目 录

1 概述

我们小组的糖尿病知识图谱任务各部分概述如下：

- 在本体构建部分，我们针对任务特点和获取到的先验知识，基于 Protege 构建了适配于本项目需求的知识图谱。
- 在命名实体识别部分，主要采用 GlobalPointer 框架，对 PDF 进行字符抽取，对抽取后的文本进行命名实体的序列标记。
- 在进一步的表格信息抽取、消歧部分，采用 clueAI 模型提取药品，对药品、症状都进行编码处理，并有相应的对应关系及解编码器。
- 在推荐模型部分，我们尝试了两种模型：基于多标签分类的推荐模型和基于随机游走的推荐模型。对于两类模型我们给出了详细的建模思路以及实验结果并对结果进行了仔细的分析。

2 本体构建和关系抽取

本体的选择与构建是知识图谱的底层基础，我们主要基于 Protege 和已给出的 excel（内分泌降糖药物和糖尿病病人住院数据）进行构建。这里的介绍顺序和实际任务实现过程中的步骤顺序保持一致。

2.1 本体策划与构建

我们主要采取针对任务需求进行构建的策略。总体上，相比于往届学长工作的本体结构（下图 1），我们的较为简略（下图 2）：

- 考虑到需要串联个体信息和疾病诊断、用药，所以设计了 id 类，用具体 id 表示个体。
- 将诸如“入院体重”、“用药”等都设置为数据属性便于后续的图搜索。

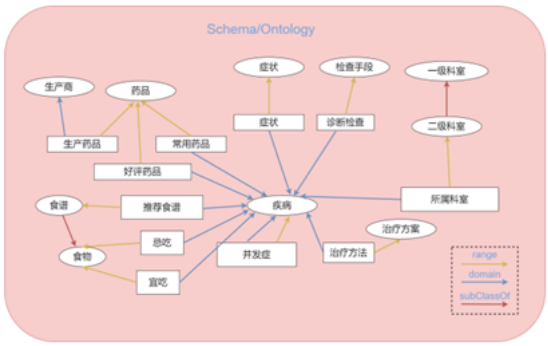


图 1: 往届工作的本体结构

在药物本体方面，我们搜索了关于胰岛素的先验知识：基础胰岛素在任何时候都可以注射，而短中长效胰岛素有注射时间要求（如起效快的饭前注射），所以据此进行了分类，如图 3。

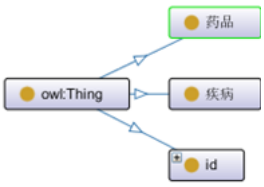


图 2: 简略版本体结构

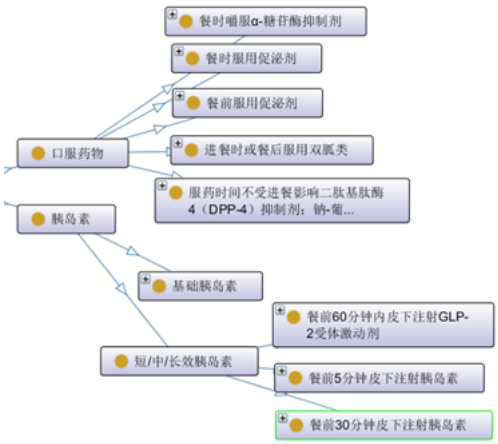


图 3: 药物本体结构

2.2 数据属性选取

2.2.1 id 数据属性选取

在 id 数据属性选取方面，经过分析，我们的任务主要面临以下问题：

- 相较于数据量，疾病诊断种类（组合）较多。
- 症状和用药之间存在隐式对应规则。
- 病人个体指标较为稀疏/缺少强关联性，需要筛选重要指标。

第一点主要是在推荐模型部分解决，这里主要针对后两个问题，我们在选取 id 数据属性时向某三甲医院内分泌科主任进行了咨询（如下图 4），了解到了一些先验的规则知识，特别的如“1 型糖尿病都用胰岛素”以及体重指数的重要意义等，因此最终选定了以下 id 属性（图 5）。

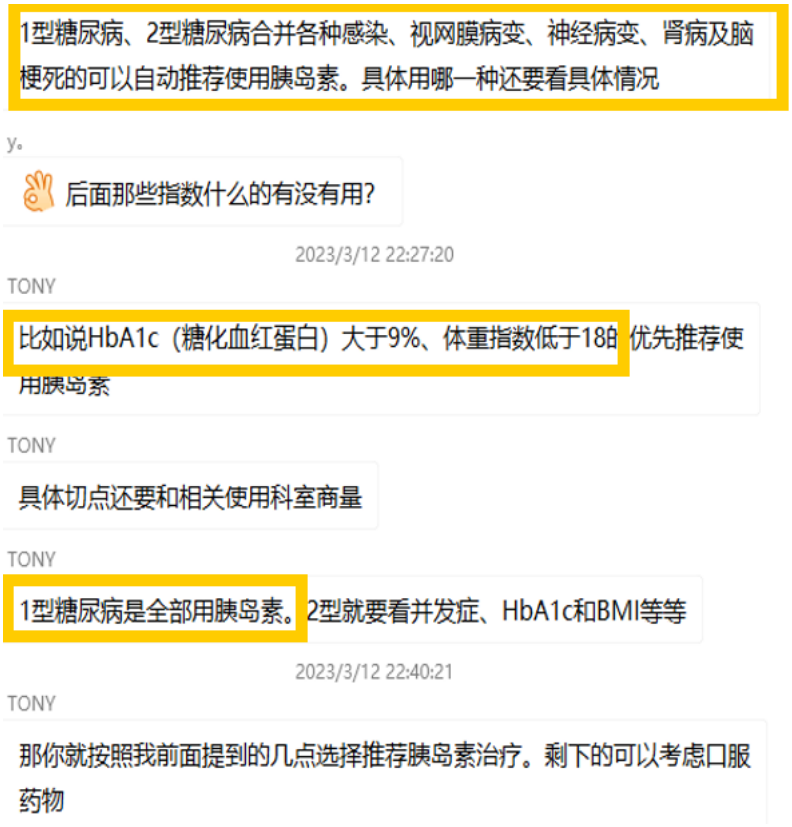


图 4: 向某三甲医院内分泌科主任咨询

图 5: id 属性

2.2.2 药物数据属性选取

主要基于任务已给出的 excel 表格，选取以下属性作为某具体药物的属性（图 6）。值得一提的是，考虑到“注意事项”中有孕妇相关，所以在 id 数据属性中添加了“妊娠”这一属性。

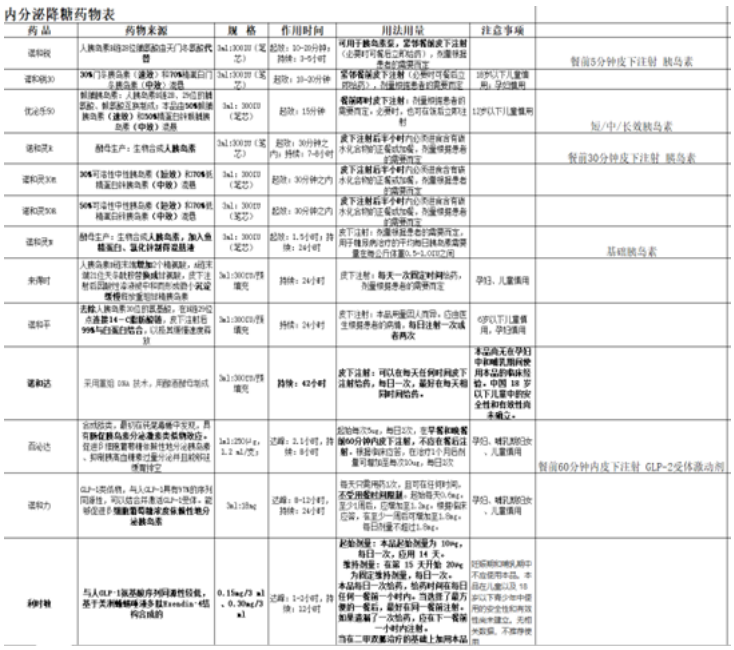
图 6: 药物属性

2.3 表格数据清洗和初步带药抽取

在关系抽取前，我们在原始数据方面遇到了以下问题：

- 部分行住院带药缺失。
- 降糖药物数据错位；sheet1 和 2 信息混杂。
- 出现普遍的同音不同字药物情况（拜糖平？拜糖苹？拜唐苹?）。

这里对问题 1 和 2 进行了手动删除和调整，清洗后的结果如下：


内分泌降糖药物表

药品	药物来源	规格	作用时间	用法用量	注意事项	
诺和锐	人胰岛素和胰岛素门冬[门冬氨酸]的	3ml:300IU（笔芯）	起效：10-20分钟，持续：3-5小时	可用于胰岛素泵，紧急餐前皮下注射（必要时可餐后立即给药），剂量根据患者血糖需调整		餐前5分钟皮下注射 胰岛素
诺和锐30	30%门冬胰岛素（速效）和70%精蛋白门冬胰岛素（中效）混悬	3ml:300IU（笔芯）	起效：10-20分钟	紧急餐前皮下注射（必要时可餐后立即给药），剂量根据患者血糖需调整	18岁以下儿童慎用，孕妇慎用	
优泌乐50	新型胰岛素：人胰岛素同类2B、25位的赖氨酸、赖脯酸互换而成，本品由50%赖脯胰岛素（速效）和50%精蛋白锌赖脯胰岛素（中效）混合	3ml：300IU（笔芯）	起效：15分钟	餐前即时皮下注射：剂量根据患者的需要而定，必要时，也可在饭后立即注射	12岁以下儿童慎用	短/中/长效胰岛素
诺和灵R	制作生产：生物合成人胰岛素	3ml:300IU（笔芯）	起效：30分钟之内；持续：7-8小时	皮下注射后半小时内应不进食含有碳水化合物的正餐或加餐，剂量根据患者的需要而定		餐前30分钟皮下注射 胰岛素
诺和灵30R	30%可溶性中性胰岛素（胰岛）和70%低精蛋白锌胰岛素（中效）混悬	3ml：300IU（笔芯）	起效：30分钟之内	皮下注射后半小时内应不进食含有碳水化合物的正餐或加餐，剂量根据患者的需要而定		
诺和灵50R	50%可溶性中性胰岛素（胰岛）和50%低精蛋白锌胰岛素（中效）混悬	3ml：300IU（笔芯）	起效：30分钟之内	皮下注射后半小时内应不进食含有碳水化合物的正餐或加餐，剂量根据患者的需要而定		
诺和灵N	制作生产：生物合成人胰岛素，加入鱼精蛋白、氯化锌制得混悬液	3ml：300IU（笔芯）	起效：1.5小时；持续：24小时	皮下注射：剂量根据患者的需要而定，用于糖尿病治疗的平均每日胰岛素需要量范围公开注量0.5-1.0IU/公斤		基础胰岛素
来得时	人胰岛素和短效胰岛素[加2个精氨酸，4位天冬酰胺已天冬氨酸被赖氨酸，皮下注射后因酸性溶液中和而形成微沉淀，从而使作用时间延长]的混合物	3ml:300IU/预填充	持续：24小时	皮下注射：每天一次固定时间给药，剂量根据患者的需要而定	孕妇、儿童慎用	
诺和平	去除人胰岛素30位的苏氨酸，在N端29位点连接14-C脂肪酸链，皮下注射后99%与白蛋白结合，以极其缓慢速度释放	3ml:300IU/预填充	持续：24小时	皮下注射：本品用餐同人有异，应由医生根据患者的资质，每日注射一次或者两次	6岁以下儿童慎用，孕妇慎用	
诺和达	采用重组 DNA 技术，用酵母菌作生产菌	3ml:300IU/预填充	持续：42小时	皮下注射：可以在每天任何时间皮下注射给药，每日一次，最好在每天相同时间给药。	本品尚无在孕妇中和哺乳期间使用本品的临床经验。中国 18 岁以下儿童中的安全性和有效性尚未确立。	
百泌达	合成肽类，最初从怪异毒蜥中发现，具有胰高血糖素样多重生物活性，在进行性胰腺葡萄糖非耐性地对诱胰岛素、抑制胰高血糖素过量分泌并且能够延缓胃排空	1ml:250μg，1.2 ml/支	达峰：2.1小时，持续：8小时	起始每次5μg，每日2次，皮下餐前晚餐前60分钟内皮下注射，不给在餐后注射。根据临床应答，在治疗1个月后剂量可增加至每次10μg，每日2次	孕妇、哺乳期妇女、儿童慎用	餐前60分钟内皮下注射 GLP-2受体激动剂
诺和力	GLP-1类似物，与人GLP-1具有97%的序列同源性，可以结合并激活GLP-1受体，能够促进β细胞葡萄糖依赖性地分泌胰岛素	3ml:18mg	达峰：8-12小时，持续：24小时	每天只需用药1次，且可在任何时间，不受用餐时间限制。起始每天0.6mg，至少1周后，应增加至1.2mg，根据临床应答，在至少一周后可增加至1.8mg，每日剂量不超过1.8mg。	孕妇、哺乳期妇女、儿童慎用	
利拉鲁	与人GLP-1依基酸序列同源性较低，基于艾美酶精神通多肽exendin-4结构合成的	0.15mg/3 ml、0.30mg/3 ml	达峰：1-3小时，持续：12小时	起始剂量：本品起始剂量为 10mg，每日一次，连用 14 天。维持剂量：在第 15 天开始 20mg 为稳定维持剂量，每日一次。本品每日一次给药，给药时间在每日任何一餐前一个小时内。当患者了最方便的一餐后，最好在同一餐前注射。如果遗漏了一次给药，应在下一餐前一个小时内注射。当在二甲双胍治疗的基础上加用本品	妊娠期间哺乳期中不应使用本品。本品在儿童以及 18 岁以下青少年中使用的安全性和有效性尚未建立。无相关数据，不推荐使用	

图 7: 表格数据清洗

对于问题 3，考虑到知识图谱的完整性和为后续的模型抽取提供可靠的补充信息，这里针对已给出的降糖药物进行了初步抽取（到表格中）。由于范围已给出，所以采用原始的正则表达式方法，可以准确地识别同名不同字实体。代码和结果如下：

2.4 构建结果展示

在使用 Protege 进行表格关系抽取时遇到了数据量过大的问题，所以每次只添加单个 rule 进行抽取；构建结果如下所示：

```
col_name=df.columns.tolist()#将列名全部提取出来存放在列表里
col_name.append('0')#将新增的列添加到最后
col_name.append('1')#将新增的列添加到最后
col_name.append('2')#将新增的列添加到最后
col_name.append('3')#将新增的列添加到最后
col_name.append('4')#将新增的列添加到最后
col_name.append('5')#将新增的列添加到最后
dr=df.reindex(columns=col_name)
for i in range(0,3597):
    s=df.iloc[i,6]
    name = re.findall(r"诺和锐|优泌乐50|诺和锐30|诺和灵R|诺和灵30R|诺和灵50R|诺
    for j in range(len(name)):
        df.loc[i,j+90]=name[j] #把新列的数据放到指定的列名下
df.to_excel('./y_糖尿病病人住院数据.xlsx')#将整个dr写入excel
```

图 8: 正则表达式抽取

1	格华止	拜糖苹	捷诺维	亚莫利	格华止
0	诺和锐	诺和平	格华止		
0	格华止				
0	诺和锐				
0	百泌达	格华止			
0	诺和锐	来得时			
1	来得时	格华止	诺和龙		
0					
0	诺和锐				
1	诺和锐	格华止	拜糖平		
1	捷诺维	亚莫利			
0					
0	诺和锐	诺和平	拜唐苹	格华止	
1					
1	格华止	诺和锐	诺和平		
0					
1	达美康缓释				
0	诺和锐	诺和平	格华止	拜唐苹	
0	优泌乐50	拜唐苹	格华止		
0	格华止	捷诺维			
1	来得时	拜唐苹			
0	捷诺维				
0	诺和锐	来得时	拜唐苹		

图 9: 初步抽取结果

3 命名实体识别和关系抽取

3.1 命名实体识别

在这项工作中，我们将其视为 NER，即命名实体识别任务。具体步骤包括对 PDF 进行字符抽取，对抽取后的文本进行命名实体的序列标记。因为需要识别的命名实体是给定的，因为我们得以将其转化为序列标记任务。在模型选择上，我们选择了 GlobalPointer 框架来进行 NER 任务。

3.1.1 模型框架：GlobalPointer

我们都知道名称实体识别的任务可以被认为是序列标记问题。一般来说，我们使用 BER-LSTM-CRF 模型来解决这个问题。但 CRF 模型不能并行运行，速度较慢。所以我们使用另一个模型，由苏剑林介绍的 GlobalPointer。

GlobalPointer 将问题转化为多标签分类问题。具体来说，假设要识别文本序列长度为 n，简单起见先假定只有一种实体要识别，并且假定每个待识别实体是该序列的一个连续片段，长度不限，并且可以相互嵌套（两个实体之间有交集），长度为 n 的序列有 n(n+1)/2 个不同的连续子序列，这些子序列包含了所有可能的实体，而我们要做的就是从这 n(n+1)/2 个“候选实体”里边挑出真正的实体，其实就是一个“n(n+1)/2 选 k”的多标签分类问题。如果有 m 种实体类型需要识别，那么就做成 m 个“n(n+1)/2 选 k”的多标签分类问题。这就是 GlobalPointer 的基本思想，以实体为基本单位进行判别。

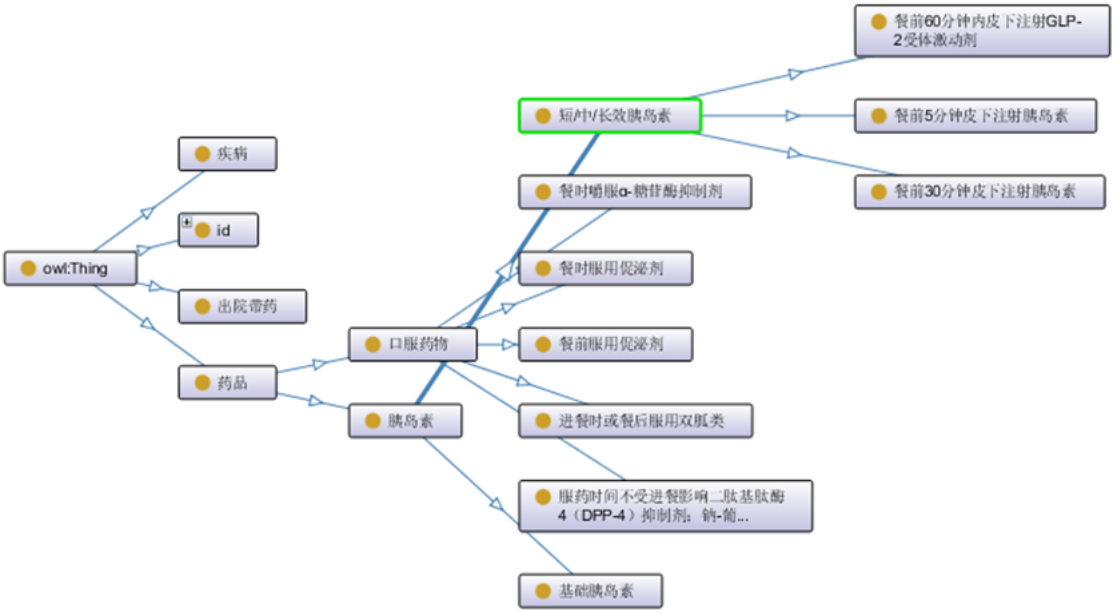


图 10: 本体构建结果

将问题转化为多标签分类问题后，我们可以构造阶数为 n 的方阵，n 是句子的最大长度。当跨度是一个实体时，我们只在相应的位置标记 1。当我们的模型输出相似矩阵时，我们可以计算 softmax。

3.1.2 整体模型框架

下图是我们整体的模型框架。我们这里是将句子都进行 bert 编码，再通过一个线性层构造出初始的 query 和 key 向量，然后进行旋转位置编码，最后计算多头注意力得到相似度矩阵。然后我们引入了 span level 的监督信息，以进行多标签 softmax 的 loss 的计算。

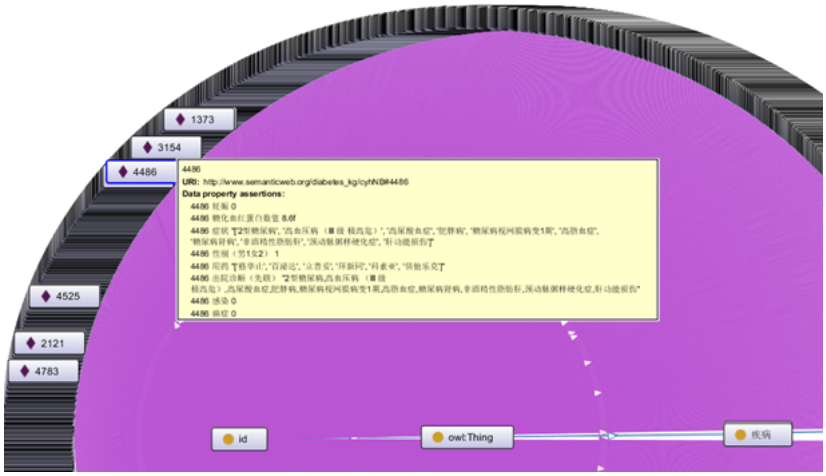


图 11: 实例层结果

图 12: GlobalPointer 模型

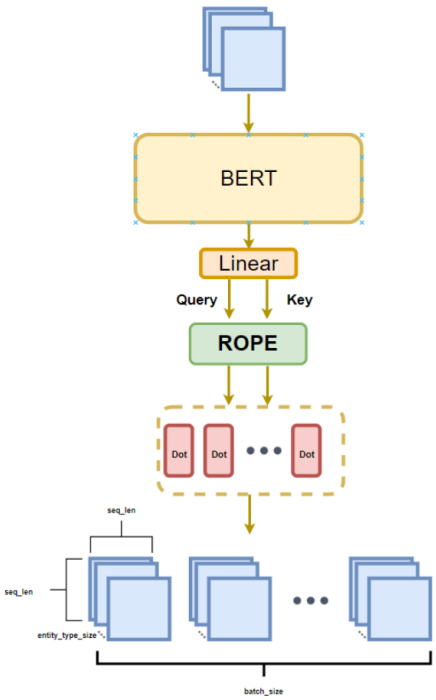


图 13: NER 模型

3.2 多标签 Softmax Loss

在获得有关标签的输出矩阵后，我们需要一个恰当的损失函数以进行优化。因为 GlobalPointer 将序列标注问题转化为多标签分类，因此在损失函数设计方面一个很直观的思路便是将其分为 n（n+1）/ 2 个二分类。然而实际使用时 n 往往并不小，那么 n（n+1）/ 2 更大，而每个句子的实体数不会很多（每一类的实体数目往往只是个位数），所以如果是 n（n+1）/ 2 个二分类的话，会带来极其严重的类别不均衡问题。

为了解决这个问题，我们引入了多标签 softmax loss，它是单目标多分类交叉熵的推广，特别适合总类别数很大、目标类别数较小的多标签分类问题。其形式也不复杂，在 GlobalPointer 的场景，它的表达式如下所示

$$J = \log\left(1 + \sum_{(i,j)\in P_a} e^{-s_a(i,j)}\right) + \log\left(1 + \sum_{(i,j)\in Q_a} e^{s_a(i,j)}\right)$$

其中 Pa 是该样本的所有类型为 a 的实体的首尾集合，Qa 是该样本的所有非实体或者类型非 a 的实体的首尾集合，注意我们只需要考虑 i　j 的组合，因为头尾是有顺序的，我们只需要观察头在前尾在后的组合。

4 表格信息抽取、消歧

这一部分主要由张屹灵和游皓月同学负责，完成从所给表格中提取我们需要的信息，包括每个病人对应症状、对应用药以及药品间的关系等。然后基于原始提取数据进行消歧操作，便于后续知识图谱构建。最后将得到的几个 csv 表格导入到 Neo4j 中，对知识图谱进行可视化。

4.1 表格信息抽取

我们观察了“糖尿病病人住院数据.xlsx”表格文件每列数据特征，认为需要重点处理“出院带药”和“出院诊断”两列。目标是将它们格式化方便下一步消歧编码。

4.1.1 特征观察

“出院带药”列本身是文字段落，大部分药品出现在句首，后接用药剂量与用药时间。数据间格式差异和语义差异成正比，具有一定规律，如下图所示。只要提取结果够好，后续处理能十分方便，因此对分词提取模型要求较高。

“出院诊断”列本身是数组形式的字符串，但包含许多格式异常的数据。数据间格式差异和语义差异不成正比，可能对后续对齐造成干扰，需要一些人为的定义限制与处理。

4.1.2 信息提取

“出院带药”列使用 clueAI 模型 [?]，选择其中的“医疗信息抽取”模块，初步提取药物信息。观察提取结果发现，许多药品拥有别名，因为其涉及样本数不多，为确保精确性，我们选择人工校对，记录别

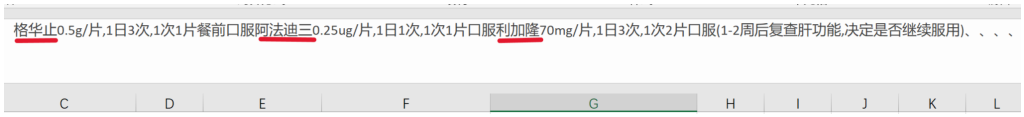


图 14: 出院带药数据格式示例

名的同时反馈一些提取不到位的数据，在下一步清理函数中进行相应处理。别名制作成“药品-别名.xlsx”表格，为之后消歧做准备。

	A	B	C	D	E
1		全名	别名1	别名2	别名3
2		阿法骨化醇软胶囊	阿法迪三		
3		复合维生素片	爱乐维		
4		地奥司明片	爱脉朗		
5		厄贝沙坦片	安博维		
6		达格列净	安达唐		
7		沙格列汀(片)	安立泽		
8		奥美拉唑肠溶胶囊	奥克		
9		艾塞那肽注射液	百泌达	卡博平	
10		多环醇片	百赛诺	双环醇片	
11		阿司匹林肠溶片	拜阿司匹灵	拜阿司匹林	
12		利伐沙班	拜瑞妥		
13		阿卡波糖片	拜糖平	拜唐苹	卡波糖片
14		硝苯地平控释片	拜新同		
15		盐酸氨溴索片	贝莱		
16		缬沙坦氢氯地平片	倍博特		
17		琥珀酸美托洛尔缓释片	倍他乐克	倍他乐克缓释片	
18		硫酸氢氯吡格雷片	波立维		
19		雷贝拉唑缓释片	波利特		
20		硫酸氢氯吡格雷	波利维		
21		(盐酸)二甲双胍缓释片	卜可		
22		格列齐特缓释片	达美康		
23		缬沙坦胶囊	代文	缬克	
24		羟苯磺酸钙胶囊	导升明		
25		胰酶肠溶胶囊	得美通		
26		匹维溴铵片	得舒特		
27		乳果糖口服液	杜密克		
28		苯溴马隆片	尔同舒		
29		瑞格列奈片	孚来迪		
30		阿伦磷酸钠(片)	福善美		
31		缬沙坦氢氯噻嗪片	复代文		
32		复方磺胺甲噁唑	复方新诺明		
33		碳酸钙D3片	钙尔奇		
34		骨化三醇(软)胶囊	盖三淳		
35		(盐酸)二甲双胍片	格华止	二甲双胍	

图 15: 药物-别名关系表部分

clueAI 模型是 CLUE 中文测评社区发布的一个神器，接口全部封装，三分钟就能构建完 NLP 模型的 API，可以输入不同的中文提示来支持不同类型的任务。我们尝试将其融入本项目时，本地部署模型效果不是很好，最后还是选择了调用线上 API。

“出院诊断”列可将字符串转数组，通过数组操作进行筛选处理，将异常格式（如符号异常等）删除或修正。

4.1.3 数据清洗

对“出院带药”列进行特殊字符的清洗（如删除剂量相关的“u”，时间相关的“餐前”、“早晚”等）。

对“出院诊断”列进行进一步语义统一（如所有与手术有关数据统一为“术后”），对具体数据进行模糊处理（如“1-2 节脊椎”、“3-4 节脊椎”统一为“脊椎”）。

二者处理完后分别进行去重、排序，输出为药品、症状表格，作为中间文件待下一步消歧处理。

```python
import clueai

# initialize the Clueai Client with an API Key
cl = clueai.Client("", check_api_key=False)
prompt= '''
信息抽取:
格华止0.5g/片,1日3次,1次1片餐前口服阿法迪三0.25ug/片,1日1次,1次1片口服利加隆70mg/片,1日3次,1次2片口服(1-2周后复查肝功能,决定是否继续服用)、、、、
问题: 药名#剂量
答案:
'''
prediction = cl.generate(
        model_name='clueai-base',
        prompt=prompt)
# 需要返回得分的话，指定return_likelihoods="GENERATION"

# print the predicted text
print('{}'.format(prediction.generations[0].text))
```

```
药名: 格华止，阿法迪三，利加隆
剂量: 0.5g/片，0.25ug/片，70mg/片
```

图 16: 调用 ClueAI 模型 API 部分代码实现

4.2 数据消歧

4.2.1 药品消歧

选取 Jaccard 相似度对药品每一项进行聚类、编号，它实现比较方便，效果也很好。一轮消歧后，根据先前人工制作的“药品-别名.xlsx”表格进行对齐，再次编号，输出为“Drug.xlsx”。最后与先前的 PDF 处理结果进行对齐。

在相似度计算过程中，我们认为若对每一项均计算和所有其他项的相似度，计算量将会非常大。考虑到前面分析药名数据的形式差异和语义差异成正比，我们选择将数据排好序后，只选取前后三项进行相似度计算。这样不会对结果造成太大影响，同时大幅度节省了时间。

同时，结合前一部分同学从 PDF 中提取出的相关信息和“内分泌降糖药物.xls”表格，对部分药品进行对齐，并进行归类处理，为后续知识图谱中“属于”关系的建立提供数据基础。

4.2.2 症状消歧

我们尝试过使用同样的方式聚类，但是效果很差，我们对此进行了原因分析，认为这是数据本身含义的问题：细微表述差别会对应不同症状、细微症状差别又会对应不同药品。若完全由人工处理工作量将会很大，且之后的编码对应等工作难以进行。最后我们的处理方式是设了一个很高的阈值，保留了几乎全部数据。

得到症状表后，再结合病人 ID，创建“Disease_id_path.xlsx”表格，将病人和症状结合起来，方便之后知识图谱的构建。

4.3 解编码器

根据以上步骤得到的编码表，我们编写了相应的编码器和解码器，在后续的处理中，输入数据经过这两个编码器编码，经过我们的模型得到药物编码后再解码成文字返回。其中涉及的主要问题是，针对 UNK 情况，我们根据 jaccard 相似度，返回相似度最大的编码。

4.4 表格信息抽取及消歧流程图

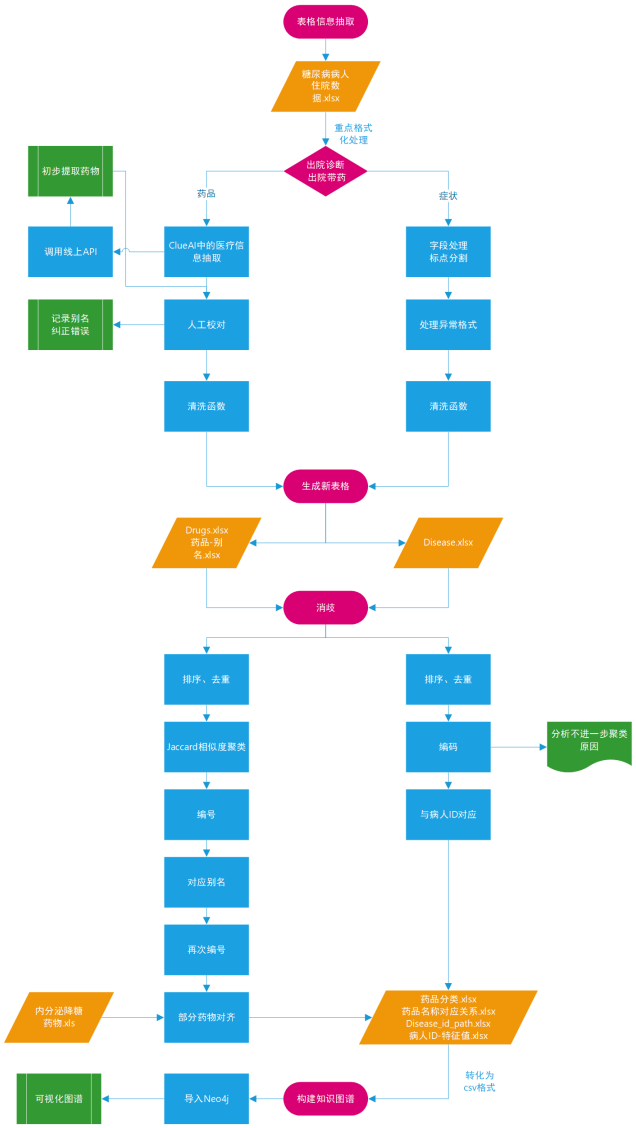


图 17: 表格信息抽取及消歧流程图

4.5 总结和思考

4.5.1 工作亮点

- 采用 clueAI 模型提取药品，简化工作量的同时提高了结果质量。
- 药品别名的处理能让结果更加精确，便于消歧和知识对齐。
- 对药品、症状都进行编码处理，并有相应的对应关系及解编码器，使处理简单的同时可以以一种简便的形式显示结果。

4.5.2 工作难点及改进展望

- 症状数据的干扰项太多，函数处理无法涉及所有方面，需人工处理。
- 症状和药品之间更直接的对应关系难以准确提取，缺乏相关专业知识，原始数据也未具体给出一一

对应关系，这对之后模型预测准确率有很大影响。

我们讨论并提出了一些改进方案。针对此项目而言，我们可以结合更多更细致的数据库，同时可以征求相关专业人士提供医疗相关知识帮助，排除非糖尿病并发症的疾病，从而建立更符合题目“糖尿病”的专业知识图谱。而再进一步的精确对齐可能需要更高级的语言模型来进行语义匹配，而不是单纯停留在字符串的匹配上。

5 知识图谱构建及可视化

根据之前本体构建、信息抽取等工作，基于得到的表格转化为 csv 文件，导入 Neo4j，实现知识图谱可视化。

5.1 知识图谱架构

- **Labels**：病人、症状、药品
- **Relationship**：使用、属于、患（病）、治疗、禁忌
- **Property Keys**

图 18: 知识图谱架构

5.2 知识图谱部分截图展示

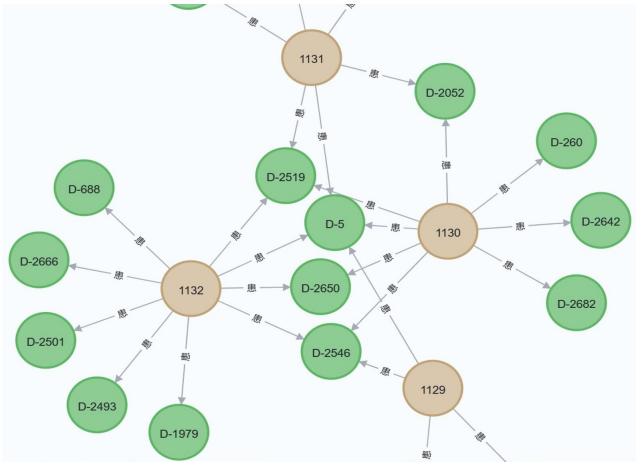


图 19: 患【病人-症状】

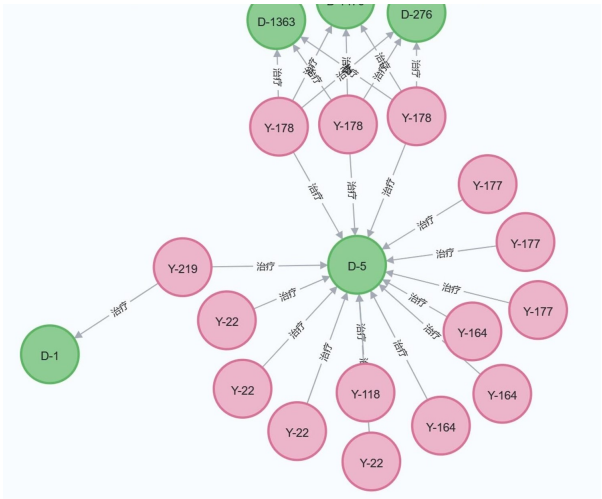


图 20: 治疗【症状-药品】

6 推荐模型

6.1 方案一

6.1.1 预测思路

我们将每个病人的部分信息以及症状作为特征，每个病人的用药作为标签，输入到预测模型中进行训练。

针对特征的输入，我们将症状编码成 01 向量嵌入到特征向量中，从而得到一个一维的嵌入向量，作为输入的特征；针对标签的输入，我们将药物编码成 01 向量，因此变成多标签分类问题。

6.1.2 预测模型

MLP 是解决单标签分类问题的一种常用办法，因而我们决定采用多个 MLP 进行多标签分类学习，最后利用知识图谱筛选掉存在禁忌关系的药物。

6.1.3 预测思路

在模型评估方面，我们选取了三个评价指标，精度（precision），召回率（recall），F1 分数。

精度表示预测正确的药物个数占所有预测药物个数的百分比，其公式如下：

$$Precision = \frac{|PredictionSet \cap ReferenceSet|}{|PredictionSet|} \tag{1}$$

召回率表示预测正确的药物个数占所有实际药物个数的百分比，其公式如下：

$$Recall = \frac{|PredictionSet \cap ReferenceSet|}{|ReferenceSet|} \tag{2}$$

F1 表示精度和召回率的调和平均，其公式如下：

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \tag{3}$$

其中，PredictionSet 为算法预测的处方药物集合，ReferenceSet 为真实的处方药物集合。最终针对所有病例计算平均 Precision，平均 Recall，平均 F1 值。

6.2 方案二

6.2.1 问题建模

这一部分将处方推荐的问题建模成模型上的链接预测任务。具体来说，令 $G = (V, E)$ 是一个有着 $|V|$ 个节点，$|E|$ 条边的无向图，图中。首先我们先学习节点的嵌入函数 $g(\cdot)$

$$g : \mathbb{R}^{|V| \times n} \to \mathbb{R}^N \tag{4}$$

然后我们构建一个回归模型

$$f : \mathbb{R}^N \times \mathbb{R}^N \to [0, 1] \tag{5}$$

模型输入是两个节点的嵌入表示，输出是这两个节点之间存在节点的概率。然后根据回归模型得到处方

$$\begin{gathered} Prescription = \{p_j \mid f(g(p_i), g(p_j)) > \alpha,\ p_i \in V, class(p_i) = person, class(p_j) = medicine\}, \\ \alpha \in (0, 1) \end{gathered} \tag{6}$$

其中，$class(\cdot)$ 返回节点的类别，α 是一个超参数。当 p_i 满足 $f(g(p_i), g(p_j)) > \alpha$ 时说明两节点直接存在链接。

6.2.2 节点嵌入

经过调研选择论文《A modified DeepWalk method for link prediction in attributed social network》中的方法获取节点的表示。虽然这篇论文发在一篇不知名期刊上，但是谷歌学术上的引用次数达到了 52 次。而且论文中使用的数据集的节点数和我们知识图谱的节点数相当并取得了相当好的甚至超越 GraphSAGE 的表现。

这篇文章提出的目的是解决属性网络中的链接预测问题。属性网络中的链接预测是近年来的热门话题之一。在许多现实世界的系统中，节点还伴随着各种属性或特征，称为属性网络。链接预测的最新方法之一是嵌入方法来生成图中每个节点的特征向量并找到未知连接。DeepWalk 算法是最流行的图嵌入方法之一，它使用纯随机游走捕获网络结构。论文旨在提出一种基于纯随机游走的深度游走的修改版本，用于解决属性网络中的链接预测问题，它将用于网络结构和节点属性，并将引入用于链接预测的新随机游走模型通过整合网络结构和节点属性，基于假设网络上的两个节点由于在网络中相邻或由于相似属性而连接。结果表明，在拥有更多结构和属性相似性的情况下，两个节点更有可能建立链接。为了证明该提议的合理性，作者在六个真实世界的属性网络上进行了许多实验，以与最先进的网络嵌入方法进行比较。

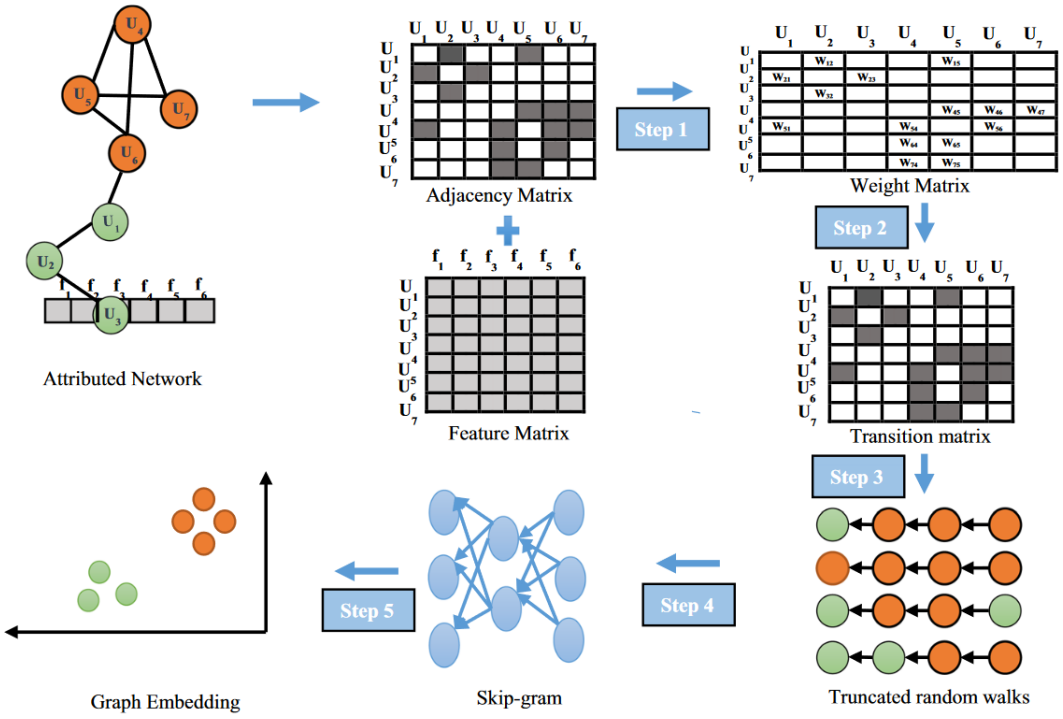


图 21: 算法流程图

计算相似度 正如上面提到的，算法的基本假设是相似的节点之间更有可能产生链接。而节点之间相似性分为结构上的相似性和属性上的相似性。这就需要计算两个矩阵：结构相似性矩阵 (MSS) 和属性相似性矩阵 (MAS)。

结构相似性矩阵（MSS）使用节点邻居集合的相似度来衡量节点的结构相似度：

$$MSS_{i,j} = \mathrm{sim}\left(\mathrm{v_i}, \mathrm{v_j} \mid \alpha\right) = A_{ij} \cdot \frac{\left|\mathrm{Cover}\left(v_i, \alpha\right) \cap \mathrm{Cover}\left(v_j, \alpha\right)\right|}{\left|\mathrm{Cover}\left(v_i, \alpha\right) \cup \mathrm{Cover}\left(v_j, \alpha\right)\right|} \tag{7}$$

其中，A 是邻接矩阵，$Cover(v_i, \alpha) = \{v_j \in V | dist(v_i, v_j) \leq \alpha\}$，$\alpha$ 表示最远的邻居的距离，$dist(v_i, v_j)$

表示 v_i 和 v_j 之间的最短路径距离。在实验中 α 被设置为 2，因为距离大于 2 的邻居对相似性的影响可以被忽略。

属性相似矩阵（MAS）基于节点本身属性的相似而与节点的结构无关：

$$MAS_{i,j} = \text{Jaccard coefficient } (v_i, v_j) = \frac{|I(v_i) \cap I(v_j)|}{|I(v_i) \cup I(v_j)|} \tag{8}$$

其中 $I(v_i)$ 是节点 v_i 的属性集合。

综合 MSS 和 MAS，我们可以得到一个权重矩阵 W：

$$w_{i,j} = \begin{cases} \{\beta * \text{MAS}(v_i, v_j) + (1-\beta) * \text{MSS}(v_i, v_j)\}, & \text{if } A(v_i, v_j) = 1 \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

$\beta \in [0, 1]$ 也是一个超参数，表示我们对结构信息和属性信息的偏好。

有偏游走采样 相较于传统的 DeepWalk 算法中纯随机的游走策略，这里采用的是考虑节点属性和结构信息的有偏的游走策略。具体来说，当游走到节点 v_{i-1} 时，下一个节点 v_i 将在 v_{i-1} 的邻居中依概率 $P(v_i|v_{i-1})$ 选择：

$$P(v_i \mid v_{i-1}) = \begin{cases} \frac{w_{ij}}{\sum_{j \in \Gamma(i)} w_{ij}}, & \text{if } (v_i; v_{i-1}) \in E \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

在获取多组游走序列后我们将其输入到 skip-gram 模型。Skip-gram 模型是一种典型的 word2vec 模型。Word2Vec 是通过学习文本来用词向量的方式表征词的语义信息，即通过一个嵌入空间使得语义上相似的单词在该空间内距离很近。Embedding 其实就是一个映射，将单词从原先所属的空间映射到新的多维空间中，也就是把原先词所在空间嵌入到一个新的空间中去。我们从直观角度上来理解一下，cat 这个单词和 kitten 属于语义上很相近的词，而 dog 和 kitten 则不是那么相近，iphone 这个单词和 kitten 的语义就差的更远了。通过对词汇表中单词进行这种数值表示方式的学习（也就是将单词转换为词向量），能够让我们基于这样的数值进行向量化的操作从而得到一些有趣的结论。比如说，如果我们对词向量 kitten、cat 以及 dog 执行这样的操作：kitten - cat + dog，那么最终得到的嵌入向量将与 puppy 这个词向量十分相近。而这种在 NLP 模型在一定假设情况下可以直接用在图的节点序列上以获取节点的嵌入向量。

对于一组节点序列 $\left(S_{vi} = s_{vi}^1, s_{vi}^2, s_{vi}^3 \ldots s_{vi}^k\right)$，skip-gram 模型通过对中心词进行窗口采样。假设窗口大小为 5，中心词为 S_{vi}^3，则采样窗口为 $(S_{vi}^1, S_{vi}^2, S_{vi}^3, S_{vi}^4, S_{vi}^5,)$，进而得到正例:$(S_{vi}^1, S_{vi}^3)$,$(S_{vi}^2, S_{vi}^3)$, (S_{vi}^4, S_{vi}^3), (S_{vi}^5, S_{vi}^3)。同时，skip-gram 模型将随机采样不在 S_{vi}^3 周围的节点作为负例。根据假设，skip-gram 将最大化正例共同出现的概率而最小化与负例共同出现的概率。损失函数如下：

$$minimize_w - \sum_{m=1}^{C} [\sum_{j=1}^{N} log\, \sigma(c_{pos_i} \cdot w) + \sum_{i=1}^{K} log\, \sigma(-c_{neg_i} \cdot w)] \tag{11}$$

其中 c_{pos} 和 c_{neg} 是正负例样本的 one-hot 编码，C 是中心词个数，N 是每个中心词的正例样本数，K 是每个中心词的负例样本数，w 是嵌入矩阵，$c \cdot w$ 是嵌入向量。经过迭代优化即可得到节点的嵌入表示。

6.2.3 链接预测

在得到节点嵌入之后，我们直接使用链接两端节点嵌入表示的哈德吗积作为链接的嵌入表示。预测模型采用带有 $L2$ 正则化的逻辑回归。对于回归模型 $f : \mathbb{R}^N \times \mathbb{R}^N \to [0, 1]$:

$$\begin{gathered} f\left(x_i, x_j\right) = \sigma\left(\left(x_i * x_j\right) \cdot w + b\right) \\ x_i, x_j \in \mathbb{R}^{\mathbb{N}} \end{gathered} \tag{12}$$

其中 $\sigma(\cdot)$ 是 softmax 函数。

6.2.4 实验

首先通过 modified deepwalk 算法得到节点的嵌入表示。然后对训练数据中每个病人节点采用 10 个已经存在的”药物-治疗-病人“链接作为正例并随机采样 10 个与之没有链接的药物节点作为负例。采样得到的所有数据根据 8：2 的比例划分为训练集和验证集。验证集用来对 Word2vec 中的嵌入维度以及窗口大小两个超参数进行交叉验证。最终选择了嵌入维度为 1024，窗口大小为 5。在完成训练之后遍历所有测试数据中病人和药物，预测两两之间存在链接的概率。之后，我们选取分类阈值 γ，如果概率大于 γ 则认为两节点之间存在链接，反之则不存在。特别地，在获取药物类别时，为了消除歧义，我们先将测试数据中给出的药物通过编码器获取药物的编码，示例见图 22。

```
(start) D:\Life\Study\Grade_3\知识工程实践\DeepWalk_linkprediction>python getNum.py
2型糖尿病,高血压病2级（高危组）,窦性心动过缓,前列腺癌术后,十二指肠乳头腺瘤术后,骨量减少,萎缩性胃炎,肝囊肿,左肾囊肿,胆囊内壁胆固醇结晶
['D-5', 'D-2677', 'D-1783', 'D-1552']
2型糖尿病,高血压病3级（极高危组）,阑尾切除术后,糖尿病周围神经病变,糖尿病肾病,脑萎缩,骨量减少,胆囊息肉,筛窦黏膜增厚,双下肢动脉粥样硬化症,左肾囊肿
['D-5', 'D-2682', 'D-1552']
```

图 22: 药物编码示例

6.2.5 结果分析

最终，调节逻辑回归不同的分类阈值分别计算 *precision*，*recall* 和 $F1$ 指标如图23：

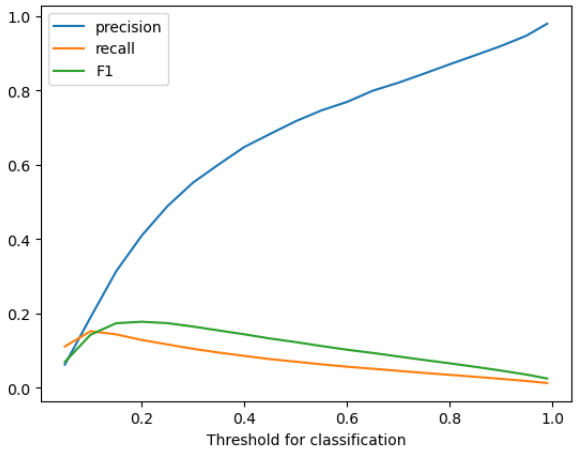


图 23: 分类各指标随分类阈值的变化

在分类阈值为 0.12 时 $F1$ 指标达到最大，为 0.182。此后随着阈值的增大，*precision* 持续上升，*recall*

先升后降。

可以看出饰演的结果并不理想，显著特点是精度高召回率低，说明模型在开处方时开了很多不必要的处方。我们分析可能的原因有以下几方面：

- **数据**
  - 由于专业知识的匮乏，在编码症状时很难将同类节点消歧，只能通过字符之间的相似性来进行消歧。这就导致疾病节点过多，每个疾病节点的邻居较少，导致得到的嵌入表示不够好
  - 没有考虑病人的体检数据
- **模型**
  - 采用链接两端节点的哈德吗积来表示链接，而图中存在不同类别的链接，导致图中的信息不能被完全利用
  - 没有仔细调节模型中的超参数, 比如获取 W 矩阵时的加权系数, 直接取的 0.5

6.2.6 改进方向

为了解决以上问题, 我们可以采用异质图神经网络, 比如 2019 年提出的 Heterogeneous Graph Attention Network 来考虑链接的类别信息, 获取链接的嵌入完成链接预测任务。

7 分工

姓名顺序和报告前后部分顺序保持一致。

- **俞家琛** 底层本体构建和关系抽取，表格数据清洗，初步带药抽取（仅针对已给出药物）
- **牟卓翊** PDF 文本处理，非结构化数据知识抽取，命名实体识别，数据增广
- **荣建凯** PDF 文本处理，非结构化数据知识抽取，命名实体识别，数据增广
- **张屹灵** 表格信息抽取、消歧，知识对齐
- **游皓月** 表格信息抽取、消歧，知识对齐
- **欧荣煌** 知识图谱构建、处方推荐模型
- **蔡英豪** 数据处理，整体流程设计管理，处方推荐模型