度学习；自然语言处理；风险监测预警

# 目录

[第一章 前言 1](#_Toc523227659)

[第二章 系统性金融风险 3](#_Toc523227660)

[一、系统性金融风险 3](#_Toc523227661)

[二、系统重要性金融机构 3](#_Toc523227662)

[三、系统重要性金融机构的识别 4](#_Toc523227663)

[第三章 深度学习与知识图谱 5](#_Toc523227664)

[一、深度学习 5](#_Toc523227665)

[二、知识图谱 6](#_Toc523227666)

[第四章 知识图谱与深度学习的应用现状 20](#_Toc523227667)

[一、国外开放知识图谱 20](#_Toc523227668)

[二、中文开放知识图谱联盟介绍 23](#_Toc523227669)

[三、人工智能、知识图谱在金融行业中的应用 24](#_Toc523227670)

[第五章 证券公司风险监测预警机制研究 30](#_Toc523227675)

[一、证券公司自营持仓主体以及交易对手图谱构建 31](#_Toc523227676)

[二、风险事件图谱构建 41](#_Toc523227677)

[三、公司健康度判别模型 55](#_Toc523227678)

[第六章 证券行业系统性风险监测预警的建议 58](#_Toc523227679)

[一、建立系统重要性证券公司的监测预警体系 58](#_Toc523227680)

[二、建立多方合作的风险监测预警体系 58](#_Toc523227681)

[三、制定系统重要性证券公司评判标准 58](#_Toc523227682)

[四、建立健全监管科技体系 58](#_Toc523227683)

[参考文献 60](#_Toc523227684)

# 第一章 前言

金融是现代经济运行的血脉和资源配置的枢纽，是国家重要的核心竞争力，金融安全是国家安全的重要组成部分。十九大报告中明确指出要“健全金融监管体系，守住不发生系统性金融风险的底线”。在刚结束的中央经济工作会议上，中央将防范化解重大风险列为今后3年决胜全面建成小康社会的三大攻坚战之首，并强调“重点是防控金融风险”，金融必须回归本源，支持服务实体经济的宗旨，强化金融监管，严控金融风险，确保不发生系统性金融风险。2018中国金融科技产业峰会的主要议题为“利用金融科技，强化系统性金融分析的防范能力”，深入探讨了大数据、人工智能、云计算等新兴技术在金融风险防控方面的关键技术和广泛应用场景。防范系统性金融风险，金融科技应大有可为[1]。

IMF（2009）认为系统性金融风险是由于金融系统中的部分或者全部机构遭遇经营危机时导致的整个金融市场的混乱的风险。防范系统性金融风险的重要手段就是要防范系统重要性金融机构的风险。我国系统性重要金融机构应包括规模较大的银行、保险公司、证券公司等金融机构，因此，建立高效、实用的证券公司风险的监测、预警体系，对证券公司健康、高效、稳定的运行有非常重要的意义，同时对防范系统性风险也是必不可少的关键环节[2]。

金融科技（Financial Technology）融合了金融和科技两方面的内容正在加速迭代演进风靡全球。金融科技最早是国外的大型互联网公司以及创业公司将信息技术用于非银行支付交易的流程改造和风险监测，后来在其他前沿的IT技术领域与金融领域得到快速发展。国际金融理事会（FSB）将金融科技定义为技术带动的金融创新，即金融供应商运用互联网、大数据、云计算、区块链、人工智能、生物科技等各种技术创造与优化业务模式、应用流程及产品和服务[3]。

依托科技的进步与网络的普及，金融科技在国内的也得到了迅猛的发展。中国的金融科技已不仅仅局限于电子支付、互联网信贷等形式，其服务范围与形式进一步拓展并带来新的发展趋势。传统金融机构积极拥抱金融科技并日益融合发展。传统金融机构面临新兴互联网企业跨界开展业务和自身转型升级的压力，深入推进金融科技发展战略，调整组织架构，在互联网消费金融、移动支付、互联网保险、直销银行、智能客服、风险管理、精准营销等领域实现重点突破。金融科技通过金融业务与信息技术的深度融合，能够有效降低金融交易成本，提升服务效率，实现金融功能的优化与服务边界的拓展。金融科技的两大核心要素是数据和技术，借助这两者的驱动，智能性、便捷性、低成本成为金融科技变革传统金融的切入点[4]。

在风险防范领域，传统模式下事后的、手动的、基于传统机构性数据的风险防范模式已经不能满足金融科技新业态的需求，金融机构也应与时俱进的采用大数据、人工智能等新兴技术进行风险管控，减少人力支出，完成风险的主动识别与控制，有效降低风控合规成本，增强风控合规能力。

本课题基于知识图谱与深度学习事件驱动的证券公司风险监测预警机制展开研究。基于深度学习与知识图谱的前沿理论与方法，通过深入研究事件驱动对公司自营以及融资融券、股票质押等业务的风险传导机制以及国内外股票市场的运行规律，构建科学合理、操作性强的适合券商内部风险监测预警机制，提升公司的风险管理水平，促进公司业务的健康、快速发展，从而有效的防范系统性金融风险的发生。

# 第二章 系统性金融风险

## 一、系统性金融风险

系统性金融风险一词很早就被提出，但由于其涉及的面比较广,当前学术界对其尚无统一的定义。事实上，系统性金融风险通常反映为单个或少数几个金融机构的破产或巨额损失导致整个金融系统崩溃的风险以及对实体经济产生严重负面效应。IMF(2009)认为系统性金融风险是指由于金融系统中的部分或者全部机构遭遇经营危机时导致的整个金融市场混乱的风险，这种混乱很可能对实体经济产生严重负面效应。我们对系统性金融风险的研究多关注于金融机构、金融市场、金融基础设施与更广义的经济体之间的相互作用。

2007年全面爆发的国际金融危机彻底暴露了国际监管组织在系统重要性金融机构监管上的不足，一系列规模大、全球活跃度较高的金融机构相继发生危机，对整个金融体系产生巨大冲击的同时，也对实体经济造成了极大的损害。在这场金融危机中，雷曼兄弟申请破产保护，贝尔斯登、美林公司分别被摩根大通和美国银行收购，房利美、房地美和美国国际集团相继被美国财政部接管。在这场浩大的金融海啸中，这些赫赫有名的大型金融机构非但没有起到“金融稳定器”的作用，反而成为了系统性金融风险的发源地和集散地。与此同时，监管当局缺少用于防止风险进一步扩散并稳定市场的监管工具。

## 二、系统重要性金融机构

防范系统性金融风险的重要手段就是制定一套完整的政策框架来解决系统重要性金融机构（Systemically Important Financial Institutions，简称 SIFIs）的风险防范问题。

系统重要性金融机构在国际货币基金组织（IMF）、国际清算银行（BIS）和金融稳定理事会（FSB）联合发布的《系统重要性金融机构、市场和工具的评估指引》中定义为在金融市场中承担了关键功能，倒闭可能会给金融体系造成损害并对实体经济产生严重负面影响的金融机构。2010年金融稳定理事会（FSB）首次将一些自身业务规模较大、复杂性和系统关联性较强等原因，如果经营失败一旦倒闭将会给整个金融机构乃至实体经济带来显著破坏的金融机构，定义为“系统重要性金融机构（SIFIs）”。金融稳定理事会与巴塞尔银行监督委员会、各国监管机构确定了多项标准对系统重要性金融机构进行识别。

## 三、系统重要性金融机构的识别

系统重要性金融机构的识别方法包括指标法和市场法两种。指标法是指根据的核心特征确定相应的指标，再根据金融体系的实际运行情况为各项指标确定不同的分值，并以此确定的具体名单。而市场法则通过运用数据的整理和分析对比，考察金融机构相关指标的市场波动情况，衡量单个金融机构对整个金融体系的风险贡献程度，进而确定该金融机构是否具有系统重要性。

巴塞尔委员会公布了2017年的全球系统重要性银行名单，其中国内的五大国有银行全部入围，全国性股份制商业银行除少数几家外也悉数入围。但是，国内仍缺少针对证券公司制定的系统重要性金融机构的评判标准。由于大型证券公司业务模式多元化、业务数据复杂化等特点，对这些证券公司风险状况监测预警在防范系统性风险方面显得尤为重要。

我们可以实时获取与系统重要性证券公司自营以及融资融券、股票质押等业务相关的股票持仓主体以及关联方相关的新闻、舆情信息等全网信息，并通过自然语言处理技术（NLP）对全网信息进行处理。知识图谱结合深度学习可利用处理过的全网信息对持仓主体以及关联方的风险程度进行累加进而实现对证券公司的健康度进行监测预警，进而达到防范系统性金融风险的目的。

# 第三章 深度学习与知识图谱

## 一、深度学习

深度学习的概念由Hinton等人于2006年提出，其前身为神经网络。2016年DeepMind公司所开发的AphaGo击败围棋世界冠军李世石之后，深度学习这一概念逐渐被人认可。通过深度学习技术，可有效地处理机器翻译、语音识别、图像分类以及自然语言处理等机器学习的问题。深度学习的基本原理是通过神经网络结构进行低维到高维的映射，从而能够对数据的特征进行提取，最后能够得到数据的特征表示[5]。典型的深度学习模型见图3-1。

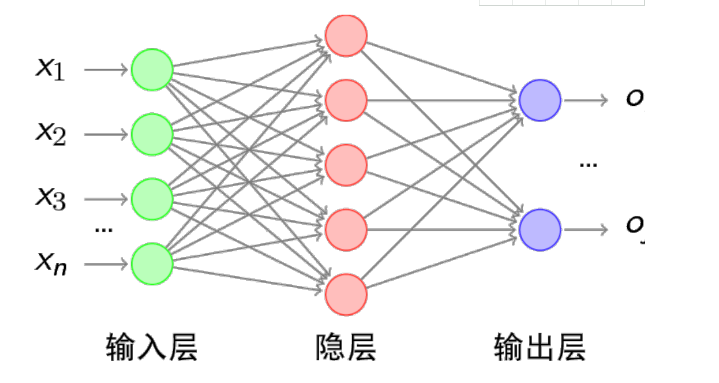


图3-1 深度学习模型示例图

深度学习本质上是构建含有多隐层的机器学习架构模型，通过大规模数据进行训练，得到大量更具代表性的特征信息。从而对样本进行分类和预测，提高分类和预测的精度。这个过程是通过深度学习模型的手段达到特征学习的目的。深度学习模型和传统浅层学习模型的区别在于：（1）深度学习模型结构含有更多的层次，包含隐层节点的层数通常在5层以上，有时甚至包含多达10层以上的隐藏节点；（2）明确强调了特征学习对于深度模型的重要性，即通过逐层特征提取，将数据样本在原空间的特征变换到一个新的特征空间来表示初始数据，这使得分类或预测问题更加容易实现。和人工设计的特征提取方法相比，利用深度模型学习得到的数据特征对大数据的丰富内在信息更有代表性。

深度学习算法打破了传统神经网络对层数的限制，可根据设计者需要选择网络层数。它的训练方法与传统的神经网络相比有很大区别，传统神经网络随机设定参数初始值，采用BP算法利用梯度下降算法训练网络，直至收敛。但深度结构训练很困难，传统对浅层有效的方法对于深度结构并无太大作用，随机初始化权值极易使目标函数收敛到局部极小值，且由于层数较多，残差向前传播会丢失严重，导致梯度扩散，因此深度学习过程中采用贪婪无监督逐层训练方法。即在一个深度学习设计中，每层被分开对待并以一种贪婪方式进行训练，当前一层训练完后，新的一层将前一层的输出作为输入并编码以用于训练；最后每层参数训练完后，在整个网络中利用有监督学习进行参数微调。

深度神经网络的训练过程主要是先逐层预训练初始化深度神经网络每一层的参数，然后通过监督学习对整个深度结构神经网络进行微调。

训练过程如下：

* 1. 将第一层作为一个自编码模型将第一层作为一个自编码模型，采用无监督训练，使原始输入的重建误差最小；
  2. 将自编码模型的隐单元输出作为另一层的输入；
  3. 按步骤2迭代初始化每一层的参数；
  4. 采用最后一个隐层的输出作为输入施加于一个有监督的层（通常为输出层），并初始化该层的参数；
  5. 根据监督准则调整深度结构神经网络的所有参数。

## 二、知识图谱

“知识图谱（Knowledge Graph）”的概念由Google公司在2012年提出，是其用于提升搜索引擎性能的知识库。知识图谱的出现是人工智能对知识需求所导致的必然结果，但其发展又得益于很多其他的研究领域，涉及专家系统、语言学、语义网、数据库，以及信息抽取等众多领域，是交叉融合的产物而非一脉相承[6]。

知识图谱本质上是语义网络，是一种基于图的数据结构，由节点和边组成。在知识图谱里，每个节点表示现实世界中存在的“实体”，每条边为实体与实体之间的“关系”。知识图谱是关系的最有效的表示方式。知识图谱就是把所有不同类型的信息连接在一起而得到的一个关系网络，他提供了从“关系”的角度去分析问题的能力[7]。

尽管知识图谱本质上来源于语义网，然而当前环境下的知识图谱却有着新的特点——海量规模，这也是为什么知识图谱技术能够展现出巨大潜力和活力的原因所在。传统语义知识图谱的构建是以任务驱动型为导向的自顶向下方式驱动的，是依赖领域专家人工的对领域概念和术语体系进行建模，如CYC、WordNet等。因此数据规模非常有限，WordNet只包含大约71万个三元组，Cyc包含大约160万个三元组。然而，在当前数据爆炸的时代，随着知识抽取技术的不断完善和数据来源(文本、图像、传感器、视频等)的不断丰富，知识图谱的规模正在呈爆炸式增长。以开放关联数据(Linked Open Data,即LOD)项目为例，2009年语义Web中包含RDF三元组数目约为44亿条，2010年该数目增长至130亿条，而2011年9月统计的RDF三元组总数已超过310亿条，截止到2016年11月，LOD项目上发布了包含总共超过1300亿的RDF三元组[8]。当前知识图谱已从传统的小规模本体性知识演化为大规模海量的事实性知识。从某种程度上来说，知识图谱已经等同于海量知识图谱的概念。

知识图谱主要有自顶向下(top-down)与自底向上(bottom-up)两种构建方式。自顶向下指的是先为知识图谱定义好本体与数据模式，再将实体加入到知识库。该构建方式需要利用一些现有的结构化知识库作为其基础知识库，例如Freebase项目就是采用这种方式，它的绝大部分数据是从维基百科中得到的。自底向上指的是从一些开放链接数据中提取出实体，选择其中置信度较高的加入到知识库，再构建顶层的本体模式。目前，大多数知识图谱都采用自底向上的方式进行构建，其中最典型就是Google的Knowledge Vault。

知识图谱模型具备以下几个显著特点：

1. 概念性，将知识库的实体进行概念分解，利用解析实体内涵却无外延的概念抽象模型，降低歧义度；
2. 共享性，将模型中的每一概念进行数据编号，满足查询用户或相关应用领域的知识组织形式，具有人机共识的特性；
3. 形式性，建模中概念及关系的抽取采用数学表达式形式，具有完整一致的特性；
4. 明确性，以知识库为根据，明确定义了实体的概念，关系以及与相关的其他部分。

知识图谱的构建流程见图3-2。

实体关系抽取

构建知识图谱

命名实体识别

爬虫

图3-2 知识图谱构建流程图

1. 命名实体识别

命名实体识别（Named entity recognition，NER）是指通过自动化手段从非结构化文本中识别具有特定含义的一类实体概念的技术，也可以称作实体抽取。命名实体识别的准确性直接决定了关系识别的质量，影响到知识图谱构建的后续环节，因此是是构建知识图谱的一个首要任务[9]。

命名实体识别的任务被定义为识别出文本中出现的专有名称和有意义的数量短语并加以归类。命名实体是文本中基本的信息元素，是正确理解文本的基础。狭义地讲，命名实体是指现实世界中的具体的或抽象的实体，如人、地点、机构等，通常用唯一的标志符(专有名称)表示，如人名、地名、机构名等。广义地讲，命名实体还可以包含时间、数量表达式等。至于命名实体的确切含义，只能根据具体应用来确定。比如，在具体应用中可能需要把住址、电子信箱地址、电话号码、会议名称等作为命名实体。

利用统计学习方法，或者说机器学习方法来解决命名实体识别这样的自然语言处理问题，多年来已经有了较多的解决方案[10]。其中条件随机场模型（CRF）等监督式学习方法虽然已经有了较为稳定的效果，但是这类方法通常需要配合繁琐的人工特征处理过程，如CRF需要人工编辑特征模板生成特征函数集合，利用这些特征函数去学习语料中的“知识”。而本项目使用的解决方案则是近几年来较为流行的神经网络学习模型或者说深度学习模型。该方案相比较于传统的机器学习方法改进之处在于能够通过复杂的神经网络结构来学习数据的特征，并且可以由低层次至高层次自动地学习特征的层次结构，省去了需要人工介入的特征提取工程，并且可以获得更优秀的效果。

本课题所采用的命名实体识别方案整体模型结构见图3-3。

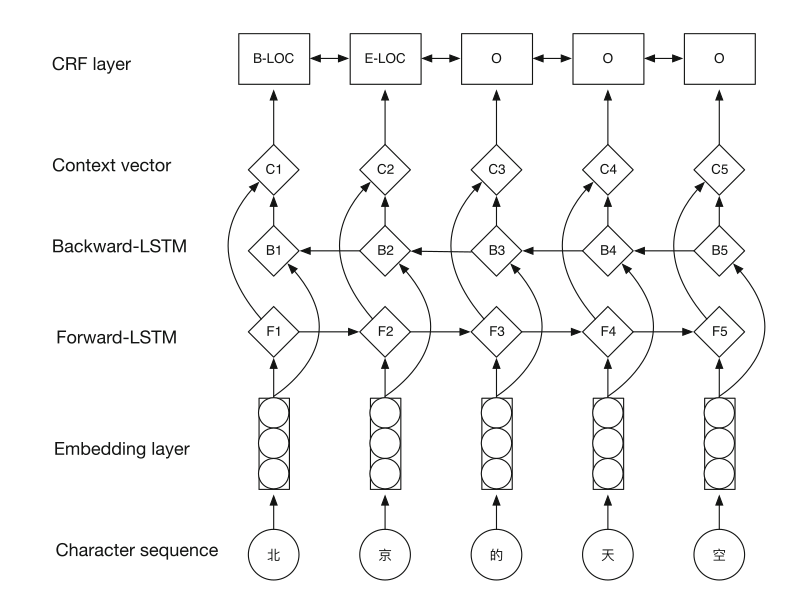


图3-3 总体模型结构

图中圆圈节点为模型输入层，也即word embedding部分，菱形节点代表长短时记忆模型的双向网络结构，将正向（左侧）学习结果与逆向（右侧）学习结果组合为输入单词的整体上下文表达，并将该结果作为条件随机场模型的输入，正方形代表 CRF 模型中的随机变量[11]。该模型要训练的参数包括条件随机场模型中的序列转移矩阵以及双向长短时记忆模型中的参数，每一层网络节点与节点的线性权值。

命名实体识别的主要技术包括语言文本表示、特征学习模型、序列标注模型、条件随机场模型等，具体技术如下：

1. 语言文本表示

利用机器学习方法处理文本数据的首要任务是对输入的文本数据进行建模，即数学符号化的自然语言表示。对于自然语言来说，如何将字、词、句子等基本元素组成的文本转换为抽象的数学模型是一个较为棘手的问题。

为了增强词向量的表达能力，使之更加符合自然语言的实际意义，可以结合文档空间中所有单词的出现频次、词的共现以及单词的搭配信息来“学习”语言中的“逻辑”。这里的逻辑也即在文档空间中，什么样的单词搭配及组合是最常出现、最为合理的。

在本项目中采用的便是“字嵌入”（Word-Embedding）的词向量方案，利用无监督的学习算法，从大量的未标注文本语料中学习出向量，这时的词向量通常维数在几十至上百，并且与传统的词向量方案不同，该向量通常都是稠密的向量，虽然无法合理地解释每一维度的实际含义，但是其表达能力相较传统词向量方案要提升许多[12]。

Word2vec是Mikolov在 2013 年提出的用于快速有效地训练词向量的模型。作者的目标是要从海量的文档数据中学习高质量的词向量，该词向量在语义和句法上都有很好地表现，已经广泛应用于自然语言处理的各种任务中[12]。Word2vec包含了两种训练模型，分别是CBOW和Skip\_gram模型，见图 3-4。

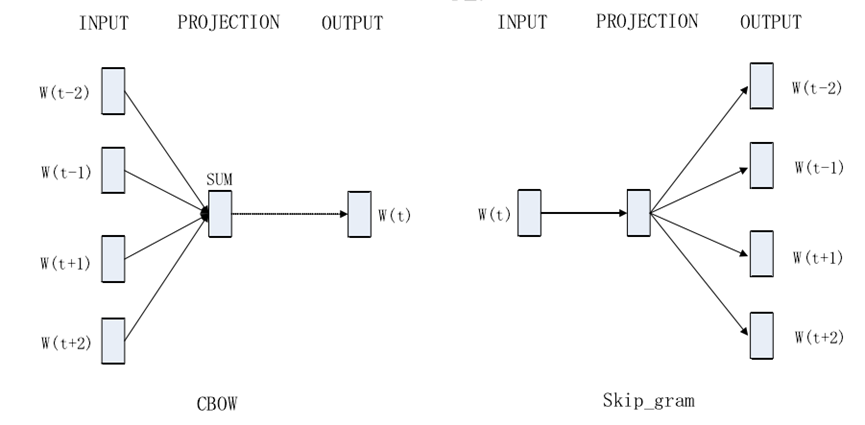


图3-4 CBOW和skip gram模型图

可以看出 CBOW 模型和 Skip\_gram 模型均包含输入层、投影层和输出层。其中 CBOW 模型利用上下文预测当前词，而Skip\_gram模型利用当前词预测其上下文。Word2vec模型还有两套优化方法来提高词向量的训练效率，分别是Hierarchical Softmax和Negative Sampling。将训练模型与优化模型进行组合可以得到 4 种训练词向量的框架。

1. 特征学习模型

深度学习也即深层神经网络，是一种特征学习的方法，最早于 1943 年被沃伦·麦卡洛克提出，具备多层神经网络结构。近年来在图像处理领域获得了重要的突破和成果。而在自然语言处理领域，由于涉及到对语言模型复杂结构以及序列的处理，因此需要特殊的神经网络结构。在本项目中，使用的特征模型是长短时记忆神经网络模型（Long Short-Term Memory）[13]。LSTM是在循环神经网络（Recurrent Neural Network）的基础上改进而来的模型，而循环神经网络对于传统神经网络模型的改进之处在于其对于时间序列输入的处理。

传统的神经网络结构在处理相互独立的输入数据上有着较好的效果，但是其不足之处在于无法处理具有时间序列关系的数据。本项目涉及的自然语言便是这样的场景。在一段语言文本中，上下文之间的联系通常对于局部文本的理解具有相当重要的作用。把一句话作为输入序列，不同的序列项之间总是存在着某些联系而不是相互独立的，这是人类语言的天然特质。

传统神经网络的结构认为输出结果只与输入有关，而循环神经网络的改进之处在于在网络内部隐藏层设置了隐藏节点与自身上一时刻的状态有关，也即输出结果不仅与当前输入有关，而且与上一时刻的输出有关。

本项目使用的长短时依赖循环神经网络（Long Short-Term Memory Networks）模型通过改变循环神经网络的神经元结构来解决长距离依赖的信息丢失问题。在LSTMs的结构中，设计了一种称为记忆细胞（Memory-Cell）的结构，在该结构中，设计了几个门结构（gate）来控制细胞的输入比例，包括当前输入数据以及上一时刻输出数据以及细胞的上一时刻状态值，通过这种结构来丢弃或者保留网络中传输的参数也即记忆，从而达到保留长期信息依赖的效果[14]。一个典型的 LSTM 神经元结构见图3-5。

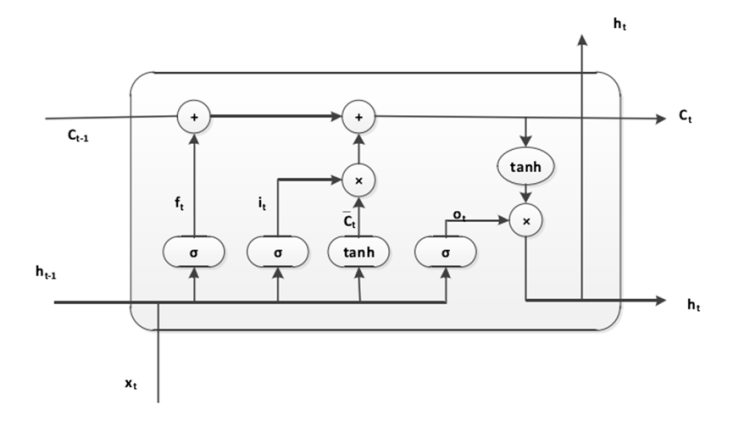


图3-5 LSTM神经元结构图

在本项目的应用场景中，使用的特征学习模型是双向的长短时记忆网络（Bi-LSTMs），这是由于自然语言中一段文本通常隐含了上下文的信息，也就是说在处理文本时不仅需要考虑当前文本之前的依赖信息同时也需要考虑在后文中的依赖信息。给定一个包含 n 个单词的句子输入(x1，x2，x3，x4……，xn)，其中每个单词为一个多维向量（也即前文介绍的词向量），网络将会首先计算针对每一个单词 t的左侧输出 htl，接下来计算代表该单词右侧的输出 htr，其方法是将输入序列逆转后再进行一次相同的计算并输出，这样的结构便构成了双向的 LSTM 网络结构，网络中不同方向的网络其训练出来的参数是不相同的。最后，结合两个方向的计算输出便可以很好地代表在一个文章中的每一个单词。

1. 序列标注模型

从本质上来说，命名实体识别问题可以被看作是一个序列标注问题，即对于给定的输入序列，在一个有限的标注集合内为其标注标签并输出。要识别一段文本中的命名实体，则是在一个大规模的语言模型中标记输入文本中的那些序列片段可以作为命名实体的组成部分而哪些片段不可以作为组成部分。前述的步骤已经可以用序列向量将文本表达出来并且利用特征学习模型很好地表达了一段输入序列的特征，而接下来的任务便是对其进行序列标注。

带标注的原始数据：此次{{location:中国}}个展，{{person\_name:苏珊·菲利普斯}}将与她80多岁高龄的父亲一起合作，哼唱一首古老的{{location:威尔士}}民歌{{product\_name:《白蜡林》}}。届时在{{location:画廊大厅}}中将安放6个音箱进行播放，艺术家还特意回到家乡{{location:格拉斯哥}}，同父亲一起在{{org\_name:中国音乐学院}}里为作品录制了具有{{location:中国}}元素的音乐片段[15]。

一个十分简单却有效的模型便是直接使用上述特征学习模型的输出ht来作为特征直接进行标注，那么这样的标注模型即是认为所有输入序列之间相互独立的。这种方案在词性标注问题（Part-of-Speech Tagging）中得到了很好的效果，但是词性标注问题的输出标签之间并不存在很严格的语法关系，也就是说利用已经学习到的语言特征完全可以应对“相互联系松散”的词性标注。例如，在词性标注问题中名词一般出现在动词之后，这是在词性标注领域所存在的模型逻辑或者说模型特征，而这些特征层次较低，一般我们在进行前述特征学习阶段已经可以获得。

而对于命名实体识别而言，在该层次上仍然存在着严格的限制特征，被标记为命名实体其开头、中间及末尾都有着严格的顺序限制[16]。因而，在设计标注方案时必须要考虑如何进一步学习命名实体在语法上的严格顺序特征，也即将输入文本看作是互连联系不独立的序列为其建立概率模型，在本课题中使用条件随机场（Conditonal Radom Field）模型来对序列进行建模和标注。表3-1是本项目所采用的命名实体识别标注方案标签集合。

表3-1 命名实体识别标注方案标签集合



根据标注方案标签集合将数据标注后的结果如下：

此/O次/O 中/B\_location 国/E\_location 个/O 展/O

哼/O 唱/O 一/O 首/O 古/O 老/O 的/O 威/B\_location 尔/M\_location 士/E\_location 民/O 歌/O 同/O 父/O 亲/O 一/O 起/O 在/O 中/B\_org\_name 国/M\_org\_name 音/M\_org\_name 乐/M\_org\_name 学/M\_org\_name 院/E\_org\_name 里/O 为/O 作/O 品/O 录/O 制/O 了/O 具/O 有/O 中/B\_location 国/E\_location 元/O 素/O 的/O 音/O 乐/O 片/O 段/O

1. 条件随机场模型

CRF又称为马尔可夫随机域，是一种用于标注和切分有序数据的条件概率模型，集合了最大嫡模型和HMM模型的特点，从形式上来说CRF可以看做是一种无向图模型，考察给定输入序列的标注序列的条件概率。

假设x是所要标注的观察序列上的随机变量，y是相应的标注序列的随机变量，且所有y的组成部分包括在一个固定的符号集L中。例如x可能是一句话，而y是对应于这个句子的标注序列，L就是所有可能的标注符号集。

给定一个无向图G=(V,E)，其中V是图的顶点集合,E是边集合。y上的每个元素相对应的随机变量yv都在图中对应一个节点。如果每个基于x的随机变量yv在图G上都服从马尔可夫特性,即

其中*w*～*v*表示两个顶点之间有连接边,即表示*w*和*v*在图G中位置相邻,我们就称符合上述条件的(x,y)为一个条件随机域,CRF是基于观察序列x的全局随机域。

如果在需要模型化的标注序列上做出马尔可夫独立性假设,那么图G的结构可以是任意的。通常,线性链结构是最简单也是最常用的CRF图结构。在对序列标注问题建模时,图G中的标注序列就会形成一条马尔可夫链，见图3-6。

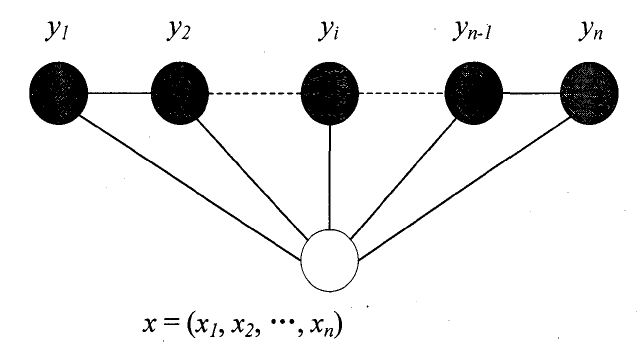


图3-6 马尔可夫链结构

CRF可以看作是具有规格化转换概论的有限状态模型。它允许在观察序列上的任意的依赖关系，并且特征不需要一定是一个完整的状态或者观察值。CRF模型和最大熵模型基于相同的指数形式，因此它可以进行有效的、完备的、非贪婪的有限状态推导和训练。所以，可以尝试用较少的训练数据来训练出模型。另外，CRF具有损失函数的凸面性，所以说CRF拥有了一般的最大嫡模型的所有优点。CRF的参数估计过程实际上就是求取模型参数的过程,训练时利用训练数据库，通过极大似然估计得到条件概率模型，然后使用该模型预测。

CRF学习的过程就是描述一些特征配置，比如当前词语为xx，上个词为xx，满足这种配置的，特征函数输出就是1，不满足就是0。每个词都有同样多的特征函数判断。预测过程就是利用每种特征配置给标签打分，然后打分结果加权求和，打分最高的标签就是预测结果。

1. 实体关系抽取

实体关系抽取是NLP领域最重要的研究方向之一。其相关工作不仅包含监督模式下的分类学习，也涉及无监督学习的关系发现。其中，大量的工作主要是围绕如何设计出更好的能够描述实体之间句子的特征来识别这些关系。

而在多数情况下，关系抽取被看作一个多分类的监督问题。这些方法大致可分为两类：基于特征的方法以及核方法。在基于特征的方法中，人们利用不同的方法将句子序列特征、句法树特征转化成用于分类的特征向量。然而，这类方法在将结构化句子表达形式转化为特征向量时，往往会遇到特征选择的问题。核方法提供了另一种自然的替代方案来使用这些结构化的句子表达形式，它不需要进行明确的特征选择操作。常见的核方法有卷积树核、子序列核和依存树核等。这些方法面临的问题是缺少足够的标注数据用来训练。Mintz等人在2009年提出的远监督方法解决了训练数据不足的问题。

监督学习方法被证明可以有效解决关系抽取问题，且己经取得了不错的分类效果。但是传统的监督学习方法严重依赖特征的质量。随着近年来表示学习和深度学习技术的发展，大量深度学习的技术在NLP中得以应用，并且己经取得了许多非常惊人的效果，研究者们发现深度学习非常适合进行自动的特征学习。在NLP领域，深度学习一开始主要用于词的表示学习，例如词向量等。后来，Socher等人将递归神经网络和语法树结合起来用于关系分类任务，取得了很好的效果。正如前面提到的，利用传统的NLP工具手动设计高质量的特征是非常困难的。Zeng等人首次在不借助其他NLP工具的情况下，将卷积神经网络应用到了关系分类上，并取得了理想的效果，分类效果甚至超过了最好的基于特征工程的方法，可见深度学习在关系分类任务上有着显著的应用前景。

为了对资本市场企业和人物建立关系图谱模型，获取实体之后如何获得实体之间的关系是更为重要的一步。在资本市场的实体关系与开放领域的知识图谱实体关系存在较大差异，其中最显著的一点便是关系类型的定义。在开放领域中实体本身就包含多种类别，除了物理实体还包含抽象概念实体，在此基础上的实体关系类型更是无法准确而全面地做出定义。

1. 实体关系类型定义

根据资本市场领域特点，关系属性表示该种关系类型的效用范围。例如法定关系，则代表该种关系类型是在证券行业相关法律规定范围内的关系，而社会关系的效用范围不仅仅局限于资本市场领域，更是一种社会范围内有效的关系类型；市场关系目前主要指企业行业内部的供应链上下游关系，是由于企业经营行为而产生的非法定关系。

1. 关系提取方案

本项目中涉及的关系类型大部分可以从权威的结构化数据中直接提取，例如法定关系的企业投资关系和股东关系，作为资本市场的上市公司必须在指定平台上披露公司的资本构成情况，因此从公司自有的结构化数据库中便可以提取所有上市公司的法定资本构成关系，包括公司股东、公司对外投资、分支机构以及公司的管理层人事信息（包括董事会成员、监事会成员及其高级管理人员情况）。针对市场关系，从公司披露的年报及相关公告中可以获取公司的产业链上下游信息，这些信息基本上以半结构化数据的方式获取，利用信息抽取技术挖掘出其中的结构化信息并存储[16]。

对于人与人之间的社会关系，主要提取方式是从人物在资本市场的相关履历中获得人物的个人经历，通常在文档中以半结构化数据方式展示，通过基本的正则匹配等信息抽取技术可以获取该人物的教育经历、工作经历等信息，将该信息结构化存储后通过简单的结构化查询语句便可以获取人物之间的校友、同事（前同事）关系。由于实体间关联关系的确定在业务上是权威性较强的，因此在本项目目前的方案中只采用了较为基本的提取方案以获取权威性较强的关联关系[17]。

在实体关系识别中，将实体对作为分析的数据，一般选择字符特征、实体的类型特征、实体对的相对位置、上下文窗口特征等特征进行判别，有的研究还会将依存树路径特征纳入分析中。

目前大部分实体关系识别的研究都是在基于语句级的，即研究同一个句子中两个实体之间是否存在关系。而对于一句话中研究的是两个实体之间的关系还是两个以上实体之间的关系没有一个统一的规范。本项目采用语句级的粒度，研究一句话中任意两个实体之间是否存在联系。

本项目所提出算法的层次结构主要包括五个部分：输入层、词向量模型训练层、双向LSTM层、attention注意力机制层、关系分类输出层[18]。结构图见图3-7。

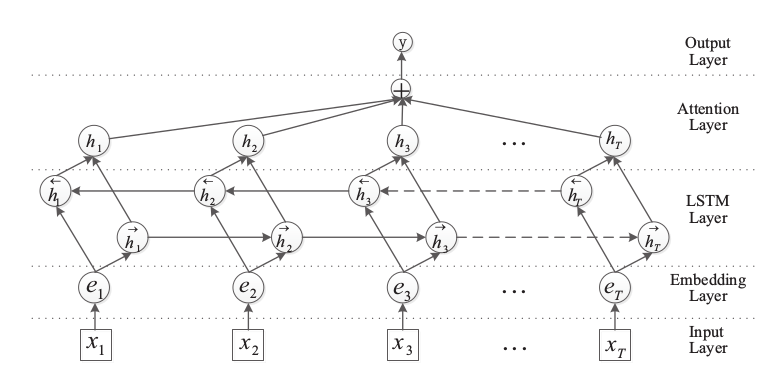


图3-7 BiLSTM+ATT模型结构图

其中的词向量训练模型与LSTM模型与命名实体识别中的相似，不再赘述。

在上述模型中Attention机制是非常重要的组成部分，要介绍Attention Mechanism结构和原理，首先需要介绍下Seq2Seq模型的结构。其主要结构见图3-8。

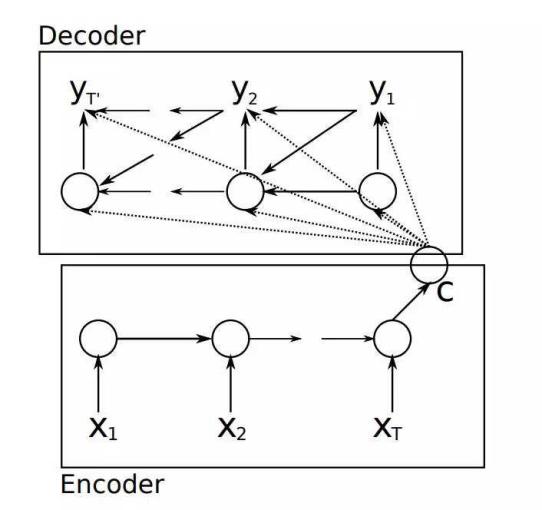


图3-8 Seq2Seq模型结构图

其中，Encoder把一个编程的输入序列X1，X2，X3....Xt编码成一个固定长度的隐向量（背景向量，或上下文向量context）c，c有两个作用：一是作为初始向量初始化Decoder的模型，作为Decoder模型预测y1的初始向量；二是作为背景向量，指导y序列中每一个step的y的产出。Decoder主要基于背景向量c和上一步的输出yt-1解码得到该时刻t的输出yt，直到碰到结束标志（<EOS>）为止。

如上文所述，传统的Seq2Seq模型对输入序列X缺乏区分度。因此引入了Attention Mechanism来解决这个问题，他们提出的模型结构见图3-9。

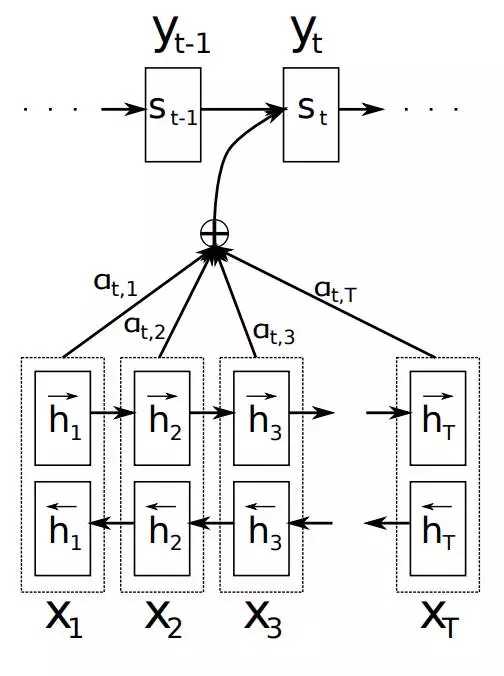


图3-9 Attention Mechanism结构图

在该模型中，定义了一个条件概率

其中，是Decoder中RNN在在i时刻的隐状态，其计算公式为

这里的背景向量的计算方式，与传统的Seq2Seq模型直接累加的计算方式不一样，这里的是一个权重化（Weighted）之后的值

其中，i表示Encoder端的第i个词，表示Encoder端的第j和词的隐向量，表示Encoder端的第j个词与Decoder端的第i个词之间的权值，表示源端第j个词对目标端第i个词的影响程度

是一个softmax模型输出，概率值的和为1。表示一个对齐模型，用于衡量Encoder端的位置j个词，对于Decoder端的位置i个词的对齐程度（影响程度），换句话说：Decoder端生成位置i的词时，有多少程度受Encoder端的位置j的词影响。对齐模型的计算方式有很多种，不同的计算方式，代表不同的Attention模型，最简单且最常用的的对齐模型是dot product乘积矩阵，即把target端的输出隐状态ht与source端的输出隐状态进行矩阵乘。常见的对齐计算方式如下：

其中, = 表示源端与目标单词对齐程度。可见，常见的对齐关系计算方式有，点乘（Dot product），权值网络映射（General）和concat映射几种方式。

# 第四章 知识图谱与深度学习的应用现状

## 一、国外开放知识图谱

DBpedia 是一个大规模的多语言百科知识图谱，可视为是维基百科的结构化版本。DBpedia 使用固定的模式对维基百科中的实体信息进行抽取，包括 abstract、infobox、category 和 page link 等信息。图 4-1 示例了如何将维基百科中的实体“Busan”的infobox 信息转换成RDF三元组。DBpedia目前拥有127种语言的超过两千八百万个实体与数亿个RDF三元组，并且作为链接数据的核心，与其他的数据集都存在实体映射关系。而根据抽样评测，DBpedia中 RDF 三元组的正确率达88%。DBpedia支持数据集的完全下载。

Yago 是一个整合了维基百科与WordNet的大规模本体，它首先制定一些固定的规则对维基百科中每个实体的infobox进行抽取，然后利用维基百科的category进行实体类别推断（Type Inference）获得了大量的实体与概念之间的 Is A 关系（如：“Elvis Presley” Is A “American Rock Singers”），最后将维基百科的 category 与WordNet中的 Synset（一个 Synset 表示一个概念）进行映射，从而利用了WordNet 严格定义的Taxonomy完成大规模本体的构建。随着时间的推移，Yago的开发人员为该本体中的 RDF 三元组增加了时间与空间信息，从而完成了 Yago2的构建，又利用相同的方法对不同语言维基百科的进行抽取，完成了 Yago3的构建。目前，Yago 拥有 10 种语言约 459 万个实体，2400 万个 Facts，Yago 中 Facts的正确率约为 95%。Yago 支持数据集的完全下载。



图4-1 RDF三元组转换图

Wikidata是一个可以自由协作编辑的多语言百科知识库，它由维基媒体基金会发起，期望将维基百科、维基文库、维基导游等项目中结构化知识进行抽取、存储、关联。Wikidata 中的每个实体存在多个不同语言的标签，别名，描述，以及声明（statement），比如 Wikidata 会给出实体“London”的中文标签“伦敦”，中文描述“英国首都”以及图 4-2 给出了一个关于“London”的声明的具体例子。“London”的一个声明由一个 claim 与一个 reference 组成，claim 包括property:“Population”、value:“8173900”以及一些 qualifiers（备注说明）组成，而 reference 则表示一个 claim 的出处，可以为空值。目前 Wikidata 目前支持超过 350 种语言，拥有近 2500 万个实体及超过 7000 万的声明，并且目前 Freebase 正在往 Wikidata 上进行迁移以进一步支持 Google 的语义搜索。Wikidata 支持数据集的完全下载。

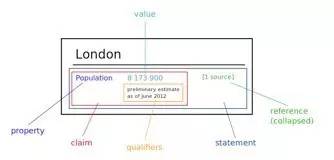


图4-2 实体属性示例图

BabelNet 是目前世界范围内最大的多语言百科同义词典，它本身可被视为一个由概念、实体、关系构成的语义网络（Semantic Network）。BabelNet 目前有超过 1400 万个词目，每个词目对应一个 synset。每个 synset 包含所有表达相同含义的不同语言的同义词。比如：“中国” “中华人民共和国” “China”以及“people’srepublic of China”均存在于一个 synset 中。BabelNet 由 WordNet 中的英文 synsets 与维基百科页面进行映射，再利用维基百科中的跨语言页面链接以及翻译系统，从而得到 BabelNet 的初始版本。目前 BabelNet 又整合了 Wikidata、GeoNames、OmegaWiki 等多种资源，共拥有 271 个语言版本。由于 BabelNet 中的错误来源主要在于维基百科与 WordNet 之间的映射，而映射目前的正确率大约在 91%。关于数据集的使用，BabelNet 目前支持 HTTP API 调用，而数据集的完全下载需要经过非商用的认证后才能完成。

ConceptNet 是一个大规模的多语言常识知识库，其本质为一个以自然语言的方式描述人类常识的大型语义网络。ConceptNet 起源于一个众包项目 Open Mind Common Sense，自 1999 年开始通过文本抽取、众包、融合现有知识库中的常识知识以及设计一些游戏从而不断获取常识知识。ConceptNet 中共拥有 36 种固定的关系，如 IsA、UsedFor、CapableOf 等，图 4-3 给出了一个具体的例子，从中可以更加清晰地了解 ConceptNet 的结构。ConceptNet 目前拥有 304 个语言的版本，共有超过 390 万个概念，2800 万个声明（statements，即语义网络中边的数量），正确率约为 81%。另外，ConceptNet 目前支持数据集的完全下载。

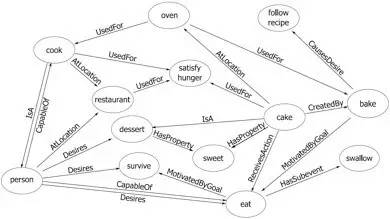


图4-3 ConceptNet结构示例图

Microsoft Concept Graph 是一个大规模的英文 Taxonomy，其中主要包含的是概念间以及实例（等同于上文中的实体）概念间的 IsA 关系，其中并不区分 instanceOf 与 subclassOf 关系。Microsoft Concept Graph 的前身是 Probase，它过自动化地抽取自数十亿网页与搜索引擎查询记录，其中每一个 IsA 关系均附带一个概率值，即该知识库中的每个 IsA 关系不是绝对的，而是存在一个成立的概率值以支持各种应用，如短文本理解、基于 taxonomy 的关键词搜索和万维网表格理解等。目前，Microsoft Concept Graph 拥有约 530 万个概念，1250 万个实例以及 8500 万个 IsA 关系（正确率约为 92.8%）。关于数据集的使用，MicrosoftConcept Graph 目前支持 HTTP API 调用，而数据集的完全下载需要经过非商用的认证后才能完成。

除了上述知识图谱外，中文目前可用的大规模开放知识图谱有 Zhishi.me、Zhishi.schema与XLore等。Zhishi.me 是第一份构建中文链接数据的工作，与 DBpedia 类似，Zhishi.me 首先指定固定的抽取规则对百度百科、互动百科和中文维基百科中的实体信息进行抽取，包括 abstract、infobox、category 等信息；然后对源自不同百科的实体进行对齐，从而完成数据集的链接。目前 Zhishi.me 中拥有约 1000 万个实体与一亿两千万个 RDF 三元组，所有数据可以通过在线 SPARQL Endpoint 查询得到。Zhishi.schema 是一个大规模的中文模式（Schema）知识库，其本质是一个语义网络，其中包含三种概念间的关系，即equal、related与subClassOf关系。Zhishi.schema抽取自社交站点的分类目录(Category Taxonomy)及标签云（Tag Cloud），目前拥有约40万的中文概念与150万RDF三元组，正确率约为84%，并支持数据集的完全下载。XLore 是一个大型的中英文知识图谱，它旨在从各种不同的中英文在线百科中抽取 RDF 三元组，并建立中英文实体间的跨语言链接。目前，XLore 大约有 66 万个概念，5 万个属性，1000 万的实体，所有数据可以通过在线 SPARQL Endpoint 查询得到。

## 二、中文开放知识图谱联盟介绍

中文开放知识图谱联盟（OpenKG）旨在推动中文知识图谱的开放与互联，推动知识图谱技术在中国的普及与应用，为中国人工智能的发展以及创新、创业做出贡献。联盟已经搭建有 OpenKG.CN 技术平台，见图 4-4 ，目前已有 35 家机构入驻。吸引了国内最著名知识图谱资源的加入，如 Zhishi.me， CN-DBPedia, PKUBase。并已经包含了来自于常识、医疗、金融、城市、出行等 15 个类目的开放知识图谱。



图4-4 中文开放知识图谱联盟

## 三、人工智能、知识图谱在金融行业中的应用

Google、IBM等国际巨头公司已经将人工智能技术渗透到各种产品，国内金融行业也将人工智能技术应用于多个场景。随着国内双创政策的推动和对人工智能产业的投资拉动，人工智能技术在金融行业的应用会越来越广泛[19]。国内一些互联网巨头以及金融机构已经在人工智能技术应用在金融行业领域发力，比如阿里巴巴旗下的蚂蚁金服设有专门的科学家团队，专门从事机器学习与深度学习等人工智能领域的前沿研究并在蚂蚁金融的业务场景下进行一系列的创新与应用；交通银行的智能网点机器人“交交”采用人工智能技术对客户进行业务办理指引；平安集团下设平安科技人工智能实验室，大规模研发人工智能金融应用。

现阶段，人工智能技术在金融行业的应用非常广泛。

1. 智能客服

智能客服主要是整合集团、公司对外的客户服务通道，提供多模式融合的在线智能客服，通过实时语音识别、实时语义理解，掌握客户需求，自动推送客户特征、知识库等内容。

通过电话客服渠道、网上客服、APP、短信、微信以及智能机器人终端与客户进行语音或文本的互动交流，理解客户业务需求，语音回复客户提出的业务咨询，并能根据客户语音导航至指定业务模块。对传统按键式菜单进行改造，用户使用自然语音与系统交互，实现菜单扁平化，提升用户满意度，减轻人工服务压力，降低运营成本。电话客服不再受限于菜单，可开展全业务的语音导航服务。

1. 反欺诈

反欺诈是风控中非常重要的一道环节。基于大数据的反欺诈的难点在于如何把不同来源的数据（结构化，非结构）整合在一起，并构建反欺诈引擎，从而有效地识别出欺诈案件（比如身份造假，团体欺诈，代办包装等）。而且不少欺诈案件会涉及到复杂的关系网络，这也给欺诈审核带来了新的挑战。 知识图谱，作为关系的直接表示方式，可以很好地解决这两个问题。 首先，知识图谱提供非常便捷的方式来添加新的数据源，这一点在前面提到过。其次，知识图谱本身就是用来表示关系的，这种直观的表示方法可以帮助我们更有效地分析复杂关系中存在的特定的潜在风险。

反欺诈的核心是人，首先需要把与借款人相关的所有的数据源打通，并构建包含多数据源的知识图谱，从而整合成为一台机器可以理解的结构化的知识。在这里，我们不仅可以整合借款人的基本信息（比如申请时填写的信息），还可以把借款人的消费记录、行为记录、网上的浏览记录等整合到整个知识图谱里，从而进行分析和预测。这里的一个难点是很多的数据都是从网络上获取的非结构化数据，需要利用机器学习、自然语言处理技术把这些数据变成结构化的数据。

1. 不一致性验证

不一致性验证可以用来判断一个借款人的欺诈风险，这个跟交叉验证类似。比如借款人张三和借款人李四填写的是同一个公司电话，但张三填写的公司和李四填写的公司完全不一样，这就成了一个风险点，需要审核人员格外的注意。

再比如，借款人说跟张三是朋友关系，跟李四是父子关系。当我们试图把借款人的信息添加到知识图谱里的时候，“一致性验证”引擎会触发。引擎首先会去读取张三和李四的关系，从而去验证这个“三角关系”是否正确。很显然，朋友的朋友不是父子关系，所以存在着明显的不一致性。

不一致性验证涉及到知识的推理。通俗地讲，知识的推理可以理解成“链接预测”，也就是从已有的关系图谱里推导出新的关系或链接。 比如在上面的例子，假设张三和李四是朋友关系，而且张三和借款人也是朋友关系，那我们可以推理出借款人和李四也是朋友关系。

1. 组团欺诈

相比虚假身份的识别，组团欺诈的挖掘难度更大。这种组织在非常复杂的关系网络里隐藏着，不容易被发现。当我们只有把其中隐含的关系网络梳理清楚，才有可能去分析并发现其中潜在的风险。知识图谱，作为天然的关系网络的分析工具，可以帮助我们更容易地去识别这种潜在的风险。举一个简单的例子，有些组团欺诈的成员会用虚假的身份去申请贷款，但部分信息是共享的。下面的图大概说明了这种情形。从图中可以看出张三、李四和王五之间没有直接的关系，但通过关系网络我们很容易看出这三者之间都共享着某一部分信息，这就让我们马上联想到欺诈风险。虽然组团欺诈的形式众多，但有一点值得肯定的是知识图谱一定会比其他任何的工具提供更佳便捷的分析手段。

1. 异常分析（Anomaly Detection）

异常分析是数据挖掘研究领域里比较重要的课题。我们可以把它简单理解成从给定的数据中找出“异常”点。在我们的应用中，这些”异常“点可能会关联到欺诈。既然知识图谱可以看做是一个图 （Graph），知识图谱的异常分析也大都是基于图的结构。由于知识图谱里的实体类型、关系类型不同，异常分析也需要把这些额外的信息考虑进去。大多数基于图的异常分析的计算量比较大，可以选择做离线计算。在我们的应用框架中，可以把异常分析分为两大类：静态分析和动态分析。

（1）静态分析

所谓的静态分析指的是，给定一个图形结构和某个时间点，从中去发现一些异常点（比如有异常的子图）。下图中我们可以很清楚地看到其中五个点的相互紧密度非常强，可能是一个欺诈组织。所以针对这些异常的结构，我们可以做出进一步的分析。

（2）动态分析

所谓的动态分析指的是分析其结构随时间变化的趋势。我们的假设是，在短时间内知识图谱结构的变化不会太大，如果它的变化很大，就说明可能存在异常，需要进一步的关注。分析结构随时间的变化会涉及到时序分析技术和图相似性计算技术。

1. 失联客户管理

除了贷前的风险控制，知识图谱也可以在贷后发挥其强大的作用。比如在贷后失联客户管理的问题上，知识图谱可以帮助我们挖掘出更多潜在的新的联系人，从而提高催收的成功率。

现实中，不少借款人在借款成功后出现不还款现象，而且玩“捉迷藏”，联系不上本人。即便试图去联系借款人曾经提供过的其他联系人，但还是没有办法联系到本人。这就进入了所谓的“失联”状态，使得催收人员也无从下手。那接下来的问题是，在失联的情况下，我们有没有办法去挖掘跟借款人有关系的新的联系人，而且这部分人群并没有以关联联系人的身份出现在我们的知识图谱里。如果我们能够挖掘出更多潜在的新的联系人，就会大大地提高催收成功率。举个例子，在下面的关系图中，借款人跟李四有直接的关系，但我们却联系不上李四。那有没有可能通过2度关系的分析，预测并判断哪些李四的联系人可能会认识借款人,这就涉及到图谱结构的分析。

1. 智能搜索及可视化展示

基于知识图谱，我们也可以提供智能搜索和数据可视化的服务。智能搜索的功能类似于知识图谱在Google, Baidu上的应用。也就是说，对于每一个搜索的关键词，我们可以通过知识图谱来返回更丰富，更全面的信息。比如搜索一个人的身份证号，我们的智能搜索引擎可以返回与这个人相关的所有历史借款记录、联系人信息、行为特征和每一个实体的标签（比如黑名单，同业等）。另外，可视化的好处不言而喻，通过可视化把复杂的信息以非常直观的方式呈现出来， 使得我们对隐藏信息的来龙去脉一目了然。

1. 精准营销

一个聪明的企业可以比它的竞争对手以更为有效的方式去挖掘其潜在的客户。在互联网时代，营销手段多种多样，但不管有多少种方式，都离不开一个核心-分析用户和理解用户。知识图谱可以结合多种数据源去分析实体之间的关系，从而对用户的行为有更好的理解。比如一个公司的市场经理用知识图谱来分析用户之间的关系，去发现一个组织的共同喜好，从而可以有针对性的对某一类人群制定营销策略。只有我们能更好的、更深入的（Deep understanding）理解用户的需求，我们才能更好地去做营销。

1. 产业链分析

产业链建模分析主要给证券、基金行业的行业研究员、基金经理、量化投资者提供分析框架。由于产业链本身就是各产业部门之间的技术经济关联，并依据特定的逻辑关系和时空布局关系客观形成的链条式关联关系形态，所以天然比较适合于用知识图谱这种图结构来展示。

知识图谱应用于产业链分析能够帮助行业研究员和基金经理透析整个行业、领域的技术与经济传导关系，实现概念发掘、事件影响分析、行情分析。目前市场上做这一部分的机构和公司逐渐多了起来，不过我觉得难点很多。核心问题是覆盖整个产业链上下游的公司、产品、事件因子异常庞大，其本质问题转化成了一个数据质量问题——即数据的完整程度决定了产业链知识图谱的完整程度。举例来说，假设我要对零售行业像苏宁云商这种个股进行产业链分析，需要关注企业链、价值链、供需链和空间链四个维度，从产品研发、原材料与设备、仓储管理、市场营销、销售物流、渠道分销、终端服务等各个环节的数据情况。这还只是零售行业的一个细分领域，一旦涉及到全行业，数据的运营、持续更新问题难度就更大。所以数据运营也成为产业链建模核心难点。

1. 担保链分析

担保链是多个企业在向金融机构融资时，通过互相担保、连环担保、联合担保等担保关系链条形成的特殊利益共同体。企业之间的债权债务复杂隐蔽，一旦某些企业发生违约就会引起区域或行业的系统性风险，所以目前银行业对担保违约现象的量化评估很困难。而且这种担保链条存在传染特性，任何一个企业出问题，风险都会传播，所以担保链风险也是银行信贷业务管理中的难点和痛点。

担保链形成的网状结构较为复杂，但基本有迹可循，可通过担保网络的类别形成如下几类：单向担保、互联互保、循环担保、复杂担保、融资担保、平台担保等。而分析网状结构的方法主要是社交网络分析：即网络中每个节点可以看作借款人和担保人，网络中的每条边作为担保关系和担保方向。

模型：针对商业银行构建担保链的方法主要是Modularity模型，也就是Community Detection，反映了群体(子网络)内部相对外部的连接密度，通过迭代式算法不断划分，最终形成担保网络。

分析：主要方法可以归纳为：结合不同担保网络的风险系统，通过违约历史，识别风险最大的担保链。然后，通过担保网络的风险指标，量化核心企业的风险指标，识别风险最大的企业。

识别：分析风险集中度较高的核心企业和风险扩散度较高的核心企业。结合多维度的借贷和担保信息,如风险等级、担保类型、贷款流向、行业分布等，多方面了解担保风险情况。

1. 用户画像与情感分析

用户画像有很多种做法，除了传统的客户标签和360视图之外，知识图谱能更好的反应与用户相关的关系、事件、行为等因子，使画像更具有结构化、动态化的过程。结合这些实时、动态、结构化的基础画像服务，为客户分类、精准营销、智能投顾、流失预警、适当性管理、反欺诈等上层应用提供基础。

用户画像的另外一个维度是情感分析。在社交网络、舆情与论坛中通过自然语言处理和语义分析，能够实时感知到群体与个体的情绪变化，反映到市场上可以作为刻画投资者情绪的情感图谱。情感分析的意义在于动态刻画投资人对于市场、公司、事件的认知和感知，能够部分反映走势与预期，这一部分目前在交易所、证券公司均有研究。

1. 智能知识库

智能业务知识库目的在于构建机构内部的知识管理，能够对知识的业务特性、角色、访问权限等知识体系和属性进行分类，提供知识体系结构。也可以通过人工智能和机器学习将问题场景与业务知识进行自动关联分析。智能知识库是一种混合了智能问答、知识图谱、语义搜索等场景的综合服务，在机构中通常作为中间件形式存在，底层与大数据和机器学习平台对接，相当于内部构建了问答式的业务维基百科，属于人工智能在业务流程优化、数据服务上的创新应用。

# 第五章 证券公司风险监测预警机制研究

本课题计划利用深度学习与自然语言处理等人工智能技术，对证券公司自营以及融资融券、股票质押等业务相关的股票持仓主体以及关联方主体进行画像，对新闻、公告等风险信息进行分析，分别构建自营以及融资融券、股票质押等业务相关的股票持仓主体知识图谱、交易对手主体知识图谱、风险事件图谱，对证券公司进行风险监测预警研究。通过网络爬虫系统对持仓主体及交易对手主体相关的新闻、公告等互联网信息进行信息爬取，将爬取信息运用自然语言处理等技术作为输入向量写入风险事件图谱，同时关联自营持仓主体、交易对手主体知识图谱通过深度学习模型输出对证券公司的风险监测预警信息。

课题整体研究框架见图5-1。



图5-1 课题研究整体框架

研究框架中主要涉及以下组成部分：

## 一、证券公司自营持仓主体以及交易对手图谱构建

证券公司自营持仓主体以及交易对手主体知识图谱主要是描述主体的固定属性以及各种关联关系。例如自营持仓主体的基本属性包括公司介绍、高管、集团、股东、债务、行业、上游、下游、概念、主营业务等属性。

知识图谱技术解决了知识的获取与整合，但仍需储存于某一介质，用于后续的查询和可视化展示。图形数据库对节点和节点间复杂关系的良好支持，成为了知识图谱存储的首选。本文选用图形数据库存储整合的知识图谱数据。

图形数据库是一种非关系型数据库。在一个图形数据库中，节点信息和节点间的链接信息构成了整个数据集。节点信息性质类似于传统的关系数据库中的表结构，每一个节点对应表中的一条信息。而关系则是图形数据库所特有的组成，与关系数据库对应的是表间数据的关联。

相较于传统的关系型数据库，图形数据库在涉及大量关系的存储和表示上更具优势。在需要描述大量关系时，传统的关系型数据库已经不堪重负。关系数据库的强项在于存储单独的实体信息而非实体间的关系。当记录和查询涉及大量对实体间的关系的操作时，数据库性能会直线下降。而对关系支持良好的图形数据库更适合这种场景。通过图形数据库可以更轻易和直观的表达节点间的相互关系。

Neo4J是Neo Technology公司研发的图形数据库，能让使用者借助图形处理不同行业和案例的关系。它具有以下特征：

1. 对事务的支持。Neo4j数据的修改操作是事务级的，能够保证数据操作过程中的一致性。
2. 对存储的横向扩展。为了应对负载增大给数据库带来的压力，通常有两种解决方案，分别是一写多读的主从复制方案和分片方案。因为图数据库侧重存储节点和其它结点的关联关系，因此不太适用于分片机制进行横向扩展。Neo4j 采用主从复制方案进行读写分割，单个Neo4j实例可以存储几十亿个结点及关系，这种横向扩展能力足以应对大部分的企业级应用。
3. 强大的图形搜索能力。Neo4j查询操作语言是Cypher，是专门为操作图形数据库设计的声明式图查询语言，可以高效的查询和更新图数据库。此外Neo4j也对目前主流的编程语言设计了对应的操作项目，如Java语言的Spring DataNeo4j，方便不同开发人员快速上手Neo4j的操作。

Cypher近似于关系数据库的SQL语言，但在关联查找能力上比SQL更加强大。以六度分隔理论为例，该理论提出任意两人的关系网络间隔人数不会超过六个。假设数据完整，通过Cypher查询可以很轻易的找到两人间隔的轨迹，而SQL查询很难找到这种轨迹。

完整的Cypher查询语句通常由以下结构构成：

1. 匹配子句，通过对数据库中的实例进行匹配查询，从而获取满足查询条件的数据；
2. 条件子句，作用类似SQL中的WHERE子句，通常作为匹配子句的组成部分，用于条件筛选；
3. 返回子句，用于返回指定的信息字段；
4. 创建子句，用于创建节点、关系或属性等。

Neo4j数据库存储结构见图5-2。

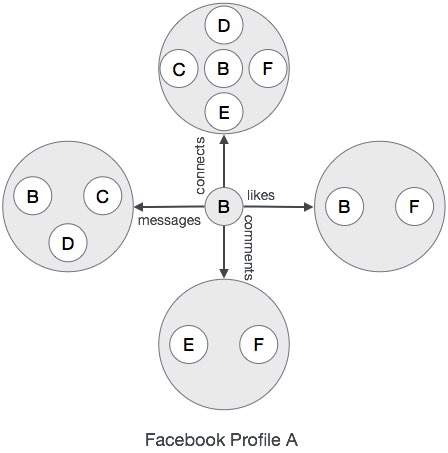


图5-2 Neo4j数据库存储结构图

Neo4j是一个原生的图数据库引擎，它存储了原生的图数据。因此，可以使用图结构的自然伸展特性来设计免索引邻近节点遍历的查询算法，即图的遍历算法设计。图的遍历是图数据结构所具有的独特算法，即从一个节点开始，根据其连接的关系，可以快速和方便地找出它的邻近节点。这种查找数据的方法并不受数据量的大小所影响，因为邻近查询始终查找的是有限的局部数据，不会对整个数据库进行搜索。所以，Neo4j具有非常高效的查询性能，相比于RDBMS可以提高数倍乃至数十倍的查询速度。

本文希望通过图形数据库来实现知识图谱的绘制，主要采用开源图形数据库Neo4j，绘制步骤见图5-3。

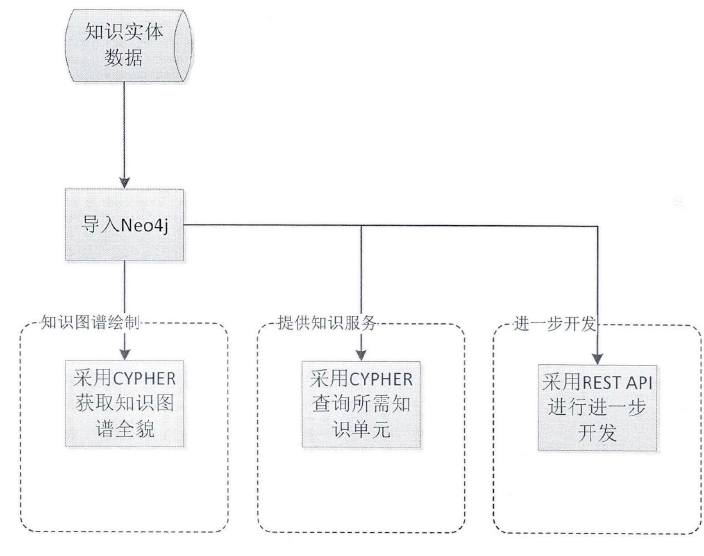


图5-3 绘制知识图谱流程图

1. 数据入库，釆用批量导入的方式将识别好的知识实体与实体关系导入至图形数据库中。
2. 釆用Cypher查询语言查询所有节点及关系即能获取整个知识图谱的全貌。
3. 釆用Cypher语言搜索所需的节点及关系信息，可以为使用者提供个性化的知识服务。
4. 采用编程的方式可以调用Neo4j的REST API接口进一步开发知识图谱界面。

首先将通过爬虫等方法得到的结构化数据可以直接存入Neo4j数据库，将得到的非结构化数据按照第二章构建知识图谱的过程进行命名实体识别和实体关系抽取后，存入Neo4j数据库中。

证券公司自营持仓主体以及交易对手图谱构建的方法及过程如下：

1. 命名实体识别

将爬虫等方法得到的原始数据按照前面提到的BMEO规则进行标注。例如“{{company:上海石化}}现有炼油、化工、塑料、[化纤](https://baike.baidu.com/item/%E5%8C%96%E7%BA%A4)等主要生产装置72套，以现代化、大型化、连续化为主要特征。目前，公司拥有年{{product:[原油](https://baike.baidu.com/item/%E5%8E%9F%E6%B2%B9)}}加工1400万吨，年产{{product:[乙烯](https://baike.baidu.com/item/%E4%B9%99%E7%83%AF)}}95万吨、{{product:[成品油](https://baike.baidu.com/item/%E6%88%90%E5%93%81%E6%B2%B9)}}及{{product:化工品}}510万吨、{{product:合成树脂}}及{{product:塑料制品}}95万吨、{{product:合纤原料}}及{{product:合成纤维}}138万吨的生产能力；拥有独立的水、电、汽、气公用工程供应系统，独立的环保处理系统，以及海运、[内河航运](https://baike.baidu.com/item/%E5%86%85%E6%B2%B3%E8%88%AA%E8%BF%90/12770449)码头和铁路、[公路运输](https://baike.baidu.com/item/%E5%85%AC%E8%B7%AF%E8%BF%90%E8%BE%93/4987156)等设施。至2004年底，公司总资产达到288亿元，在册员工总数2.85万人。”

然后使用Python将标注好的数据处理成字标注的形式，将每一个字和标注转换成id的形式，供模型训练使用。部分模型代码如下：

1. **for** line **in** input\_data.readlines():
2. line=line.strip()
3. i=0
4. **while** i <len(line):
5. **if** line[i] == '{':
6. i+=2
7. temp=""
8. **while** line[i]!='}':
9. temp+=line[i]
10. i+=1
11. i+=2
12. word=temp.split(':')
13. sen = word[1]
14. output\_data.write(sen[0]+"/B\_"+word[0]+" ")
15. **for** j **in** sen[1:len(sen)-1]:
16. output\_data.write(j+"/M\_"+word[0]+" ")
17. output\_data.write(sen[-1]+"/E\_"+word[0]+" ")
18. **else**:
19. output\_data.write(line[i]+"/O ")
20. i+=1
21. output\_data.write('\n')
22. **def** X\_padding(words):
23. ids = list(word2id[words])
24. **if** len(ids) >= max\_len:
25. **return** ids[:max\_len]
26. ids.extend([0]\*(max\_len-len(ids)))
27. **return** ids
29. **def** y\_padding(tags):
30. ids = list(tag2id[tags])
31. **if** len(ids) >= max\_len:
32. **return** ids[:max\_len]
33. ids.extend([0]\*(max\_len-len(ids)))
34. **return** ids
35. df\_data = pd.DataFrame({'words': datas, 'tags': labels}, index=range(len(datas)))
36. df\_data['x'] = df\_data['words'].apply(X\_padding)
37. df\_data['y'] = df\_data['tags'].apply(y\_padding)
38. x = np.asarray(list(df\_data['x'].values))
39. y = np.asarray(list(df\_data['y'].values))

按照第三章构建知识图谱的步骤与方法，进行命名实体识别。深度学习模型使用了深度学习框架Pytorch，部分模型代码如下：

1. **class** BiLSTM\_CRF(nn.Module):
2. **def** \_\_init\_\_(self, vocab\_size, tag\_to\_ix, embedding\_dim, hidden\_dim):
3. super(BiLSTM\_CRF, self).\_\_init\_\_()
4. self.embedding\_dim = embedding\_dim
5. self.hidden\_dim = hidden\_dim
6. self.vocab\_size = vocab\_size
7. self.tag\_to\_ix = tag\_to\_ix
8. self.tagset\_size = len(tag\_to\_ix)
9. self.word\_embeds = nn.Embedding(vocab\_size, embedding\_dim)
10. self.lstm = nn.LSTM(embedding\_dim, hidden\_dim // 2,
11. num\_layers=1, bidirectional=True)
12. self.hidden2tag = nn.Linear(hidden\_dim, self.tagset\_size)
13. self.transitions = nn.Parameter(torch.randn(self.tagset\_size, self.tagset\_size))
14. self.transitions.data[tag\_to\_ix[START\_TAG], :] = -10000
15. self.transitions.data[:, tag\_to\_ix[STOP\_TAG]] = -10000
16. self.hidden = self.init\_hidden()
17. **def** \_score\_sentence(self, feats, tags):
18. score = torch.zeros(1)
19. tags = torch.cat([torch.tensor([self.tag\_to\_ix[START\_TAG]], dtype=torch.long), tags])
20. **for** i, feat **in** enumerate(feats):
21. score = score + \
22. self.transitions[tags[i + 1], tags[i]] + feat[tags[i + 1]]
23. score = score + self.transitions[self.tag\_to\_ix[STOP\_TAG], tags[-1]]
24. **return** score

通过命名实体识别，将得到的数字id根据之前的对应关系转换回汉字，稍做处理就可以获得文本中的实体，例如：

公司：上海石化；产品：乙烯；产品：成品油；产品：合成树脂……

1. 实体关系抽取

然后可以对文本进行实体关系抽取，进行实体关系抽取的原始数据，例如“消息 7月18日晚间，[上海石化](http://stockdata.stock.hexun.com/600688.shtml)([600688](http://stockdata.stock.hexun.com/600688.shtml),[股吧](http://guba.hexun.com/600688,guba.html))发布[公告](http://stockdata.stock.hexun.com/2009_jrgg.shtml)称，单独或者合计持有 50.56%公司股份的股东[中国石油](http://stockdata.stock.hexun.com/gghq_601857.shtml)化工股份有限公司，在 2017 年 7 月 14 日提出临时提案并书面提交公司董事会，提名李远勤为公司第九届董事会独立非执行董事，并提请将相关议案作为临时提案提交公司 2017 年第一次临时股东大会进行审议。”

可以从中抽取出实体“上海石化”“中国石油化工股份有限公司”“李远勤”。通过定义实体之间的关系[上海石化，股东，中国石油化工股份有限公司]、[上海石化，董事，李远勤]。将该段文本转化为词向量表示，计算句子中每个词距离两个实体在句子中的距离，把距离转换成距离向量，与词向量合并，作为模型输入。部分代码如下：

1. **from** compiler.ast **import** flatten
2. all\_words = flatten(datas)
3. sr\_allwords = pd.Series(all\_words)
4. sr\_allwords = sr\_allwords.value\_counts()
5. #print sr\_allwords
6. set\_words = sr\_allwords.index
7. set\_ids = range(1, len(set\_words)+1)
8. word2id = pd.Series(set\_ids, index=set\_words)
9. id2word = pd.Series(set\_words, index=set\_ids)
10. **print** "word2id",len(word2id)
12. all\_word = flatten(labels)
13. sr\_allword = pd.Series(all\_word)
14. sr\_allword = sr\_allword.value\_counts()
15. **print** sr\_allword
17. max\_len = 50
18. senssslen = 0
19. **def** X\_padding(words):
20. #print words
21. ids = list(word2id[words])
22. **if** len(ids) >= max\_len:
23. **return** ids[:max\_len]
24. ids.extend([0]\*(max\_len-len(ids)))
25. index1 = words[2:].index(words[0])
26. index2 = words[2:].index(words[1])
27. **for** i **in** range(max\_len):
28. ids.append(i-index1+max\_len)
29. **for** i **in** range(max\_len):
30. ids.append(i-index2+max\_len)
31. **return** ids
33. df\_data = pd.DataFrame({'words': datas, 'tags': labels}, index=range(len(datas)))
34. df\_data['x'] = df\_data['words'].apply(X\_padding)
35. df\_data['y'] = df\_data['tags']
36. x = np.asarray(list(df\_data['x'].values))
37. y = np.asarray(list(df\_data['y'].values))

然后就可以通过深度学习模型进行训练，仍然使用Pytorch深度学习框架，模型部分代码如下：

1. **class** Net(nn.Module):
2. **def** \_\_init\_\_(self,word\_size,word\_embedding\_dim,position\_size,position\_embedding\_dim):
3. super(Net, self).\_\_init\_\_()
4. self.word\_embeds = nn.Embedding(word\_size,word\_embedding\_dim)
5. self.position\_embeds = nn.Embedding(position\_size,position\_embedding\_dim)
6. self.M1 = nn.Parameter(torch.randn(word\_embedding\_dim+position\_embedding\_dim\*2,word\_embedding\_dim))
7. self.M2 = nn.Parameter(torch.randn(word\_embedding\_dim+position\_embedding\_dim\*2,word\_embedding\_dim))
8. self.b1 = nn.Parameter(torch.randn(50))
9. self.b2 = nn.Parameter(torch.randn(50))
10. self.padding = nn.ReplicationPad2d((0, 0, 1, 1))
11. self.conv1 = nn.Conv2d(1, 5, kernel\_size=(3,word\_embedding\_dim+position\_embedding\_dim\*2),stride=1,padding=0)
12. self.fc = nn.Linear(5, 12)
13. **def** att(self,x,e1,e2):
14. S1 = torch.mm(x,self.M1)
15. e1 = torch.t(e1.view(1,-1))
16. S1 = torch.squeeze(torch.mm(S1,e1),1) + self.b1
17. #print S1.size()
18. S2 = torch.squeeze(torch.mm(torch.mm(x,self.M2),torch.t(e2.view(1,-1))),1) +self.b2
19. #print S2.size()
20. S1sum = 0
21. **for** i **in** range(len(S1)):
22. S1sum+=S1[i]
23. a1 = torch.diag(S1/S1sum)
24. S2sum = 0
25. **for** i **in** range(len(S2)):
26. S2sum+=S2[i]
27. a2 = torch.diag(S2/S2sum)
28. **return** torch.mm((a1+a2)/2,x)
30. **def** forward(self, x):
31. e1 = self.word\_embeds(x[0])
32. e2 = self.word\_embeds(x[1])
33. x = self.add\_position\_embeds(x)
34. x = self.att(x,e1,e2).view(1,1,50,-1)
35. x = self.padding(x)
36. x = self.conv1(x)
38. x = F.max\_pool2d(x,(50,1)).view(1,-1)
39. x = F.softmax(self.fc(x),dim=1)
40. **return** x
41. Neo4j数据库

将非结构化数据与半结构化数据，通过命名实体识别与实体关系抽取后，就可以得到关于上市公司知识图谱的三元组结果。例如：

[上海石化，产品，乙烯]；

[上海石化，董事，李远勤]；

[上海石化，上游，上海未来]；

[上海石化，股东，中国石化]。

将获得的数据三元组存入neo4j数据库中，就得到了上市公司知识图谱，见图5-4。

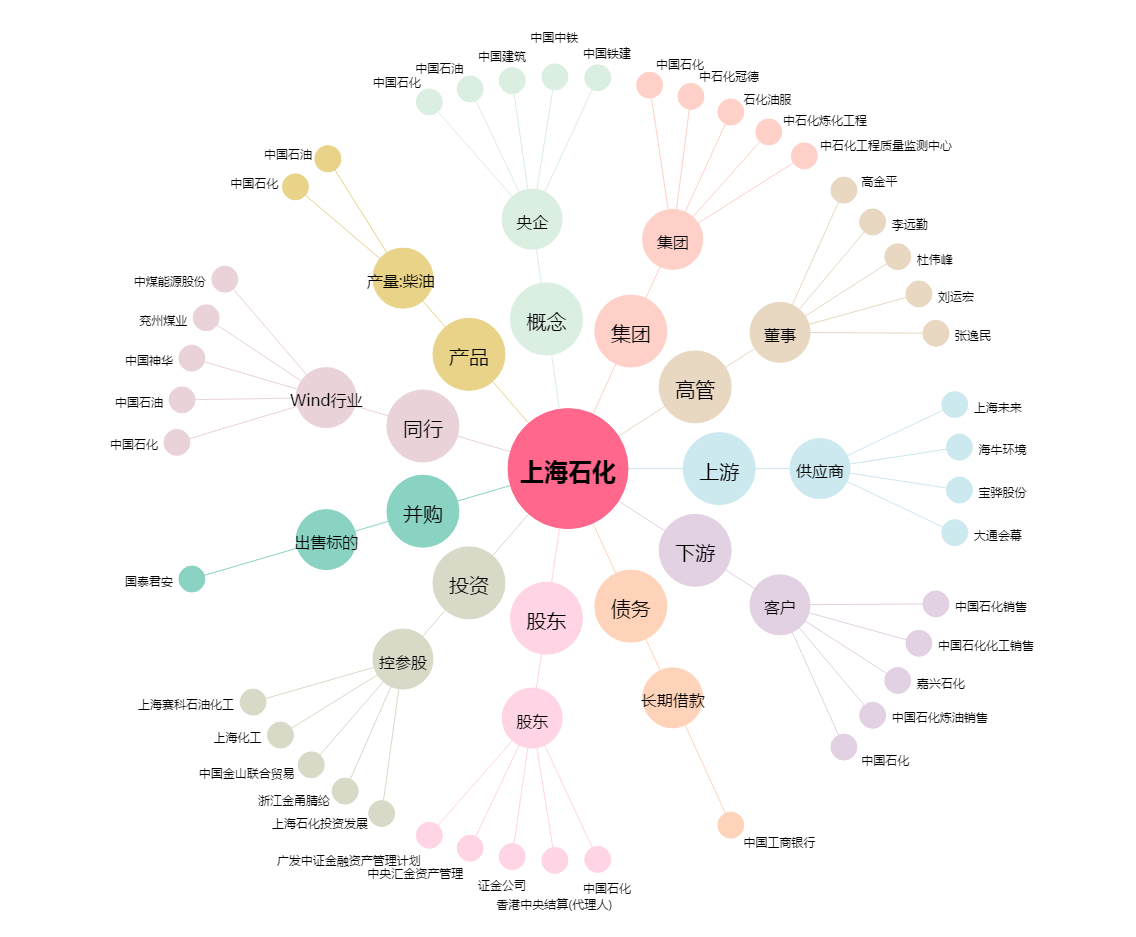


图5-4 上市公司图谱示例图

## 二、风险事件图谱构建

1. 风险事件分类以及风险评判标准

（1）关于企业风险

2006年，国务院国资委印发《中央企业全面风险管理指引》，其对风险的定义是未来的不确定性对企业实现经营目标的影响，另经查阅各类学术文献，笔者认为，所谓企业风险，均是从影响企业实现短期或长期战略、经营目标角度，对企业正常运营带来潜在威胁、潜在损失的因素或不确定性。

（2）企业风险的分类方法

按照不同的分类标准风险类别有不同的分类方法，依照风险的内容和来源，《中央企业全面风险管理指引》将企业风险分为战略风险、财务风险、市场风险、运营风险、法律风险；依照能否为企业带来盈利等机会，《中央企业全面风险管理指引》又将风险分为纯粹风险和机会风险；根据风险的来源以及范围，COSO内部控制报告认为企业层面的风险来自外部因素或内部因素，中国内部审计协会颁布的《风险管理审计准则》将风险分为外部风险和内部风险。可以看出，不管是国内权威机构发布的分类方法还是国外标准组织发布的分类方法，都存在不同类别的风险要素相互交叉的现象，在本文中，我们参照《中央企业全面风险管理指引》对风险的分类方法进行模型训练及原型分析示例，将企业面临的主要风险划分为战略风险、[财务风险](https://baike.baidu.com/item/%E8%B4%A2%E5%8A%A1%E9%A3%8E%E9%99%A9/1520903)、[市场风险](https://baike.baidu.com/item/%E5%B8%82%E5%9C%BA%E9%A3%8E%E9%99%A9/1690357)、[运营风险](https://baike.baidu.com/item/%E8%BF%90%E8%90%A5%E9%A3%8E%E9%99%A9/357544)、[法律风险](https://baike.baidu.com/item/%E6%B3%95%E5%BE%8B%E9%A3%8E%E9%99%A9/6905624)等五大类，并将分类下钻两级，形成风险三级分类，见表5-1。

战略风险：可理解为企业整体损失的不确定性，具体指影响整个企业的发展方向、企业文化、信息和生存能力或企业效益的不确定因素。

财务风险：是财务成果和财务状况的风险，具体指公司财务结构不合理、融资不当使公司可能丧失偿债能力而导致投资者预期收益下降的风险。

市场风险：指未来市场价格（利率、汇率、股票价格和商品价格）的不确定性对企业实现其既定目标的影响。

运营风险：指企业在运营过程中，由于外部环境的复杂性和变动性以及主体对环境的认知能力和适应能力的有限性，而导致的运营失败或使运营活动达不到预期的目标的可能性及其损失。

法律风险：在法律实施过程中，由于企业外部的法律环境变化，或由于包括企业自身在内的各种主题未按照法律规定或合同约定行使权力、履行义务而对企业造成负面法律后果的可能性。

表 5-1 企业风险分类

|  |  |  |
| --- | --- | --- |
| **一级风险** | **二级风险** | **三级风险** |
| 战略风险 | 投资风险 | 投资决策风险 |
| 投资实施风险 |
| 投资中止退出风险 |
| 政策风险 |  |
| 国际化经营风险 | 境外投资风险 |
| 国际工程承包风险 |
| 海外市场开拓风险 |
| 战略管理风险 | 战略规划风险 |
| 战略实施风险 |
| 战略调整风险 |
| 宏观经济风险 |  |
| 产业结构风险 |  |
| 改制风险 |  |
| 并购重组风险 | 估值与定价风险 |
| 尽职调查风险 |
| 执行与整合风险 |
| 公司治理风险 |  |
| 组织结构风险 |  |
| 集团管控风险 |  |
| 社会责任风险 |  |
| 企业文化风险 | 企业文化建设风险 |
| 廉政建设风险 |
| 职业道德风险 |
| 公共关系风险 | 政府关系风险 |
| 媒体关系风险 |
| 危机沟通风险 |
| 社会舆情风险 |
| 业务合作伙伴风险 | 业务合作伙伴关系风险 |
| 业务合作伙伴信用风险 |
| 市场风险 | 竞争风险 |  |
| 价格风险 |  |
| 汇利率风险 |  |
| 市场供求风险 | 市场供应风险 |
| 市场需求风险 |
| 衍生品交易风险 |  |
| 市场营销风险 |  |
| 行业前景风险 |  |
| 市场分布风险 |  |
| 市场开发风险 |  |
| 证券市场风险 |  |
| 客户风险 | 客户信用风险 |
| 客户关系维护风险 |
| 客户商业模式风险 |
| 品牌与声誉风险 | 品牌策略风险 |
| 品牌推广及维护风险 |
| 声誉风险 |
| 财务风险 | 现金流风险 | 融资风险 |
| 资金短缺风险 |
| 债务风险 |
| 应收/预付账款风险 |
| 资金管理风险 | 资金使用风险 |
| 资金安全风险 |
| 预算管理风险 | 预算编制风险 |
| 预算执行风险 |
| 预算考核风险 |
| 会计与报告风险 | 会计核算风险 |
| 财务报告风险 |
| 成本费用风险 |  |
| 担保风险 |  |
| 税务管理风险 | 税务操作风险 |
| 税务筹划风险 |
| 税金缴纳风险 |
| 关联交易风险 |  |
| 资本运作风险 |  |
| 信托风险 |  |
| 盈利能力风险 |  |
| 法律风险 | 合同管理风险 |  |
| 法律纠纷风险 |  |
| 合规管理风险 |  |
| 知识产权风险 |  |
| 重大决策法律风险 |  |
| 运营风险 | 健康安全环保风险 | 安全生产风险 |
| 职业健康风险 |
| 环境保护风险 |
| 节能减排风险 |
| 人力资源风险 | 人力资源规划风险 |
| 招聘与留任风险 |
| 人员配置风险 |
| 关键人才流失风险 |
| 人才储备风险 |
| 培训与发展风险 |
| 绩效考核风险 |
| 薪酬与福利风险 |
| 劳动关系管理风险 |
| 一般项目管理风险 |  |
| 技术风险 | 技术变革风险 |
| 技术停滞、落后风险 |
| 技术引进风险 |
| 技术应用风险 |
| 技术创新风险 |
| 产品风险 | 产品结构/规划风险 |
| 产品生命周期风险 |
| 产品质量风险 |
| 资源保障风险 |  |
| 保密风险 |  |
| 研发与开发风险 | 产品研发风险 |
| 技术研发风险 |
| 存货风险 |  |
| 信息系统风险 | 信息系统安全风险 |
| 信息系统规划风险 |
| 信息系统架构风险 |
| 信息系统运行风险 |
| 运行控制风险 |  |
| 稳定风险 |  |
| 执行力风险 |  |
| 采购风险 |  |
| 业务伙伴风险 |  |
| 业务模式风险 |  |
| 业务协同风险 |  |
| 操作风险 |  |
| 业务转型风险 |  |
| 生产管理风险 |  |
| 销售风险 | 销售渠道风险 |
| 产品交付风险 |
| 退货风险 |
| 销售实施风险 |
| 供应链风险 |  |
| 物流管理风险 |  |
| 贸易风险 |  |
| 工程项目管理风险 | 工程设计风险 |
| 工程造价风险 |
| 工程概预算风险 |
| 工程招投标风险 |
| 工程分包风险 |
| 工程进度风险 |
| 工程质量风险 |
| 工程安全风险 |
| 工程竣工风险 |
| 资产管理风险 | 有形资产管理 |
| 无形资产管理 |
| 审计监察风险 | 审计计划风险 |
| 审计执行风险 |
| 审计报告风险 |
| 新业务开发风险 |  |
| 突发事件管理风险 |  |
| 退市风险 |  |

资料来源：根据百度文库关于企业风险的定义与分类公开资料自行整理

（3）企业风险评分权重

确定风险评分权重，即划分风险等级，针对各类风险确定具体的权重分值。常用的风险等级划分方法即风险评估矩阵法，主要是通过主观评估风险发生的概率以及风险发生后对企业造成的影响大小（即风险影响程度）来对各类风险进行等级划分。其中涉及到的频率及影响程度，对每个等级均赋予一个具体的分值，最终通过频率与影响程度得分乘积的形式确定某一项风险的具体等级得分。对于风险的发生频率及影响程度，可考虑以行业特征及历史经验为依据进行等级划分，作为模型原型分析，在此不再区分行业，仅以统一的标准进行示例说明。

1. 企业风险发生频率划分标准，见表5-2。

表5-2 企业风险发生频率划分标准

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段** | **备注** | | | | |
| 发生频率 | 极低 | 低 | 中等 | 高 | 极高 |
| 定性描述 | 极个别情况下发生 | 偶尔发生 | 时常发生 | 经常发生 | 频繁发生 |
| 定量描述 | 几年发生1次 | 一年发生1到3次 | 三个月内发生1到4次 | 每个月发生1到4次 | 每周发生2次或以上 |
| 得分 | 1 | 2 | 3 | 4 | 5 |

资料来源：课题组自行整理

1. 企业风险影响程度，见表5-3。

表5-3 企业风险影响程度

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **影响程度** | **描述** | **损失参考值** | | **得分** |
| **范围** | **说明** |
| 极大 | 极大的损失金额或负面影响 | [50，+∞) | 潜在损失金额在50万元（含）以上、存在较大潜在风险、导致公司分类评价扣分、可能给公司运营、声誉造成全国、行业范围内较大不良影响等危害特别严重的风险 | 5 |
| 较大 | 较大的损失金额或负面影响 | [10,50) | 潜在损失金额在10万元（含）与50万元之间、可能减少公司营收、引发监管关注并需向监管做专项说明、可能给公司运营、声誉造成区域、某业务领域范围内较大不良影响等危害较大的风险 | 4 |
| 中等 | 中等的损失金额或负面影响 | [1,10) | 损失金额在1万元（含）与10万元之间、可能引发监管关注、对公司正常营运影响较小、迅速处置后仍可对公司产生一定不良影响的 | 3 |
| 较小 | 较小的损失金额或负面影响 | [0.1,1) | 损失金额在0.1万元（含）与1万元之间，可能对公司运营产生较小影响、在本单位范围内自行妥善后未造成不良影响的 | 2 |
| 极小 | 极小的损失金额或负面影响 | [0,0.1) | 损失金额在0与0.1万元之间，危害较小，不会对公司运营及声誉产生不良影响的风险 | 1 |

资料来源：课题组自行整理

1. 企业风险权重分数确定

对于企业风险权重得分，采用对应频率与影响程序的乘积来表示，即：

某项企业风险权重得分=风险发生频率×风险影响程度

1. 企业风险权重得分举例

例如，对于企业运营风险中的审计报告风险，经主观评估，其发生频率极低，仅在极个别情况下发生，则其对应的发生频率得分为1分，该风险发生后，有可能对企业经营决策产生较大影响、掩盖企业经营中的发展问题，则其对应的影响程度得分为4分，故审计报告风险最终的权重得分为1×4=4分。

1. 风险图谱构建

通过上述公司基本信息图谱的构建作为参考，我们可以同样的采取类似的办法进行风险事件图谱的构建。相对于公司基本信息图谱的构建，风险事件图谱的构建中其基本信息数据应当更加关注公司的网络舆情、行政处罚情况、经营情况等数据。

风险事件图谱构建的主要步骤，见图5-5。

爬虫

事件抽取

构建事件图谱

图5-5 风险事件图谱构建流程图

（1）爬虫

网络爬虫（Web Crawler），又称网络蜘蛛（Web Spider）或网络机器人（Web Robot），是一种按照一定的规则自动抓取万维网资源的程序或者[脚本](http://baike.baidu.com/view/54.htm)，已被广泛应用于互联网领域。搜索引擎使用网络爬虫抓取Web网页、文档甚至图片、音频、视频等资源，通过相应的索引技术组织这些信息，提供给搜索用户进行查询。随着网络的迅速发展，万维网成为大量信息的载体，如何有效地提取并利用这些信息成为一个巨大的挑战。不断优化的网络爬虫技术正在有效地应对这种挑战，为高效搜索用户关注的特定领域与主题提供了有力支撑。网络爬虫也为中小站点的推广提供了有效的途径，网站针对搜索引擎爬虫的优化曾风靡一时。

传统爬虫从一个或若干初始网页的URL开始，获得初始网页上的URL，在抓取网页的过程中，不断从当前页面上抽取新的URL放入队列,直到满足系统的一定停止条件。聚焦爬虫的工作流程较为复杂，需要根据一定的网页分析算法过滤与主题无关的链接，保留有用的链接并将其放入等待抓取的URL队列。然后，它将根据一定的搜索策略从队列中选择下一步要抓取的网页URL，并重复上述过程，直到达到系统的某一条件时停止。另外，所有被爬虫抓取的网页将会被系统存贮，进行一定的分析、过滤，并建立索引，以便之后的查询和检索；对于聚焦爬虫来说，这一过程所得到的分析结果还可能对以后的抓取过程给出反馈和指导。

网络爬虫的基本工作流程：

1. 首先选取一部分精心挑选的种子URL；
2. 将这些URL放入待抓取URL队列；
3. 从待抓取URL队列中取出待抓取在URL，解析DNS，并且得到主机的ip，并将URL对应的网页下载下来，存储进已下载网页库中。此外，将这些URL放进已抓取URL队列；
4. 分析已抓取URL队列中的URL，分析其中的其他URL，并且将URL放入待抓取URL队列，从而进入下一个循环。

风险事件图谱的数据主要来源于网络爬虫，通过网络爬虫系统对上市公司相关的新闻、公告等互联网信息进行信息爬取。例如上市公司的公告、财报等业务信息以及公开的招聘信息，此类信息可以较为直观地反映一个公司的经营状况。通过抓取互联网上微博、微信、新闻站点中企业相关的新闻事件，可以反映该公司在各个时间段的网络舆情水平。

常用的爬虫框架基本可以分为3类：

分布式爬虫，如Nutch；

JAVA单机爬虫，如Crawler4j、WebMagic、WebCollector；

非JAVA单机爬虫，如scrapy。

（2）事件抽取技术

事件抽取是信息抽取领域一个重要的研究方向。事件抽取把含有事件信息的非结构化文本以结构化的形式呈现出来，在自动文摘、自动问答、信息检索等领域有着广泛的应用。 近些年来，事件抽取一直吸引着许多研究机构和研究者的注意力。MUC（Message Understanding Conference）和ACE（Automatic Content Extraction）会议是典型的含有事件抽取任务的评测会议。

事件抽取主要是为了抽取出于上市公司风险相关的一些事件信息。例如宏观环境风险、行业环境风险、技术风险、经营风险、财务风险以及其他影响上市公司健康度指标以及公司风险的相关事件。

事件由事件触发词和描述事件结构的元素构成。其中事件发生的时间、地点和参与角色称为事件元素。事件触发词是引起事件的发生，是决定事件类别的重要特征。例如图5-6中出生是该事件的触发词，事件的三个组成元素是“毛泽东”“1893年”“湖南湘潭”。

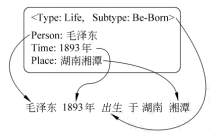


图5-6 “出生”事件的基本组成要素

所以事件抽取又可以分为以下三个步骤：

1. 事件触发词抽取和事件类型识别；
2. 事件组成元素识别；
3. 事件与元素关系识别。

其中事件组成元素识别与命名实体识别技术相似，可以使用BiLSTM+CRF模型抽取。事件与元素关系识别和实体关系识别类似，可以使用BiLSTM+Attention模型识别。

事件触发词识别和事件类型识别模型见图5-7。

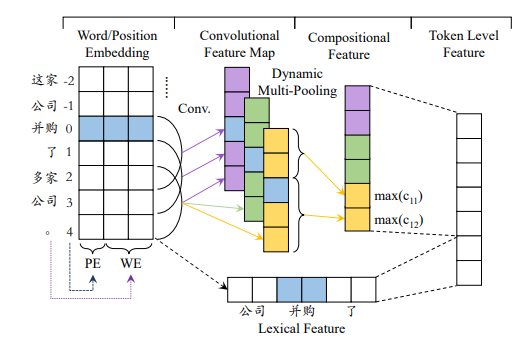


图5-7 事件触发词识别模型（1）

模型先对中文进行了分词，然后将分词后的句子通过Word Embedding和Position Embedding，再通过三个词做一次卷积的CNN。将通过CNN提取到的特征与该词向量进行拼接。拼接后的向量作为输出输入下一层网络中。下一层的网络结构见图5-8。

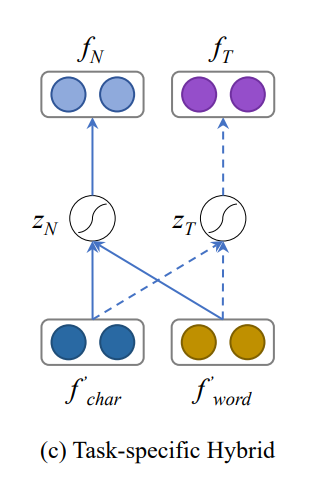


图5-8 事件触发词识别模型（2）

该神经网络层用来将从词级别和字级别抽取出来的特征进行合并，并分开供触发词识别和事件关系识别使用，其中的输入左边是词级别的特征，即上面模型抽取出来的特征，右边输入是字级别的特征，和词级别的特征抽取技术相同。然后将两个特征分别乘以各自的权重，加上一个偏置，通过sigmod激活函数。获得fN和fT两个输出，作为触发词识别和事件关系识别的输入。合并输出公式为

最后是触发词的识别分类方式。根据ACE和Rich ERE的统计，超过98.5%的触发词不超过3个汉字，所以我们定义每个字分别和前后的连个字作为可能的触发词，有6个分类再加一个非触发词，一个7个分类，见图5-9。

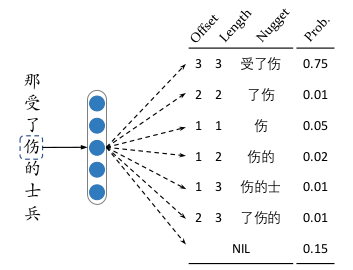


图5-9 事件触发词识别模型（3）

这里当“伤”为中心字时，它和它前后的两个字组成的词一个是如图所示的6种，模型识别出来“受了伤”是触发词的概率最大，也因此识别出了“那受了伤的士兵”这句话中的触发词。

而另一个事件类型识别即按照我们定义的事件类型，一个分成n类，把刚刚提取出来的特征作为输入，经过一次全连接输出n个输出即可。

（4）生成风险事件图谱

生成事理图谱也就是事理图谱的储存。和知识图谱类似，我们需要将事件抽取和事件关系抽取产生的三元组储存到图数据库中。目前使用较多的图形数据库主要有Neo4j、FlockDB、TAO 等，生成的风险事件图谱。

风险事件图谱具体的构建方法是将通过爬虫得到的数据，利用第二章介绍过的深度学习模型，得到某个公司相关的风险事件，将风险事件与对应的上市公司关联，存入neo4j数据库中就得到了风险事件图谱。

部分代码如下：

1. graph = Graph("http://127.0.0.1:7474",username="neo4j",password="\*\*\*")
2. **def** insert(company,event):
3. node = graph.nodes.match("company",name=company).first()
4. **if** **not** node:
5. node = Node("company",name=company)
6. graph.create(node)
7. nodeEvent = Node("event",name=event)
8. relation = Relationship(node,'event',nodeEvent)
9. graph.create(relation)

生成的风险事件图，见图5-10。

上海华信

2018.3.1 上海华信董事局主席叶简明被调查

2018.3.1 午间，华信对全部交易申请停牌

4.9日，股票被司法冻结

图5-10 风险事件图谱

## 三、公司健康度判别模型

公司健康度判别模型见图5-11。

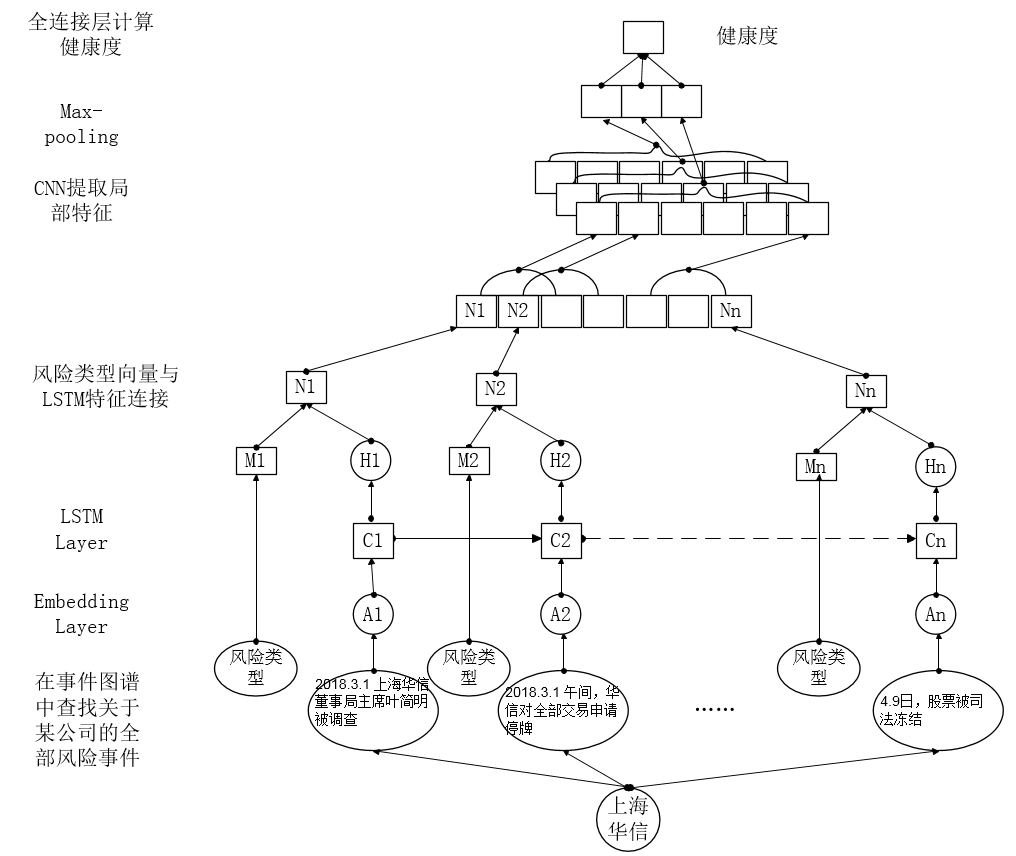


图5-11 公司健康度判别模型

判别某一个上市公司的健康度时，根据公司名称，在风险事件图谱中查找该公司的所有风险事件，以及每一个风险事件的风险类型，分别将风险事件与风险类型通过word Embedding转化成向量形式表示后，将风险事件通过LSTM提取事件特征，然后将事件特征与风险类型向量连接成一个向量，作为CNN的输入，通过卷积层和max pooling层提取该上市公司的所有 风险事件的特征，根据这些特征就可以计算该上市公司的健康度。

其中，卷积神经网络中每层卷积层由若干卷积单元组成，每个卷积单元的参数都是通过[反向传播算法](https://zh.wikipedia.org/wiki/%E5%8F%8D%E5%90%91%E4%BC%A0%E6%92%AD%E7%AE%97%E6%B3%95)最佳化得到的。卷积运算的目的是提取输入的不同特征，第一层卷积层可能只能提取一些低级的特征如边缘、线条和角等层级，更多层的网路能从低级特征中迭代提取更复杂的特征。

池化（Pooling）是卷积神经网络中另一个重要的概念，它实际上是一种形式的降采样。有多种不同形式的非线性池化函数，而其中“最大池化（Max pooling）”是最为常见的。它是将输入的图像划分为若干个矩形区域，对每个子区域输出最大值。直觉上，这种机制能够有效地原因在于，在发现一个特征之后，它的精确位置远不及它和其他特征的相对位置的关系重要。池化层会不断地减小数据的空间大小，因此参数的数量和计算量也会下降，这在一定程度上也控制了[过拟合](https://zh.wikipedia.org/wiki/%E8%BF%87%E6%8B%9F%E5%90%88)。通常来说，CNN的卷积层之间都会周期性地插入池化层。

池化层通常会分别作用于每个输入的特征并减小其大小。目前最常用形式的池化层是每隔2个元素从图像划分出2\*2的区块，然后对每个区块中的4个数取最大值。这将会减少75%的数据量。

除了最大池化之外，池化层也可以使用其他池化函数，例如“平均池化”甚至“[L2-范数](https://zh.wikipedia.org/wiki/Lp%E7%A9%BA%E9%97%B4#.E9.95.BF.E5.BA.A6.E3.80.81.E8.B7.9D.E7.A6.BB.E4.B8.8E.E8.8C.83.E6.95.B0)池化”等。过去，平均池化的使用曾经较为广泛，但是最近由于最大池化在实践中的表现更好，平均池化已经不太常用。

循环神经网络（RNN）的使用是用来处理序列数据的。在传统的神经网络中模型中，层与层之间是全连接的，每层之间的节点是无连接的。但是这种普通的神经网络对于很多问题是无能为力的。循环神经网络（RNN）指的是一个序列当前的输出与之前的输出也有关。具体的表现形式为网络会对前面的信息进行记忆，保存在网络的内部状态中，并应用于当前输出的计算中，即隐含层之间的节点不再无连接而是有链接的，并且隐含层的输入不仅包含输入层的输出还包含上一时刻隐含层的输出。理论上，循环神经网络能够对任何长度的序列数据进行处理，但是在实践中，为了减低复杂性往往假设当前的状态只与前面的几个状态相关。LSTM已经在第二章详细介绍，此处不再赘述。

本文的风险监测机制可从两方面对证券公司风险进行监测预警。

（1）监测公司的自营持仓主体以及关联对手方的健康度

通过实时监测公司自营持仓主体以及关联对手方的所关联的各类信息及其对公司健康度的影响，实时判别公司的整体健康程度，从而监测、预警证券公司的风险程度。

（2）监测市场信息、新闻等重大事件是否对证券公司相关联的主体产生影响

实时监测全网的各类风险事件，匹配信息相关的公司主体，与证券公司的持仓主体与关联方进行匹配，从而判别风险事件对证券公司健康度的影响程度。

实例分析：

证券公司购买了上市公司华信国际（002018）的股票，通过实时监测全网风险事件：

1. 2018年3月1日，市场传言上海华信董事局主席叶简明被调查；
2. 2018年3月1日当天15华信债下跌32.65%，上海华信午间对全部交易所债券申请了停牌；
3. 2018年4月9日，上海华信公告公司涉及重大诉讼及股权冻结事项，上海华信持有的华信国际股票已被司法冻结；
4. 2018年3月2日至5月2日评级机构多次下调上海华信主体评级及15华信债债项评级，主体信用等级由AAA下调至BBB-，债项信用等级由AAA下调至BBB+。

通过该风险监测机制可实现：

1. 实时监测上市公司华信国际（002018）相关的互联网信息，监测到风险事件1、2、3、4分别对应了实际控制人风险、债务风险、诉讼风险、信用风险，可通过训练好的深度学习模型输出华信国际的风险分值，根据风险分值对证券公司本身的风险程度进行判别、预警；
2. 实时监测全网的各类风险事件，监测到风险事件1、2、3、4时，监控系统自动监测出风险事件的主体是华信国际及其关联上市公司（例如叶简明作为实际控制人的其他公司，华信国际参股控股公司，大量持有上海华信债的公司等），同时通过关联公司图谱识别与证券公司相关的持仓主体与关联方的风险程度，进而对证券公司的风险程度进行监测、预警。

# 第六章 证券行业系统性风险监测预警的建议

本文提供了利用深度学习、知识图谱对系统重要性证券公司风险状况监测预警的框架，通过该框架可构建系统重要性证券公司的风险监测预警体系及系统。监管机构防范系统性金融风险可考虑以下建议：

## 一、建立系统重要性证券公司的监测预警体系

监管机构应大力发展监管科技，加大监管科技投入，利用人工智能、知识图谱等先进技术建立系统重要性证券公司的监测预警体系，优化和改造监管、监察等业务，全面掌控和监测系统重要性证券公司的健康度指标，进而防范系统性金融风险的发生。

## 二、建立多方合作的风险监测预警体系

目前，国内金融业风险监测体系相对独立，未形成相互之间的风险监测联动机制，无法有效监控跨市场跨行业的风险传导。监管部门应建立相关的行业技术标准实现对金融机构、金融科技公司的金融数据标准化，推动数据治理工作的有效开展，加快完善大数据监管的基础设施建设，逐步建立适用于金融行业系统性金融风险预测的全国性监控系统，以便更好的监测防范系统性金融风险。

## 三、制定系统重要性证券公司评判标准

证券公司作为国内重要的金融机构，对于国家金融行业的稳定同样起到了至关重要的作用。监管当局应加快对系统性风险、系统重要性基础理论的研究，积极与国际金融组织进行对话并开展合作，借鉴SIFIs的评估方法，建立适合我国金融发展现状与证券行业特点的系统重要性证券公司的评估判定指标体系。

## 四、建立健全监管科技体系

随着金融科技的发展，金融行业呈现出多样化的业务模式和运行机制，跨界、综合经营的模式成为重大趋势。传统的监管方式已经不能满足现有的监管要求，需要建立更加健全和相对统一的监管体系。监管机构应借鉴国外成功监管经验，加强监管科技的发展与应用，完善金融科技监管标准，建立金融科技风险全覆盖的风险管理机制。定期与系统重要性金融机构和科技公司进行沟通对话，建立具有针对性的监管原则以及有效性的监管工具，完善风险处置机制，做好压力测试与应急预案，把握好金融科技创新与风险监管的平衡。

# 参考文献

[1] 方兴起. 防范系统性金融风险是金融监管的永恒主题[J]. [福建论坛(人文社会科学版)](http://xueshu.baidu.com/usercenter/data/journal?cmd=jump&wd=journaluri:(a5801a2d6a4a6a4f)%20%E3%80%8A%E7%A6%8F%E5%BB%BA%E8%AE%BA%E5%9D%9B(%E4%BA%BA%E6%96%87%E7%A4%BE%E4%BC%9A%E7%A7%91%E5%AD%A6%E7%89%88)%E3%80%8B&tn=SE_baiduxueshu_c1gjeupa&ie=utf-8&sc_f_para=sc_hilight=publish&sort=sc_cited)，2018（1）：12-18.

[2] 张珍. 论系统重要性金融机构的监管[D]. 厦门：厦门大学，2014.

[3] 张天顶，[张宇](http://www.cqvip.com/main/search.aspx?w=%e5%bc%a0%e5%ae%87). 宏观审慎监管、系统性金融风险及国内外金融监管实践及启示[J]. [证券市场导报，2018（4）](http://www.cqvip.com/QK/81567X/201804/)：61-68.

[4] 肖博达, 周国富. 人工智能技术发展及应用综述[J]. 福建电脑, 2018(1)：98-99+103.

[5] 王菲斐. 深度学习研究现状分析[J]. 电子技术与软件工程，2018（10）：152-153.

[6] 朱木易洁, 鲍秉坤, 徐常胜. 知识图谱发展与构建的研究进展[J]. 南京信息工程大学学报(自然科学版)，2017, 9(6)：575-582.

[7] Singhal A. Introducing the knowledgegraph: things, not strings[J]. Official google blog, 2012.

[8] Xu Zeng-lin, Sheng Yong-pan, He Li-rong, Wang Ya-fang. Review on Knowledge Graph Techniques[J]. Journal of University of Electronic Science and Technology of China, 2016, 45(4): 589-606.

[9] De Abreu, D., Flores, A., Palma, G., Pestana, V., Pinero, J., Queipo, J., ... & Vidal, M. E. Choosing Between Graph Databases and RDF Engines for Consuming and Mining Linked Data. In COLD.2003.

[10] Socher, R., Chen, D., Manning, C. D., & Ng, A. (2013). Reasoning with neural tensor networks for knowledge base completion. In Advances in Neural Information Processing Systems (pp. 926-934).

[11] Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., & Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. In Advances in Neural Information Processing Systems (pp. 2787-2795).

[12] Jenatton, R., Roux, N. L., Bordes, A., & Obozinski, G. R. (2012). A latent factor model for highly multi-relational data. In Advances in Neural Information Processing Systems (pp. 3167-3175).

[13] Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, Bo Xu (2016). Attention-Based Bidirectional Long Short-Term Memory Networks forRelation Classification.

[14] Chuanhai Dong, Jiajun Zhang, Chengqing Zong1(B), Masanori Hattori2,and Hui Di2(2016). Character-Based LSTM-CRF with Radical-LevelFeatures for Chinese Named Entity Recognition.

[15] Hongyu Lin, Yaojie Lu, Xianpei Han, Le Sun. Nugget Proposal Networks for Chinese Event Detection.

[16] Yubo Chen, Liheng Xu, Kang Liu, Daojian Zeng, andJun Zhao. 2015b. Event extraction via dynamic multi-pooling convolutional neural networks. InProceedings of ACL 2015.

[17] Xiaocheng Feng, Lifu Huang, Duyu Tang, Bing Qin, Heng Ji, and Ting Liu. 2016. A languageindependent neural network for event detection. In Proceedings of ACL 2016.

[18] Aher S B, Lobo L. Combination of machine learning algorithms for recommendation ofcourses in E-Learning System based on historical data[J], Knowledge-Based Systems, 2013, 51:1-14.

[19] 肖博达, 周国富. 人工智能技术发展及应用综述[J]. 福建电脑, 2018(1)：98-99+103.