文章编号: 1007-1423(2021)32-0109-05 DOI: 10.3969/j.issn.1007-1423.2021.32.021

基于金融知识图谱的互联网文本金融实体识别技术研究

胡庆锋

(里外(深圳)网络科技有限公司，深圳 518000)

摘要：在社交网络和网上论坛中，每时每刻都有新的信息在发布。如何利用金融知识图谱从这些非正式文本中及时准确的识别其中的金融实体，捕捉关键信息并辅助投资决策是人们关心的问题。本文研究了如何从Reddit网络论坛获得实时的金融讨论文本数据，通过实体识别模型，识别文本中蕴含的金融实体。针对互联网非正式文本存在着大量的不规范文本，包括名称缩写、简写、拼写错误等，我们构建了一个包含实体别名、简写与常见错误拼写的金融知识图谱，并训练了一个Albert(small) + CRF fine tune模型。在试验测试中，其模型的准确率，召回率都优于基准对比模型。另外模型的推断速度达到了5129 QPS(quest per second)，提升了金融实体识别的实时性，有利于快速找到金融决策信息。

关键词：金融知识图谱；金融实体识别；非正式文本；NER

0 引言

金融知识图谱是金融行业语义理解和知识搜索的基础技术，可以为风险评估、预测、反欺诈、精准营销、智能搜索等提供技术支撑。在金融知识图谱中，知识被表示为“事实”的集合，以(主语、谓语、宾语)三元组的形式表示，其中主语和宾语是实体，谓语是这些实体之间的关系。在金融知识图谱的构建过程中，金融实体的实时识别是其中的关键技术之一[1]。根据2020年美国股票市场算法自动交易的调查报告，约73%的股票交易是由计算机算法自动发出的交易决策。通过使用自然语言处理与金融知识图谱技术，计算机的交易模型可以在第一时间知道上市公司的年度报告或季度业绩报告中传递的关键信息，另外通过对互联网新闻或帖子的内容分析，计算机可以知道新闻中的事件传达出怎样的情绪，表达的情绪是积极的，消极的，或者中立的。这种情绪分析的结果再结合金融知识图谱中的关联信息可用于推理并再做出交易决策。

在社交网络和网上论坛中，每时每刻都有新的信息在发布。如何利用金融知识图谱从这些非正式文本中及时准确的识别其中的金融实体，捕捉关键信息并辅助投资决策是本文研究的重点。同时，互联网上存在着大量的非正式文本，例如一个句子“what r ya talking abt”，其正式的写法应该是“What are you talking about?”。这就为模型从非正式文本中提取信息提出了要求。非正式文本是指包含大量缩写和拼错单词的文本。因此，系统必须纠正这些词。系统应该能够实时生成非正式形式的文本以避免延迟问题。此外，在尝试生成语法正确的文本时，不应改变文本的真实含义，因为这可能会对后续的任务产生反作用。

1 互联网金融文本

1.1 金融文本数据

Reddit已成为社交媒体驱动型交易员和投资者的活动中心，这些市场参与者已经证明自己有能力影响市场，美国股市上的不少股票已经被掀

起了一场又一场的波动性风暴。典型例子如发生在2021年1月的游戏驿站轧空事件(或称游戏驿站空头挤压事件，指美国电子产品销售商游戏驿站(GameStop)股票发生持续轧空的现象，导致对部分对冲基金造成重大财务影响)。本文研究的社交文本数据来自于Reddit，虽然它似乎不太可能是这种金融事件的源头，但是这个论坛一直是互联网网民讨论金融信息的中心之一。

Reddit的很多子论坛是为投资者提供交流服务的，投资者可以自由的对市场上正在发生的事情进行大量讨论。人类已经无法实时阅读如此庞杂的信息，因此我们设计了信息抽取模型来处理这些数据。

要获取到Reddit的帖子文本数据，目前有两种方法。第一种方法我们可以使用 requests 库直接与 Reddit API 接口。第二种方法是使用 PRAW 库(Python Reddit API Wrapper)，它在访问 Reddit API 时添加了一个额外的抽象层。在这个项目中，我们将通过requests库直接与Reddit API 交互。

1.2 文本获取

```
import requests
import pandas as pd
class GetRedditData:
    def __init__(self, user_id, secret_token, username, password):
        auth = requests.auth.HTTPBasicAuth(user_id, secret_token)
        # 构建登陆Reddit所需要的信息
        login = {'grant_type': 'password',
                 'username': username,
                 'password': password}
        # 构图一个数据头部
        headers = {'User-Agent': 'MyBot/0.0.1'}
        # 发送一个请求去获得一个授权口令
        res = requests.post(f'https://www.reddit.com/api/v1/access_token', auth=auth, data=login, headers=headers)
        # 从返回数据中解析出授权口令
        token = res.json()['access_token']
        # 将口令信息加入数据头部中
        headers['Authorization'] = f'bearer {token}'
        self.headers = headers
        self.api = 'https://oauth.reddit.com'
    def get_news_text(self, subreddit, iters):
        # 初始化一个数据帧(pandas dataframe)来储存帖子数据
        df = pd.DataFrame()
        # 初始化一些参数，并放在一个字典中
        params = {'limit': 100}
        # 程序会循环尝试多次去获取数据，以保证数据获取成功
        for i in range(iters):
            # 发生数据获取请求
            res = requests.get(f'{self.api}/r/{subreddit}/new',
                               headers=self.headers,
                               params=params)
            #检查对方服务器返回的数据里面包含有帖子内容
            if len(res.json()['data']['children']) == 0:
                print('No more found')
                return df
            # 把每个跟帖的内容解析出来
            for thread in res.json()['data']['children']:
                # 将跟帖的内容储存到dataframe中
                df = df.append({
                    'id': thread['data']['name'],
                    'created_utc': int(thread['data']['created_utc']),
                    'subreddit': thread['data']['subreddit'],
                    'title': thread['data']['title'],
                    'selftext': thread['data']['selftext'],
                    'upvote_ratio': thread['data']['upvote_ratio'],
                    'ups': thread['data']['ups'],
                    'downs': thread['data']['downs'],
                    'score': thread['data']['score']
                }, ignore_index=True)
            # 找到最早的帖子的id号码
            earliest = df['id'].iloc[len(df)-1]
            # 将这个帖子的id号码单独存储，这样就可以找到帖子之间的先后关系
            params['after'] = earliest
        return df
```

2　金融实体识别模型

在构建这样的金融信息提取框架时，我们需要做的第一件事就是确定我们提取的数据实际上是关于什么的——为此我们将使用命名实体识别（named entity recognition）。

命名实体识别（named entity recognition）是自然语言处理中的基本任务之一。在本文中，金融知识图谱中的实体包括公司名字，监管机构，金融机构，投资基金，ETF（exchange-traded fund）等。要从互联网上庞杂的数据中获得想要的信息，第一步就是从各种正式或非正式的财经新闻或帖子中识别公司实体。对于风控业务而言，如果模型对某公司的负面信息在识别时没有召回，这种信息的遗漏可能会为业务造成巨大的损失，所以模型的召回率就非常关键。另一方面，对于计算机辅助投资决策或者算法自动交易的场景而言，实体识别的实时性也非常关键。互联网上的新闻内容是实时更新的，社交网络中的帖子内容也是实时更新的。所以对于以上金融业务场景，必须设计一种速度快，召回率高的金融实体识别模型。

随着BERT[3]等大型预训练语言模型的诞生，预训练模型和CRF的经典组合，成为业界最常见的NER模型。然而，由于BERT模型规模巨大，其模型参数多，预测速度慢、容易过拟合等缺点也引起了学术界的广泛关注。近两年，各种轻量级版本的BERT层出不穷，例如DistillBERT[8]、FastBERT[7]、DistillBERT[8]、Albert[5]、Electra[2]等等。它们具有与BERT-base相似的性能，但训练和预测速度提高了数倍。其中最有影响力的模型是由Google开源的Albert和Electra。

模型的问题是一方面，在实际金融知识图谱项目中，缺少高质量的标注数据是许多算法工程师面临的棘手问题。一些研究人员试图开发一些在训练数据稀疏的情况下提高训练效果的策略。例如J. Foley等人[4]探索将命名实体识别认为视为搜索任务。其中将感兴趣的命名实体类作为搜索关键词，而搜索出来的文档则是为包含该实体的语料库该类。他们还研究了如何构建了一些人工特征并基于NER-CRF模型将它们转换为搜索关键词。L. Chen等[6]提出了一种基于BERT的远程监督下的两阶段训练算法，从而避免去大量人工标注数据。

3　测试与试验

在社交网络文本中，存在着大量的不规范文本，包括名称缩写，简写，拼写错误等。例如苹果公司可能写为Apple Inc, Apple, APPL, APP等。为解决非正式文本中的金融实体识别问题，我们采用了字符串编辑距离测量，实体在上下文共现频率统计，实体名称在嵌入式空间的距离类聚等技术挖掘清洗了一批公司实体及其简写和常见错误拼写。构建了一个包含实体别名，简写与常见错误拼写的金融知识图谱。存储知识图谱常常需要一个图数据库，它使我们可以绕开传统关系型数据库的一些不便之处。在传统数据库中，当我们用它来处理大量包含互连关系的数据时，常常需要多表join操作，导致性能下降。在金融应用场景中，存在大量互联关系数据，如果我们使用传统数据库，就会面临应用层在获取数据是性能下降的问题。本文使用了Dgraph，它是一个开源图形数据库，为Web规模的生产环境构建，用Go编写。Dgraph数据库可水平扩展，同时保持操作高效以支持实时运行任意复杂的查询。这样我们就可以高效地执行分布式连接、过滤和排序这样的复杂问题。

基于以上金融知识图谱的数据，我们构建了一个字典树（trie-tree）对网络论坛上的金融讨论文本数据进行回标。总共的回标非正式金融文本^{句子}为95万。基于这些自动生成的训练数据，我们对Albert（small）- CRF模型进行了3轮fine tune训练。注意训练的轮次不宜太大，否则可能会产生过拟合问题。训练完成之后，我们事先构建的包含1万非正式金融文本的测试数据上对模型进行了测试。测试环境：4个TITAN Xp型号的GPU显卡；CPU：48核，内存：128 GB。通过试验结果我们可以发现，Albert（small）+Fine Tune模型的准

确率，召回率都优于基准对比模型。模型的推断速度达到了5129 QPS(quest per sencond)，提升了金融实体识别的实时性，有利于快速找到金融决策信息。

表1 在互联网非正式文本上进行金融实体识别的模型对比测试实验

模型	准确率	召回率	速度(quest per second)
Bert(base)+CRF	0.893	0.793	1016
Electra(small)	0.876	0.759	4986
Albert(small)	0.878	0.762	5138
Albert(small)+Fine Tune	0.927	0.936	5129

通过试验我们发现，采用基本的Bert(base)模型加上一层CRF，在互联网非正式文本上即可用取的还不错的识别效果，准确率89%，召回率79%。但是这种方法的识别速度是一个性能瓶颈，平均每秒钟能处理1016个环联网非正式金融文本句子。为了减少模型的参数数量，减少模型训练时间和推理时间，我们采用了两种参数减少技术。第一种技术是分解嵌入参数化(factorized embedding parameterization)。通过分解将大的词汇嵌入矩阵分成两个小矩阵，我们将隐层的层数与词嵌入句子的大小的解耦了。这种分离使我们可以很容易的增加隐层的层数，不会显著增加词嵌入的参数数量。第二种技术是跨层参数共享。这种技术可以防止参数增长随着网络的深度。这两种技术都显着减少了BERT模型参数的数量，而且不会严重明显影响准确性，从而提高参数效率。试验中我们发现推理速度从1016提高到了5000左右，速度提升了5倍，准确率只下降了2个百分点。其中艾伯特Albert(small)+Fine Tune训练到方法在互联网金融文本中的准确率为92.7%，召回率93.6%，推理速度为每秒钟5129个句子文本。

4 结语

本文给出了如何从Reddit网络论坛获得实时的金融讨论文本数据，通过实体识别模型，识别文本中蕴含的金融实体。针对互联网非正式文本存在着大量的不规范文本，包括名称缩写、简写、拼写错误等，我们构建了一个包含实体别名，简写与常见错误拼写的金融知识图谱，并训练了一个Albert (small) - CRF fine tune模型。在试验测试中，其模型的准确率，召回率都优于基准对比模型。另外模型的推断速度达到了5129 QPS(quest per sencond)，提升了金融实体识别的实时性。

参考文献：

[1] ZHANG H. Improving NER's Performance with Massive financial corpus[J],2020.

[2] CLARK K,LUONG M T,LE Q V,et al. Electra:pre-training text encoders as discriminators rather than generators[J],2020.

[3] DEVLIN J,CHANG M W,LEE K,et al. Bert:Pre-training of deep bidirectional transformers for language understanding[J],2018.

[4] FOLEY J,SARWAR S M,ALLAN J. Named entity recognition with extremely limited data[J],2018.

[5] LAN Z Z,CHEN M D,GOODMAN S,et al. Albert:a lite BERT for self-supervised learning of language representations[J],2019.

[6] LIANG C,YU Y,JIANG H M,et al. Bond:Bert-assisted open-domain named entity recognition with distant supervision[J]. ACM,2020.

[7] LIU W J,ZHOU P,ZHAO Z,et al. FastBERT:a self-distilling BERT with adaptive inference time[C]//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics,2020.

[8] SANH V,DEBUT L,CHAUMOND J,et al. DistilBERT,a distilled version of BERT:smaller,faster,cheaper and lighter[J],2019.

作者简介：

胡庆锋(1985—),男,广东深圳人,本科,算法工程师,研究方向为自然语言处理

收稿日期：2021-09-05 **修稿日期：**2021-10-13

(下转第120页)

Research on the Design of Smart Tourism Recommendation System Based on Big Data Mining Technology

Li Wei

(School of Mathematics and Computer Science, Liupanshui Normal University, Liupanshui 553000)

Abstract: With the advent of the era of big data, tourism has begun to shift to smart tourism. Tourism service information recommendation system uses big data technology to strengthen the connection between tourists and tourism information, which plays an important role in smart tourism. Against the traditional tourist information services, research using data mining technology, improve the collaborative filtering algorithm based on the project, analysis of the historical data of visitors and tourists for visitors interested in interaction with the system behavior, use the Mahout implementation distributed intelligent recommendation system, provide passengers with efficient personalized information recommendation service.

Keywords: intelligent tourism; recommendation system; collaborative filtering

(上接第112页)

Internet Text Financial Entity Recognition Technology Based on Financial Knowledge Graph

Hu Qingfeng

(Liwai(Shenzhen)Network Technology Co., LTD, Shenzhen 518000)

Abstract: In social networks and online forums, new information is released every moment. How to use the financial knowledge graph to identify the financial entities in these informal texts in a timely and accurate manner, capture key information and assist in investment decision-making is a problem that people are concerned about. This article studies how to obtain real-time financial discussion text data from the Reddit network forum, and use entity recognition models to identify financial entities contained in the text. In view of the large number of irregular texts in the Internet informal texts, including name abbreviations, abbreviations, spelling errors, etc., we constructed a financial knowledge graph containing entity aliases, abbreviations and common misspellings, and trained an Albert (small)- CRF fine tune model. In the experimental test, the accuracy and recall rate of the model are better than the benchmark comparison model. In addition, the inference speed of the model has reached 5129 QPS (quest per second), which improves the real-time performance of financial entity recognition and is conducive to quickly finding financial decision information.

Keywords: financial knowledge graph; informal text; named entity recognise