基于网页文本自然语言处理的分类预测 (项目 B)

颜铭　2013365

摘要

基于 WebKB 项目的网页文本数据集，通过爬虫解析 html 文本，通过自然语言处理的相关知识提取文本特征并利用关键词和解析文本作语义量化，再利用监督学习的典型方法进行网页标签分类和预测，并利用部分 html 网址的指向性构建无向图信息，并利用图学习理论的经典模型图卷积神经网络 gcn 通过预训图理论模型相关无监督指标，得到准确率较高的预训练测试文件，用来提高传统机器学习方法预测网页的准确性并对以上方法作性能比较

关键词： 网页文本爬虫, 自然语言处理, 文本特征量化, 随机森林算法, 梯度提升树, 支持向量机, 图卷积神经网络

一　导言

自然语言处理是机器学习方法应用的重点领域，其目标是通过机器识别人类语言特征理解人类的语义意图，从而在大量的文本信息沟通领域解放生产力。自上世纪末期的网络迅速发展，如何对以互联网为载体的海量信息流作出智能反馈个性化分析和多模态预测成为日常工作中需要解决的问题。作为重要数据主体的文本信息，随着量化和分类集成学习与统计方法在机器学习上的不断发展，逐渐成为技术中核心研究的数据，并以此为依据逐渐分化出了 NLP(nature language process 自然语言处理) 这一应用领域。随着文本特征提取技术的成熟，出现了很多经典的统计学习文本预处理方法如词袋模型，TF-IDF 特征，词嵌入模型等。深度学习时代，随着卷积神经网络的出现了以 Transformer 结构为代表的 Self-Attention 机制模型，使得自然语言处理在任务中的精准性较大提升，在一些领域获得了接近甚至超越人类的基线评分的效果　现如今自然语言处理在机器学习和深度学习结合的模式下不断运用到具体的数据挖掘数据分析项目中，如本文实验的网页文本分类就是早期互联网搜索引擎算法的重要任务。自然语言处理具体的任务场景可以为如下的几个方面:

1. **文本分类回归任务:**
   本实验的主题，传统机器学习方法最为通用的任务，核心流程是提取向量化自然语言特征并用学习模型加以训练预测，代表任务有语义分析，情感分析，文本模式识别等

2. **信息检索、文本匹配任务:**
   对网页文本为代表的信息交互如关键词检索和问答匹配等应用构造特征并利用合适模型提训练复杂语义分析和逻辑判断的匹配能力

3. **序列标注任务:**
   关注信息标签的序列生成，常见运用有语音文本转化机器翻译等，这主要用深度学习卷积神经网络模型，也会结合传统的条件随机场、隐马尔可夫模型等算法

4. **机器阅读任务:**
   对应给定问题文本通过训练得出符合文本要求的答案和方法，属于自然语言处理的前沿领域，基本的思路是利用上下文语义分析识别特定场景答案

本实验基于传统机器学习分类器，如监督学习分类器朴素贝叶斯法、K 近邻算法、支持向量机。集成学习中的 boosting 梯度提升树和 bagging 随机森林算法等综合方法，结合文本的统计方法量化特征，学习文本特征预测网页推送标签，并就如何提高整体或者个类的准确性实现基于图学习理论的训练预测优化

二　问题定义

(一)　编码问题

由于 WebKB 数据来自多个语言的学生网页和相关国际资源，且整体时间较早，在编写静态网页时遗留下一些现在严谨规则文本解析器无法读取的字符编码形式，典型的就是中文字符编码 gbk 的过早版本无法适用于现在的文本文件 utf-8 编码模式。为解决数据集这一问题，需要以二进制文件读取并统一以 utf-8 编码重新写入。对于一些原始编码取消造成无法识别编码，需要通过打印报告文件名，在 WebKB 文件夹中亲自以文本文件打开并手动编制为 UTF-8 编码

(二)　html 文本简介

HTML 指的是超文本标记语言，是通用制作**静态网页**的超级文本文档的简单标记语言，HTML 命令包括描述文字，图形，动画，声音，表格，链接等。例如 <Header> 描述浏览器需要的信息，代码作用主体包含要描述的具体内容

具体地，浏览器渲染网页时，会把 HTML 源码解析成一个标签树，每个标签都是树的一个节点。这种节点就称为网页元素。所有元素可以分为块级元素和行内元素。属性，是标签的额外信息，使用空格与标签名和其他属性分隔。属性可以用等号指定属性值，属性值一般放在双引号里面。常见的全局属性有标识符 id 属性，网页元素分类属性 class, 元素附加说明属性 title 等。同时 HTML 的关键索引部分为标签，其名称都带有基本语义，是**网页文本直观语义信息来源**，在网页爬虫抓取文本信息时可以很方便地解析语义结构的网页骨架，一些常用的命令容器有放置页面或文档的导航信息的 <nav>, 表示页面的主体内容的 <main>, 表示页面里面一段完整的内容的 <article>. 除此之外，类似命令操作符的通用文本标签是提取纯文本必须关注的重点对象，如 语块标签， 目的指向标签，
 换行标签，<dt> 解释补充标签等，利用爬虫库 **BeautifulSoup4** 的 html 解析器过滤这些命令和其引导的多余格式，并过滤没有命令控制的乱码字词，就得到了初步的有效纯文本

(三)　文本数据预处理

现在得到的有效纯文本是基本的语料，但是初步的文本信息缺乏逻辑相关性和显示量化特征，大量的相同文字位置和顺序的数据信息显著降低了机器学习算法运行效率，同时对其字频字符比描述语义等信息都缺乏了解，因此我们需要通过自然语言处理中的统计方法去量化提取文本的表征作为文本数据的预处理部分，本实验采用了无关词语料筛选，词袋模型、TF-IDF 向量化，词元解释语义法，语句正则化，去均值和方差归一化，n-grams 特征添加来预处理精简转化庞大的文本信息较显著量化特征的信息增益。

我们确定网页预测任务的基础是监督学习下的分类，对来自四所大学 cornell 康奈尔大学，wisconsin 威斯康辛大学，texas 得克萨斯大学，washington 华盛顿大学加增加数据的一所 misc 密歇根大学总共七个类:

资源类 course, 部门类 department, 学院类 faculty，其他类 other, 项目类 project, 职员类 staff, 学生类 student

作出具体类别的分类训练和测试预测，因此可选的划分训练和测试数据集的方案有选出一所大学作为测试集而其他大学的网站集合用做训练，或者将五所大学的不等量各类网站集合在一起随机按标签划分满足一定比例的训练-测试数据集，实际中对两种方法都进行了实现　　在确定文本内容量化数据和类标签后，即可通过机器学习手段进行分类拟合和测试集的预测并报告准确率，准确率由以下指标度量:

评估方法的基本术语: 真正例:TP，假正例 FP，真反例 TN，假反例 FN，且有 TP +

$FP + TN + FN$ = 样本总数, 其定义见表格:

真实情况	预测结果	
	正例	反例
正例	TP(真正例)	FN(假反例)
反例	FP(假正例)	TN(真反例)

$$\begin{cases} precsion\text{精度} \quad P = \frac{TP}{TP+FP} \\ recall\text{召回率} \quad R = \frac{TP}{TP+FN} \\ f1 - score\text{调和平均值} \quad \frac{2\times P\times R}{P+R} \\ macro - avg\text{宏均值} \quad \frac{P+R}{2} \\ macro - P, macro - R, macro - f1 \quad \text{类似的宏均值} \end{cases}$$

这就完成了基于文本内容的量化训练和预测评估, 但同时网页文本具有普通文本不同的属性体现在交互性上。对静态网页, 其最主要的交互性体现在各个网页间的指向性上, 因此可以引入图学习理论机器学习模型**图神经网络**结合深度学习卷积网络方法, 预训练这些关联的网页提升最终预测的准确性。此外, 对参考论文中的损失函数无监督特征提取的数值分析方法得到量化特征也进行了实验比较连接特征和基于内容的训练的机器学习模型的优劣

三　设计算法原理

(一)　文本特征量化

1. 词袋模型

词袋模型是一种最简单、最直接的特征提取方式, 选择忽略词的位置信息如上下文关系, 假设词性间独立, 通过丢失一定预测精度的代价用词频表征语句信息。也即统计字词出现的字数频率并输出为字典对应的稀疏向量, 经过标签变码后即可将这些离散特征向量化后利用机器学习手段进行训练

2. TF-IDF 向量化

为完善词袋模型上下文语境缺失的不足, 增强考量词本身在句子中的重要性, 产生了 TF-IDF 方法, 通过对频率进行加权优化词为单位的表征能力:

- TF 计算法为: 词在句子中出现次数 \文档中的词总数
- IDF 计算法为: log(文档总数 \包含词的文档总数)

基于上述式子的 TF 和 IDF, 再将两者相乘即可得到 TF-IDF, 词袋模型结合 TF-IDF 方法使用稀疏矩阵来表达语句含义, 具有简单易用速度快的优点, 但是对于文本语料较少字典大小过大的情况缺乏基本构建语料时, 表征能力极大受到统计数据的制约, 很容易导致模型在训练过程中过拟合。这在本实验中自行训练的随机森林和非线性核支持向量机中有体现

3. N-gram 模型

在自然语言处理中, 一个重要课题是如何基于统计方法研究句子表征。结合无记忆性的状态空间中经过从一个状态到另一个状态的转换的随机过程马尔科夫链以及每个词出现概率只和前面几个词相关的马尔科夫假设, 构建了句式概率模型: 已知句子 $S(w_1, w_2, w_3, \cdots, w_n)$, 这里 w_i 代表句子中的词, 计算句子出现的概率 $p(S)$, 则表达式为

$$p(S) = p(w_1) \times p(w_2) \times p(w_3) \times \cdots \times p(w_n)$$

如果引入马尔科夫假设进行修正, 进一步可以得到 N-gram 模型公式

$$p(S) = p(w_1) \times p(w_2|w_1) \times \cdots \times p(w_n|w_{n-1})$$

结合词袋模型的基本理念可以进一步提升文本特征的预测能力

具体地, 构建二元模型 Bi-gram :$p(w_1, w_2, \cdots, w_n) = \prod_{i=1}^{N} p(w_i|w_{i-1})$ 以及三元模型 Tri-gram: $p(w_1, w_2, \cdots, w_n) = \prod_{i=1}^{N} p(w_i|w_{i-1}|w_{i-2})$ 生成额外的稀疏特征向量, 这样再运用到词袋模型加 TF-IDF 特征上加强表征能力同时又能获取一定上下文信息, 但同时也提高了我们处理长序列相关性的要求

4. 其他相关量化模型

对文本中的近义和歧义词, 为利用文本相似度来计算未标注的文本信息的语境词, 进一步将词性量化推广

为词向量，并提出词嵌入模型优化问题。词向量基于先验假设并结合上下文推断对特殊的罕见词和多义词有很好的泛化能力，再以查询表的形式记录每一个词训练的生成向量即可灵活提取不同特征。此外深度学习结合文本传导编码机制不断演化出了上下文相关的预训练模型如 BERT、GRT、ELMO 模型等。由于 WebKB 数据集的样本数量并不多而且网页构成与类别相关性比较简单，因此并不需要这些结合深度学习机制的方法就可以得到不错的文本预测效果

5. 如何精简文本

去除纯文本中的冗余部分能有效简化运算和集成特征标识。一些简单的谓词和介词无疑会对与其相关的表述文本事件的名词和动词的特征提取产生干扰，因此可以使用 nltk 的 Stopwords 词性库去除这些无关的破碎词组，对有效的字符信息进行基于语言归并工具的词根-语义的语言维度简化，整合上下文本的一般性。此外文本信息的标点符号和一连串数字分布在词袋模型下没有显然规律，对机器学习训练无疑是负样本也应积极地在句子外支删去。对删除可能造成的大面积空格情况，也应归并成一个空格，仅作为分隔标识使用　　同时作为网址的文件名的编写方式也与现在的超文本协议格式有出入，为使得预测结果更实用，需要转译网址格式。对文本的字典和简单工作令牌标识处理可以快速索引同时监督修正文本流水线的词云生成，根据适应度表达局部特征和语块连贯性

(二)　机器学习算法选择

1. 单端基本学习分类器

利用朴素贝叶斯方法和 K 近邻方法可以测试其对多维特征过程数据的单输入输出的预测分类效果。由于朴素贝叶斯法是基于特征条件独立性假设求后验概率最大化，在上下文语义相关的场景下会受到很大的噪声干扰与信逻辑和复杂语义信息差异常导致的概率修正偏置异常，明显贝叶斯神经网络会跟适合语义逻辑的相关传递性质。同样地，K 近邻算法受到 k 值选择的制约，而对于基于量化关键语义的语言关联定位的多维数据点的距离度量叠加信息就要求不能过小的 k 值选择避免局部的分化语义词组的噪声干扰，而只要扩展领域搜索上下文不断的反馈又会引起模型适用 k 值对快速增长，这和要求文本局部信息到整体信息的近似误差越小越好的要求直接矛盾。因此无论是贝叶斯还是 K 近邻算法这样的单端度量决策模型都并不能很好地提升预测准确性。实际上也只能二分类复杂随机拟合，准确率也就 0.5 左右

2. 数值分析方法的模式识别分类器

典型的就是 SVM 支持向量机，支持向量的概念可以作为文本的词向量的补充，也较好地分析了文本特征矩阵的词组频率聚合和语义归并表达的关联词特征空间间隔等问题，其参数选择的调制可以通过数值分析方法计算损失并结合语义传递的过程迭代中自适应集中干扰点并利用支持向量标签化潜在的语义，线性技巧符合文本相关性信息差的约束和上下文局部特征分解的优化目标定位,其软间隔综合文本凸优化对偶形式为:

$$\min_{\alpha} \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i \cdot x_j) - \sum_{i=1}^{N}\alpha_i$$
$$\text{s.t.} \sum_{i=1}^{N}\alpha_i y_i = 0$$
$$0 \leqslant \alpha_i \leqslant C, i = 1, 2, \cdots N$$

而非线性核技巧可以更适合内容本身对向量的字符串词根范围内积高维空间的特征解析，具体字符串 (语句串) 核函数构建如下:
一个有限的字符表 σ，字符串 s 为从 σ 中取出的有限字符序列，长度为 n 的字符串集合为 σ^n，字符串 s 和字符串 t 的连接为 st，记所有字符串集合为 $\sigma^* = \bigcup_{n=0}^{\infty}\sigma^n$，对于字符串 s 的子串 u，给定指标序列 $i = (i_1, i_2, \cdots, i_{|u|})$ 且长度非减，子串定义为 $u = s(i) = \prod_{j=1}^{u} s(i_j)$，长度为 $l(i) = i_{|u|} - i_1 + 1$，且当 $l(i) = |u|$ 时 i 连续否则左边恒大于右边

现有 $\mathcal{S}$ 为长度大于或等于 n 的字符串集合，$s \in \mathcal{S}$，建立字符串集合到特征空间 $\mathcal{H}_n = R^{\sigma^n}$ 的映射 $\phi_n(s)$，

映射在字符串 s 对应特征 $= R^{\sigma^n}$ 空间的一个向量，定义在子串 u 维上的取值为 $[\phi_n(s)]_u = \sum_{i:s(i)=u} \lambda^{l(i)}$, 只需让 $0 \leq \lambda \leqslant 1$ 为一个衰减常数即可，求和在 s 的所有与 u 模式串匹配的子串上进行，字符串核函数 $k_n(s,t)$ 为基于映射的特征空间内积:

$$\begin{aligned} k_n(s,t) &= \sum_{u \in \sigma^n} [\phi_n(s)]_u [\phi_n(t)]_u \\ &= \sum_{u \in \sigma^n} \sum_{(i,j):s(i)=t(j)=u} \lambda^{l(i)} \lambda^{l(j)} \end{aligned}$$

字符串核函数 $k_n(s,t)$ 给出了两个字符串的长度等于 n 的所有子串组成的特征向量余弦相似度并由动态规划算法快速计算，在文本分类的信息检索中有很好泛化能力，在拉格朗日函数上利用核函数约化 $\frac{1}{2} y^T \alpha \alpha^T y$ 即可

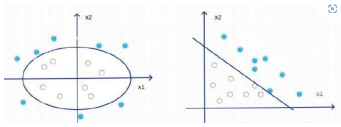

图 1: 非线性核函数技巧特征

因此无论是对纯文本内容还是对精简文本的测试 SVM 都能表现出优良的鲁棒性。而当文本模型引入基于马尔科夫假设的 N-grams 特征时，支持向量机求解效率和准确性都大为降低，本质原因是概率分割的记忆启发假设规划了多个分类子任务，演化出训练多个分类器表征，我们需要集成学习 boosting 或者 bagging 方法去训练多个分类器

3. 随机森林算法的 Bagging

Bagging 是一种并行式的集成学习方法，通过对数据集自身采样获取不同的子集并针对训练基分类器作为模型集成，其核心概念为**自助采样**, 即给定包含 m 个样本的数据集有放回地随机抽取一个样本放入采样集中，经过 m 次采样得到一个和原始数据集大小一致的采样集，并最终采样得到 T 个包含 m 个样本的采样集并基于分类器实现并行训练。典型的算法是随机森林，其每个分类器是决策树的组合这样并行结合反馈综合特征适用于高频转折文本信息流。对决策树原理不展开，简略说明随机森林算法流程如下

- 假设有 M 个样本，有放回地随机选择 M 个样本
- 假设样本有 N 个特征，在决策时每个结点需要分裂时随机从这 N 个特征中选取 n 个特征，满足 $n << N$, 从这 n 个特征进行结点分裂
- 基于抽样的 M 个样本 n 个特征按照结点分裂方构建决策树
- 按照前面三步的方法构建大量决策树组成随机森林然后对结果进行综合 (分类使用多数表决法，回归使用均值方差法)

并行模式下文本的类别潜在语义交互信息能在训练分类层更好地回归，这也使得随机森林算法获得了高性能和高精度，成为综合处理文本信息增益多样句式词根分化的热门机器学习算法。值得注意的是，并行分裂文本特征结点注重整体强化分类组合的协调，由于剪枝决策的影响和无记忆性区分子树叶节点的特性，随机森林算法对 N-grams 特征的适应度是最好的，这也另一个层面体现了综合文本信息流水线的优化，决策模拟的句式嵌入动态平衡链式树枝的表意回归和组合结构也是一大要素

4. 梯度提升树的 Boosting

Boosting 是集成学习将多个单模型弱分类器组合为强分类器的框架算法，提升算法因此得名。在训练多个弱分类器的过程中不断改变训练样本的概率分布，使得算法跟进上一个弱分类器的错误，并在不同阶段解析分类器综合性能的评估与试探，得到一个串行的组合强分类属性容器。其中最为成熟的算法代表是极度梯度提升树 XGboost, 利用展开到二阶的损失函数逼近真实损失提升泛化能力。提升树知识不展开叙述，简化推导构建流程为:

- 对损失函数 $L = \sum_{i=1}^{n} l(y_i \hat{y}^i) + \sum_{i=1}^{t} \Omega(f(i))$ 二阶泰勒展开得到 $l(y_i, \hat{y}(t-1) + g_i f_i(x) + \frac{1}{2} h_i f_i^2(x_i)$, 同时对前 $t-1$ 棵树进行量化
- 定义一棵树 $f_c(x) = w_q(x)$，同时定义数的复杂度 $\gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$
- 对树的定义做叶子结点分组 $\sum_{j=1}^{T} \left\{ (\sum_{i \in l_j} g_i) w_j + \frac{1}{2} (\sum_{i \in l_j} h_i + \lambda) w_j^2 \right\} + \gamma T$，得到最终的损失函数 $\sum_{j=1}^{T} \left\{ G_j w_j + \frac{1}{2}(H_j + \lambda) w_j^2 \right\} + \gamma T$
- 进行最优点以及最优取值:$L = -\frac{1}{2} \sum_{j=1}^{T} \frac{G_j^2}{H_j + \lambda} + \gamma T$，并以叶子结点分裂标准评估 $Gain = \frac{1}{2} \left\{ \frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} \right\} - \gamma$

极度梯度提升树 XGboost 对真实损失度量的评估性能是集成学习中最为优势的地方，因此对文本的简化很敏感，这样成为性能评估上的不足，但是总体性能还是相对稳定。但也正因 Boost 如此的泛化能力提升使其表现出其他机器学习方法没有的重叠词云文本的特征整合检索能力，由于 WebKB 样本结合了大学官网和教师和学生私人界面，对于浮动的分布式集中更有利于关注，正则归并文本表达内容框架内的个性化注释并调动离散文本生成词性敏感来源的权重反馈机制加强词到句的特征完备性。不同于精简文本，纯文本的样本数量增加因此也有意义可以有效地追踪权重文本解释的分配并作为预测的转移依据。

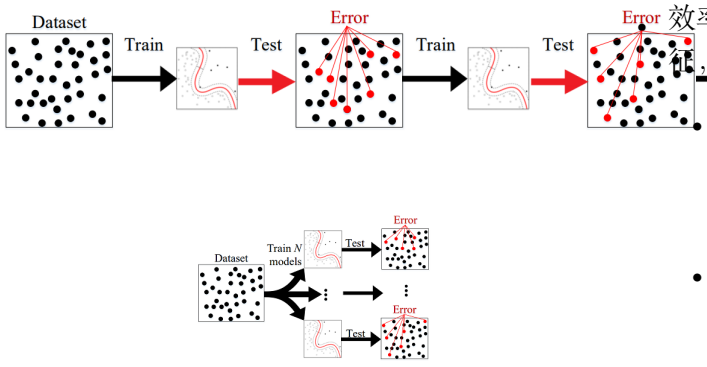

图 2: Boosting 和 Bagging 性能理解

5. 图学习模型

图作为一种非欧式结构的结构化数据，它由一系列的对象 (结点) 和关系类型 (边) 组成，具有局部连接的特点，能表示更为复杂的信息经典的图学习模型通过关系对组合为无向边，并结合网址结点的并发的入度和出度关系，以前向链表点-边关系构建邻接矩阵，通过正交归一化得到对称正定阵并进行最小二乘回归或矩阵分解等方法得到样本的无监督表征信息。结合神经网络的知识就得到了 Gnn 神经网络，相比较于神经网络最基本的网络结构全连接层（MLP），特征矩阵乘以权重矩阵，图神经网络多了一个邻接矩阵。计算形式很简单，三个矩阵相乘再加上一个非线性变换

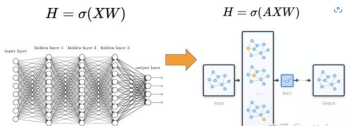

图 3: 图神经网络理论发展模型

在此基础上引入卷积核就得到了图卷积神经网络 GCN, 其本质目的就是用来提取拓扑图的空间特征。基于频谱的方法建立在全局的归一化的图拉普拉斯矩阵 (实对称矩阵，是无向图的数学表示) 之上，故而假定了图为无向图，难以处理大规模图，有向图以及动态图；而基于空间的方法较为灵活，可以引入采样技术来提高效率。实验代码基于频谱的卷积提取邻居结点的聚合特征，简化步骤分为:

- 由复杂的谱图分解理论推导得到简单的网络函数结果表达式 $h_v = f(\frac{1}{|u| \in N(v)} W x_u + b), \forall v \in \mathcal{V}$, 也即聚合特征的邻居表达作线性变化
- 同时为了使 GCN 可以捕捉到最优 k 个邻居的信息，堆叠多层 GCN 得到迭代表达式 $h_v^{k+1} = f(\frac{1}{|u| \in N(v)} W h_u^k + b^k)$, 用矩阵表达式写作

$$H^{(l+1)} = \sigma(D^{-\frac{1}{2}} \bar{A} D^{-\frac{1}{2}T} H^{(l)} W^{(l)})$$

这里 $D^{-\frac{1}{2}}\bar{A}D^{-\frac{1}{2}T}$ 也即归一化后的邻接矩阵，$H^{(l)}W^{(l)}$ 相当于对 l 层的所有结点嵌入模型做了一次线性变化，左乘以邻接矩阵表示对每个节点来说，该节点的特征表示为邻居节点特征相加之后的结果。初始的处理简单的把原始理论的对角线元素 $diag(\hat{h}(\lambda_1),\cdots,\hat{h}(\lambda_n))$ 替换为 θ。出现了图卷积核参数量大，参数量与图中的节点的数量相同、卷积核是全局的、运算过程设计到特征分解复杂度高等缺点，经过优化可以通过 k 阶多项式逼近，归一化对称阵就是关键的一环。

实验基于结点文件 WebKB.content 和关系对文件 WebKB.cites 求解归一化邻接矩阵和其分解线性相关特征向量，搭建 GCN 网络预训练子集模型，并通过准确率高达 0.97 的模型参数作为测试集的修正优化依据，提升了机器学习算法对网页类型的预测准确度，尤其是**教职工学生界面这类个性化定义和项目类界面这类欠拟合批注，缺少自由不变性描述信息算子的界面**

6. 关于参考论文的无监督特征参数手段结合支持向量机

作者以网页连接关系特征为切入点，在得到邻接矩阵 A 和无向带权图 DAG 等图模型特征的基础上，对特征矩阵进行矩阵分解实现数据降维处理，以便克服稀疏特征矩阵学习中欠拟合准确率不高的问题并突出主成分特征提高分类效率，得到了解决主成分分析的无法回溯特征分解来源以及影响因子问题的分解最优化模型:

$$\min_{Z,U}\|A-ZUZ^T\|_F^2+\gamma\|U\|_F^2$$

此外引入内容矩阵 C 并利用语义空间特征向量 V 通过 ZV^T 正则化估计函数:

$$\min_{Z,V}\|C-ZV^T\|_F^2+\beta\|V\|_F^2$$

进一步实现综合网页量化文本和图连接特征的机器学习损失函数为

$$\min_{U,V,W}\left\{J(U,V,W)\overset{\text{def}}{=}\|A-ZUZ^T\|_F^2+\alpha\|C-ZV^T\|_F^2+\gamma\|U\|_F^2+\beta\|V\|_F^2\right\}$$

并利用梯度下降法迭代求解:

$$\frac{\partial J}{\partial U}=(Z^TZUZ^TZ-Z^TAZ)+\gamma U$$
$$\frac{\partial J}{\partial V}=\alpha(VZ^TZ-C^TZ)+\beta V$$
$$\frac{\partial J}{\partial Z}=(ZU^TZ^TZU+ZUZ^TZU^T-A^TZU-AZU^T)+\alpha(ZV^T$$

```
def gradZ(Z,A,V,C,ZU1,ZU2,alpha):
    Z_grad=(np.matmul(np.matmul(ZU1,Z.T),ZU2)+\
            np.matmul(np.matmul(ZU2,Z.T),ZU1)-\
            np.matmul(A.T,ZU2)-
            np.matmul(A,ZU1))+
            alpha*(np.matmul(np.matmul(Z,V.T),V)-\
                   np.matmul(C,V))
    return Z_grad
def link_content_MF(A,C,l,alpha=0.1,beta=0.01,gamma=0
    """
    """
^^IU_grad=gradU(U,Z,Z2,A,gamma)
^^IV_grad=gradV(V,Z,Z2,C,alpha,beta)
^^IZ_grad=gradZ(Z,A,V,C,ZU1,ZU2,alpha)
^^IU -= learning_rate*U_grad
^^IV -= learning_rate*V_grad
^^IZ -= learning_rate*Z_grad
```

四　实验设计描述

(一)　数据预处理

- 首先解决编码问题，根据二进制文件读写和转化机制遍历文件路径后重新编码写入即可，注意对报告文件名的文件需要到指定路径里手动打开并修改**(ConvertUtf-8.py)**
- 网页爬虫抓取文本，提供两种方案。一种是通过 html 解析器遍历搜索提取不保存的实时纯文本数据，另一种是通过无关词和数字标点删除并词组归

并的预处理精简文本，并标记上网址学校类别等信息存入数据框 (DataFrame) 中 **(FetchData.py)**

(二) 网页特征分析

1. 特征标签构成分析

完成网址的修正并提取词频

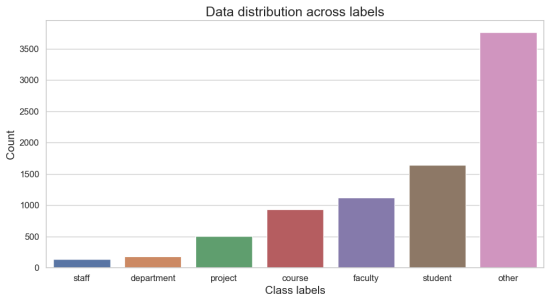

图 4: 类别统计直方图

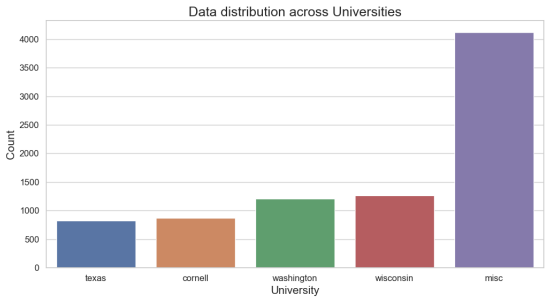

图 5: 学校来源统计直方图

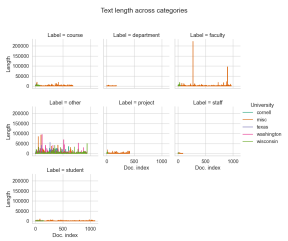

图 6: 各标签文本长度量化图

2. 词组维度特征分布

lambda 表达式通过递归启发遍历添加 2 维和 3 维相关语块维度特征

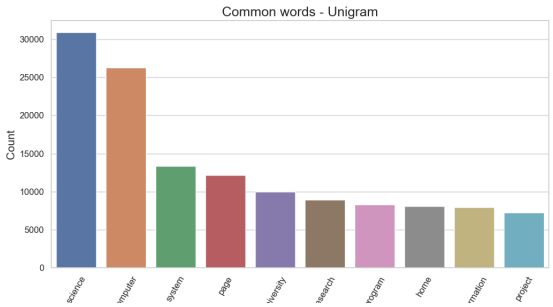

图 7: 单位词组兴趣分布直方图

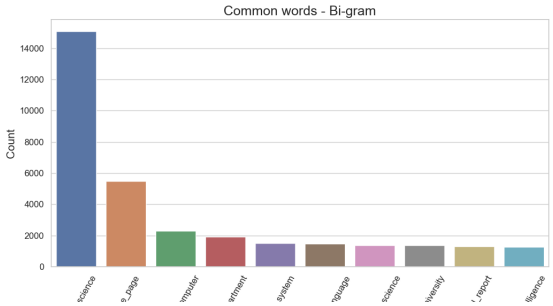

图 8: 二维特征词组兴趣分布直方图

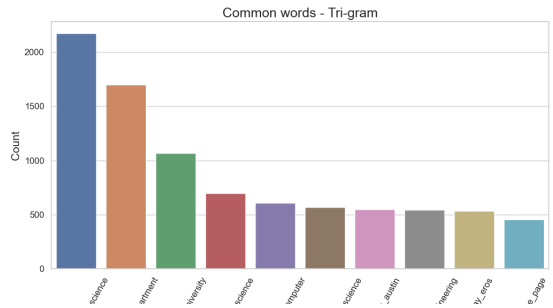

图 9: 三维特征词组兴趣分布直方图

3. 词云可视化

划分数据集，有两种方式，一种按学校，一种按类别随机抽样，得到特征索引的训练测试数据与标签集

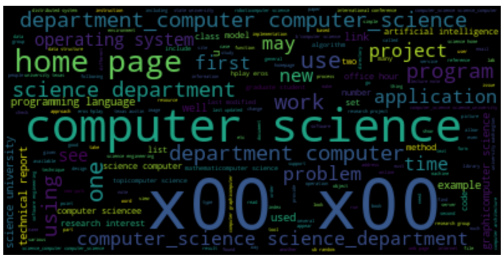

图 10: 词云叠加分析图

4. 随机森林算法的参数验证方法

使用网格搜索综合字典解析利用交叉验证确定最优的决策树个数与最佳评估 k 特征提取的 k 值

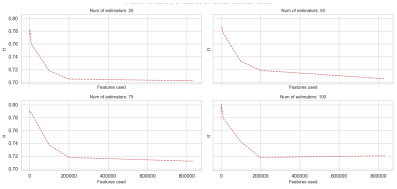

图 11: 不同树桩个数随着 k 值变动准确率曲线

5. 模型评估

- 由于没有划分验证集，使用随机打乱的原始数据进行岭回归后报告验证集准确指标。从验证曲线上可以观察在较宽的范围内模型由欠拟合到过拟合的可能过程并选择核心最优超参数。注意验证集修正后需要其他测试集体现泛化能力

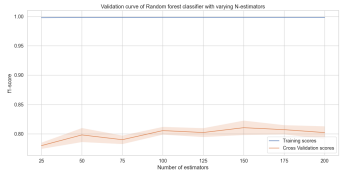

图 12: 验证曲线绘制

- 学习曲线: 内部通过交叉验证获得分数，通过画出不同训练集大小时训练集和交叉验证的准确率，可以看到模型在新数据上的表现，进而来判断模型是否方差偏高或偏差过高，以及增大训练集是否可以减小过拟合。

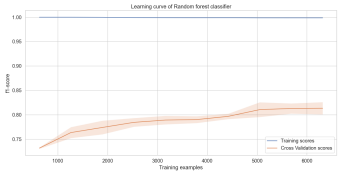

图 13: 学习曲线绘制

- 进一步对模型复杂性结合欠拟合过拟合问题使用统计量化的偏差-方差的权衡评估方法；总误差公式 $Totalerr = Bias^2 + Var + Noise$ 偏差平方加方差加噪音。偏差注重对整体模型均值的把控，方差注重个体差异的把控。若算法从数据集学习真实信号的灵活性有限时，就会出现偏差。方差关注训练数据灵活性高但是对未训练的数据泛化能力弱
高偏差模型说明问题: 模型缺乏灵活性, 模型太简单，欠拟合且偏差就是近似误差大;
高方差模型说明问题: 模型灵活性太高, 训练集上模型错误率低，但在测试集上错误率高, 无约束模型基本上记住了训练集，包括所有的噪声，导致过拟合

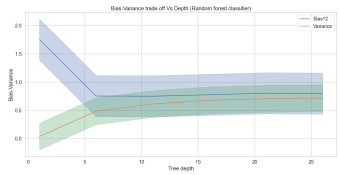

图 14: 偏差方差均衡曲线绘制

评估确定参数后即可利用随机森林训练预测数据

(三) 三个主流分类器模型的比较

- 分别在 (RandFcls.py) 中使用随机森林模型，在 (XgbCls.py) 中使用极度梯度提升树，在 (SVMcls.py) 中使用支持向量机模型，对纯文本和精简文本两种不同的数据预处理方式作准确率对比

- **结果:** 对于随机森林模型，纯文本模型准确率略高于精简文本 (见 RdfAns.txt); 对于极度梯度提升树 xgboost 模型纯文本对精简文本有了较大的预测优势，但是确不比精简文本类别预测均衡，尤其体现在私人个性化界面和无批注说明的项目类界面上 (见 XgbAns.txt); 对于支持向量机模型，整体实验上精简文本略高于纯文本但差别不大且有浮动 (见 SVMAns.txt)
- **其他:** 此外也对朴素贝叶斯法和 K 近邻方法进行了训练预测，准确率都在 0.4 0.6 左右 (见 BayesAns.txt 和 KNNAns.txt)

(四) Gnn 网络模型

首先读取 WebKB.cites 和 WebKB.content 数据并构建归一化邻接矩阵和特征矩阵 (utils.py), 并实现邻接矩阵和支持特征权重的网络聚合信息运算单神经层 (layers.py), 再搭建特征层到隐藏层再到类特征输出层的卷积神经网络模型 (models.py), 最后在主函数 (gcn_main.py) 中利用交叉熵损失函数和动量优化方法预训练图网络模型并报告准确率:

Epoch 5000 Train loss: 0.0969 Train accuracy: 0.9738 Validation loss: 0.0634 Validation accuracy: 0.9943 Time: 0.0090s

并将预测的参数模型的网址和标签写入网页分类项目路径的数据框中提供优化。**以指定威斯康辛大学测试集使用极度梯度提升树为例，标签修正占比越为测试集数量 $\frac{1}{25}$，原始测试集准确率为 0.83，结合 0.97 的预训练模型准确率，加权平均后准确率提升为 0.88 增长了 5 个百分点，确实有较为显著的模型准确率提高效果**

(五) 混合连接与内容的矩阵分解主特征量化法

首先使用和 Gnn 网络模型一样的数据读取和邻接矩阵特征矩阵生成方法，并添加监督标签 (theUtils.py), 根据论文的公式使用梯度下降法极小化损失函数迭代求解得到 $n \times l$ 维的量化矩阵 Z，并使用非线性核函数支持向量机对随机划分的训练测试数据集进行预测训练，结果准确性受参数影响较大，一般在 0.85 左右但是也会有过拟合特征降低到 0.45，确实就基于纯文本的内容分析增加了估计 0.1 的准确率，但是没有验证出论文提到的 0.95 以上更高准确率 (link_content.py)

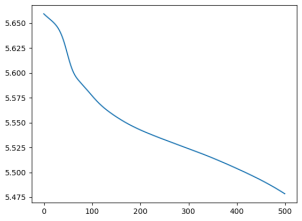

图 15: 损失函数错误参数下降曲线

(六) 利用预测结果实现的简单网页推送交互脚本

有了预测标签，就可以根据初始需要访问的页面类别标签或是大学标签来找出符合范围的前 5 个可访问网址，仍然使用精简文本，词袋模型 +TF-IDF 向量化 +N-grams 处理的数据，使用报告打印交互输入输出，，可以选择是否用 GCN 预训练的高精度模型增加准确率，具体代码见 (WebPage_Cls.py)，具体交互信息见 (IO_report.txt)

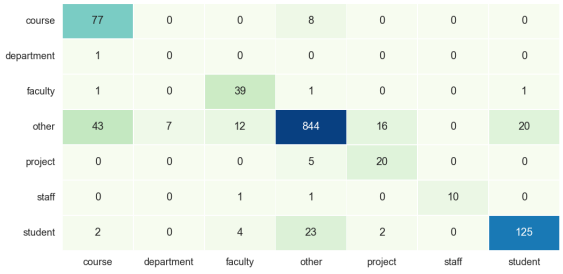

图 16: 预测分布矩阵

五 结论与展望

自然语言处理作为机器学习的热点应用领域，本次项目很好地提供了课外知识学习并回归课本基础理论

的机会。就项目本身而言，文本分类任务的机器学习方法已逐渐被深度学习方法取代，或是更多地结合两者，但是通过不同学习分类器模式的准确率性能比较差异分析可以看到，个性的官方的说明的等各类页面在潜在语义或情感语义属性的特征上对准确率和综合评估的影响，使用连接模型对个性化私人界面有很好的研究价值。第一个项目子任务根据现有的机器学习方法得以将准确率稳定在 0.8 左右，但是对于实际应用 0.8 也才是刚可以接受的标准，同时第二个子任务中 GCN 网络的方差检验也较高对测试集效果仍不如训练集理想，特征矩阵无监督主分解的数值分析方法，也卡在了 0.8 到 0.9 的大关。因此如何去突破准确性达到稳定的 0.95 标准，需要词嵌入加深度学习模型实现。由于项目主体还是机器学习方法的理论实践运用，故不再展开。

机器学习作为智能机器算法的基础中的基础，其数值和启发手段通过一个实际生活中的方面灵活多样地体现出来。机器学习的特质没有衰退而是形式上演化和内核的提升，需要进一步学习和创新项目为智能算法的发展作贡献

参考文献

[1] Zhu, Shenghuo, et al. ”Combining content and link for classification using matrix factorization.” Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval. 2007.