一、项目背景

1.1 整体背景

随着经济的高速发展，我国步入市场经济改革的关键时期，企业注册数量不断增加。国家统计局在2016年末的时候对全国企业进行了一次调查，调查结果表明截止目前，全国企业的数量达到了37万户，相比2016年初增长了近0.5万户[1]。今年来，，随着持续推进“放管服”改革、持续优化营商环境、加大双创力度等措施的实施，企业数量持续稳定增长，2018 年 1 月至 11 月期间，企业的注册数量达到了平均每日 1.81 万户，伴随企业注册量不断增长的同时，其产生的企业信息数据量也越来越多，每年呈指数级增长，由此造成了企业信息大量、冗余的现状[2]。

伴随社会的快速发展，同时在互联网逐渐步入大数据和云时代后，便可以更加快捷的获得企业的工商、司法、经营、上市、知识产权、舆情等多维度数据。2016 年 9 月，国家工商总局发布《工商总局关于新形势下推进监管方式改革创新的意见》（工商企监字〔2016〕185号），指出要“依托大数据加强监管”，“充分发挥大数据在制定完善新型市场监管制度和政策中的作用，搜集掌握经营者、消费者和社会公众的反应，跟踪监测有关制度和政策的实施效果”，“在工商登记、企业监管、网络交易、竞争执法、消费维权等领域率先开展大数据示范应用”。通过对企业从业人数、成立年限、注册资本、营业收入、风险信息，行政处罚、纳税信用等级、黑名单、上市信息、电商信息等数据关联处理，按需进行权重分割，并对这些数据进行综合分析，构建出企业全息画像，整体评估一个企业综合价值，更好的勾勒出企业的经营变化情况。

如何根据企业全息画像，面向不同的应用场景对企业分类识别，分析挖掘法人、自然人、物品（包括特种设备、产品、食品等）、案事件之间的显性和隐性关系，建立市场主体信用管理与风险评估模型，对企业的信用风险、经营风险、质量风险、安全风险、关联风险等进行全面评估及分级预警，为系统监管、精准监管、高效监管提供数据支撑，越来越受到市场监管部门的重视。

1.2 公司背景

湖南科创信息技术股份有限公司成立于 1998 年，是国内智慧政务及智慧企业领域的信息化综合服务提供商，主要致力于为政企客户提供集软件开发、系统集成、IT 运维等于一体的信息化综合解决方案。

科创信息注重跟踪最新技术发展趋势，在大数据、云计算、移动互联网、人工智能等领域都已创建了多个具有自主知识产权且处于国内领先地位的核心技术平台。在此基础上，按照“平台+应用”的研发模式，针对政府、企业客户的典型应用需求，逐步形成了“智慧政务”与“智慧企业”系列解决方案，客户覆盖湖南、湖北、云南、河南、北京等多省（市）党政机关及其下属公安、财政、税务、环保、医疗、教育等各级政府部门，并为中国铁路总公司、中国移动、中国联通、中国电信、中联重科、云南冶金、中建材集团等不同行业领域龙头企业提供了多项信息化综合解决方案。

依托强大的研发实力与突出的项目实施效果，公司解决方案及产品曾获得湖南省科技进步奖、国家重点新产品、中国优秀测绘工程奖、中国工业软件杰出贡献奖、中国十佳电子政务解决方案奖、中国智慧政务领域最佳解决方案奖、最佳智慧城市解决方案提供商奖等奖项，公司先后荣获国家高新技术企业、中国服务外包成长型100强企业、2017中国最具影响力软件和信息服务企业等诸多荣誉，是湖南省最具成长性的非公有制企业。

公司秉承“用户至上，品质优先”的服务理念，凭借多年的项目实施经验及良好的服务信誉，已在智慧政务及智慧企业领域形成了自己独特的竞争优势，并取得了良好的社会效益及经济效益。

1.3 业务背景

僵尸企业是指缺乏盈利能力却能够以低于市场最优利率成本获得信贷资源，依靠外界输血而缺乏自生能力的企业。僵尸企业的存在破坏了市场机制，加剧了信贷资源的错配，带来了严重的产能过剩问题，还对其他非僵尸企业产生了投资挤出效应。此外，由于我国金融制度不健全，僵尸企业往往会倾向于采用企业间商业信用的非正式金融方式融资，带来对僵尸企业合作关联企业的“传染”，甚至会拖累银行成为“僵尸银行”。僵尸企业如果不能够及时处置，经济动能转换以及高质量发展就无法保障，去杠杆工作的受阻将加大系统性金融风险。鉴于僵尸企业的危害性，自 2015 年以来，中央出台了僵尸企业处置的一系列政策指导文件，试图以处置僵尸企业作为深化供给侧结构性改革、调整经济结构以及促进经济行稳致远的重要抓手。

僵尸企业的识别标准主要有官方标准、CHK 标准，FN-CHK 标准以及各类FN-CHK 修正标准等，官方标准定义为不符合国家能耗、环保、质量、安全等标准，持续亏损三年以上且不符合结构调整方向；已停产、半停产、连年亏损、资不抵债要靠政府补贴和银行续贷维持经营的企业。CHK 标准的核心是企业是否接受信贷补贴，FN-CHK 标准则包含“真实利润原则”以及“常青贷款原则”，相关的 FN-CHK 修正方法大部分都是对以上两类标准的修正，对企业利润与资产负债率等指标进行调整，将企业的经营管理费用、净资产水平、企业效率和创新等指标引入僵尸企业的识别标准体系中，力求从更加多维的层次反映僵尸企业的经营特征。

通过各类标准分类识别僵尸企业均有其局限性，尤其是标准的制定有其主观性，因此基于企业画像分类识别方法成为首选。

1.4 项目内容

在数据已经成为战略资源的今天，通过数据挖掘和机器学习的方法来分析海量数据，期望挖掘出有用信息，进而帮助用户做出合理决策。企业画像，简单说就是企业给人的印象，可以跟自然人的用户画像相类比。具体做法是，依托企业真实的背景信息、风险信息、经营信息、对外投资信息、知识产权信息等，在这些真实数据的基础上为企业建立标签模型体系，将企业的具体行为属性进行抽象化、标签化，尝试从三大维度：基本信息维度、经营信息维度、风险信息维度进行标签提取，最终形成一个多元化的企业标签对象。

本项目致力于为政府机构和相关部门以及有需要的群体提供一个识别僵尸企业的大数据平台。通过用户提供的企业相关数据，包括基本信息表、知识产权表、融资表和年表，对该企业预测是否为僵尸企业，进而帮助用户进行相关决策。基于提供的企业数据，对企业进行全方位的行为分析，构建有效特征，选取合适的算法并结合实际情况进行创新改进，训练模型，并将其用于平台进行僵尸企业的识别。同时平台提供企业的画像分析，以及同行的对比和对特征进行可视化处理。

1.5 项目价值

为了整合更多的资源、打造更大的平台、提供更好的服务，作为经济实体，企业需要找寻合适的合作伙伴。而企业间的贸易合作处处隐藏着商业风险，为了尽可能的规避这些风险、找到最合适的合作伙伴，合作方的背景信息、风险信息、经营信息、对外投资信息、知识产权信息等企业信息是企业投资前的首要参考信息。对于企业自身来说，更需要时刻掌握自身发展现状和未来发展动向，认清自身发展的利弊，不断完善企业自身，从各方面提升企业品牌形象。对于广大网民而言，也有浏览企业各维度信息的需求。政府要时刻监管市场、监管企业，确保企业的良性运作。

我国目前由于社会的快速发展，产业的升级转型，出现了大批的僵尸企业，并且它们的存在严重的破坏了市场机制，同时也对于其它的非僵尸企业具有投资挤出效应，加重了社会负担，不利于市场的健康发展，因此识别企业是否为僵尸企业，对于企业本身而言，可以提醒其分析本质原因，帮助内部的改造升级。对于相关市场监管部门来说，无疑的是更为重要的，为了维持正常的市场秩序。对信贷机构，也是同样如此，如果能够准确判定该企业能否为僵尸企业，这样就可以将资金投入到合适的地方，高效利用资金。

对僵尸企业传统的判定方法，并不是十分高效，同时具备很强的主观色彩，所以结果往往不尽人意。近年来，进入大数据时代，我们能够根据海量企业行为数据，尽管数据可能存在缺失值，无效值，但我们对数据进行预处理以及特征构建，然后进行模型训练，进而用训练好的模型预测出企业类型。同时，我们对于企业本身，也做了特征可视化操作，能够为用户提供更加直观的感受。

因此，设计一款多维度、全方面的企业画像和僵尸企业识别系统显得尤为迫切。

二、市场分析和定位

2.1 基于PEST模型的市场宏观分析

2.1.1 政治环境

目前国家大力推进僵尸企业的处置工作，出台了一系列的文件。国家发展改革委、工业和信息化部等11部门在2018年联合发布《关于进一步做好“僵尸企业”及去产能企业债务处置工作的通知》，要求积极稳妥处置“僵尸企业”和去产能企业债务，加快“僵尸企业”出清，有效防范化解企业债务风险，助推经济提质增效。

通知明确了分类处置僵尸企业，并且强调严禁政府通过财政补贴位置僵尸企业的行为。通知提出，地方各级相关部门应按照处置范围并结合实际情况，定期确定需开展债务处置，原则上应在2020年前完成全部的处置工作。 在处置方式上，通知明确，分类处置“僵尸企业”和去产能企业的直接债务。依据“僵尸企业”和去产能企业的营业价值、债务清偿能力、资产负债状况等因素，按照相关法规，分别采取破产清算、破产重整、债务重组、兼并重组等方式分类处置其直接债务。对具备清偿能力的去产能企业积极进行追索，切实防止恶意逃废债行为。

通知提出，仍有部分营业价值的“僵尸企业”、资产负债率高于合理水平且偿付到期债务有困难的去产能企业，鼓励通过金融债权人委员会机制与债权人自主协商开展资产、债务和业务重组，支持引入战略投资者推动兼并重组。各相关利益方应在“僵尸企业”名单确定后六个月内协商一致形成重组方案。符合破产重整条件的“僵尸企业”，相关部门应积极推动进入重整程序，符合破产清算条件的“僵尸企业”，应坚决破产清算。

全国各地响应政府号召，进行僵尸企业的处置工作，但是僵尸企业的识别判定是头等问题，因此本项目的价值便凸显出来了，能够为相关部门提供企业画像和僵尸企业的识别判定具有强有力的指导意义。

2.1.2 经济环境

对于企业和公司来说，以可承受的利率借钱并非坏事。借到了钱，增加了债务，使得企业能建立新工厂、购买新设备、投资研发项目……各国采取宽松的货币政策给全球经济注入了活力，激发了投资者们的信心，这是为经济的未来构建宏图。然而，僵尸公司的存在却是彻底的资源浪费。Hartnett解释道：“让那些本不该继续的公司得以生存，其实是使用了大量资本和劳动力等资源，而这些资源在其他地方本可以被有效使用。”僵尸公司对于资本市场的吸取和压榨只会慢慢蚕食社会经济。去年9月来自国际清算银行（Bank for International Settlements）的经济学家发表论文，量化了僵尸公司的崛起对世界经济带来的负面影响：当一个经济体中的“僵尸”份额增加1%时，生产率增长率就会下降约0.3个百分点。

这个现象被业界称为“负僵尸拥塞效应”，即僵尸公司的生产率低，挤占了投资者对高生产率企业的投资和就业。显然，这样的现象是极不健康的。可是这些“僵尸”似乎并不为此担心：在上世纪80年代末，发达经济体中只有2%的企业是僵尸企业；到如今这个数字攀升到了13%。这些企业宁愿“苟延残喘”地长期维持下去，也绝不愿宣布破产摆脱僵尸状态。

由此看来僵尸企业对于社会经济的影响是巨大，更多的是偏向于负面的，在发达国家也不例外，多数情况下，成熟市场经济国家都能在衰退后逐步复苏，但是日本这种长达十多年的经济发展停滞则相对少见。经过研究和反思，现在多数经济学家赞成“僵尸企业”的存在和治理不善是压制日本经济改善的一个重要原因。

严谨的讲：就是说“僵尸企业”及其治理不善不是阻碍日本经济复苏的全但由于日本在治理“僵尸企业”的过程中，没有果断采取市场出清的方式，助长“僵尸企业”的数量进一步大幅增加，恶化了全社会资源配置的效率，阻碍着经济的复苏。最后形成一个恶性循环：“僵尸企业”增加，复苏困难；复苏困难，“僵尸企业”增加。据日经Needs统计，1989年日本仅有4%的企业属于“僵尸”，而到2001年时这个比例已达到了25.5%，12年间增长了近6倍。与此对应，日本经济长期不景气。根据日本产业生产率数据库（JIP database）2015年的测算结果，90年代日本全要素生产率增速仅为0.02%，比80年代的1.41%下降了1.39个百分点。全要素生产率增长持续低迷将日本经济拖入长时间的萧条之中。同期日本的GDP增速从1989年的4.85%跌至2001年的0.41%（世界银行数据）。

对应到我国具有很强的参考意义，如果想要我国经济健康的增长，对于僵尸企业的处理不容轻视，当今我国已经成为世界第二大经济体，即将成为第一大经济体，看到僵尸企业对于各个发达国家危害，更应该谨慎的处理僵尸企业。

2.1.3 社会环境

要治理“僵尸企业”，首先需要识别哪些是“僵尸企业”，哪些不是？而这并不容易做到。这从学者们对其的定义就可管窥一二。对于“僵尸企业”的定义，共性在于“僵尸企业”主要是依靠政府补贴、银行贷款或畸形权益市场融资（畸形市场会对无效率低效率的企业提供权益资本，因为提供权益资本是为了赌博，而非为了追求股东回报）才能继续生存，使本应从市场退出的非效率企业。

悖论在于：政府和银行的输血行为既可能造就“僵尸”，也可能挽救一些暂时困难的企业帮助其咸鱼翻身（国内经常探讨银行应该雪中送炭就是认为银行的帮助可以让困难的企业死而复生）。那么要事前确认企业是否沦为“僵尸企业”非常困难，几乎没有现实可操作的方法。

对于个体和和银行等信贷机构要面临各种各种的企业选择是否进行投资或贷款，但是作为投资者往往都是想获取回报，但是如果对于企业无法进行精准的判断，将资金放入了僵尸企业中，根据僵尸企业的特点我们知道，那么将会变得投资失败，出现亏损，同时更严重的是，对于社会上的非僵尸企业，资金便被分流，对社会健康成长抑制。

因此，僵尸企业的识别显的尤为重要。

2.1.4 技术环境

人们发明了CHK方法和增进版的FN-CHK方法来识别，但这依然不是具备现实可操作性的、可信赖的方法。因为“僵尸企业”的边界相当模糊不清。举个例子：如按照这两种识别方法，在过去20年相当长的时间段里，日本的东芝和索尼都高度疑似“僵尸企业”。现如今，东芝步履艰难，“僵尸形态”未改，而索尼则发生了转机，但我们依然不能确定索尼不是“僵尸企业”，而东芝一定是。

但是随着大数据时代的到来，数据凸显的价值伴随而来，可以根据海量的数据，企业的各种行为特征，进行特征构建，而且各类算法逐步变得完善，具备更强的识别预测能力，如XGBoost、Random Forest等，根据海量的数据集，进行模型的训练，进而得出一个对于僵尸企业的识别具有很好效果的判别器。

设计的系统，前后端时候才采用当前主流的技术框架，前端采用HTML+CSS+JavaScript为基础，使用Vue框架进行方便的数据绑定与更新，并实现视图层与模型的交互。此外，我们使用Element框架来美化Web界面的UI效果，通过Chart.js框架来获得简洁优美、交互性强的动态图表。后端主要由Python语言实现，分为服务器层(Nginx)、中间件层(WSGI)和应用程序层(Flask)三个部分。

2.2 市场定位

目前市场上都是暂时没有该类型的系统，能够根据企业的特征信息直接得出企业的类型(即是否为僵尸企业)，同时给出企业的画像，然后可视化给出标签，分析与同行之间的差距，给用户进行参考，所以具有很大的市场前景。

目标用户：

* + 市场散户：用于判断该企业是否具备前景，能否带来回报。
  + 独立投资人：同市场散户一样，作为投资参考依据，如果是僵尸企业那么再仔细考虑企业其他方面，再决定是否进行投资。
  + 政府相关监管部门：如果市场中存在过多的僵尸企业，无疑会扰乱市场秩序，但是面对大量的企业判断他们是否是僵尸企业，成为一个难题，因此该系统为政府相关监管部分，提供强有力帮助。
  + 银行及信贷机构：如果将资金贷给僵尸企业，那么势必非僵尸企业获得的资金将会变少，更严重的是僵尸企业往往偿还能力极差。
  + 企业自身：用于判断企业自身情况，进行内部架构升级，及时调整方向，防止变成僵尸企业。

三、可行性分析

3.1 技术可行性

* 算法可行性

在大数据时代，随着数据挖掘与人工智能相关技术的不断成熟，越来越多的机器学习算法被用在各种分类与回归问题之中。在股价预测、图片分类、兴趣推荐等实际应用领域，基于统计学习和神经网络的模型取得了不俗的效果，大大促进了相关产业的发展。本项目中的僵尸企业分类为一个常见的半监督二分类问题，针对这类问题的机器学习研究已经相当成熟，人们根据不同的数据特点和研究侧重，提出了许多准确高效的分类算法。这些算法为我们在本项目中的进一步研究与改进提供了基础与保障。

* 开发可行性

为了算法模型的应用落地，在本项目中我们需要开发一个简易的Web交互界面供用户使用。我们在开发过程中基于当前主流的前端技术，选用了时下最常用的JavaScript框架Vue，该框架上手容易，社区成熟，能够大大提升我们的前端开发效率。在后端方面，由于我们的分类模型是基于Python语言实现，我们选择了同样用Python实现的Flask框架，这有助于服务器与模型之间的交互，为Web后端的高效开发提供了便利。

3.2 经济可行性

项目的具体实施可以分为分类模型设计和交互界面开发两部分。在这两部分工作中，我们普遍选择了开源的程序设计语言、开发工具和框架，因此，项目的软件成本开销是很低的。硬件方面，训练本项目中所使用的机器学习模型需要一定的硬件支持，但相比神经网络等复杂算法，对GPU的需求不大，使用一般的家用计算机即可满足基本要求。此外，项目指导老师所负责的实验室中有大量的硬件资源可供使用，这大大缩减了项目实施的硬件成本。

在信息化、智能化时代，社会信用体系不断建成完善，对企业进行信用评估和画像分类的重要性也逐渐凸显，这为本项目所设计的系统提供了广阔的潜在市场。本系统上手容易、操作简便、可扩展性强，在针对实际需求和特定领域进行优化后，可被政府监管部门、银行信贷构使用以描述企业信用和风险状况，有很大的应用前景和经济价值。

由此可见，项目的经济价值远远高于其实施成本，因此项目在经济上是可行的。

3.3 操作可行性

在系统的整体设计和Web界面开发过程中，我们尽可能的简化了操作使用逻辑，用户通过页面提示的简单步骤即可完成系统的基本功能。在实际的设计开发过程中，我们遵循下面几项原则以确保系统的操作可行性：

* 简单性：在界面设计过程中，只保留核心交互逻辑，删除繁琐复杂操作。通过简明扼要的文字引导帮助用户使用系统进行操作。
* 高效性：在保证模型精度的基础上，尽可能选用运行速度更快的模型，以缩短用户的等待时间，提升系统使用体验。
* 鲜明性：更多的使用色彩鲜明的图片和表格来呈现数据，用户不再需要面对大量的数字和符号，方便了用户对模型预测信息的获取和认知。

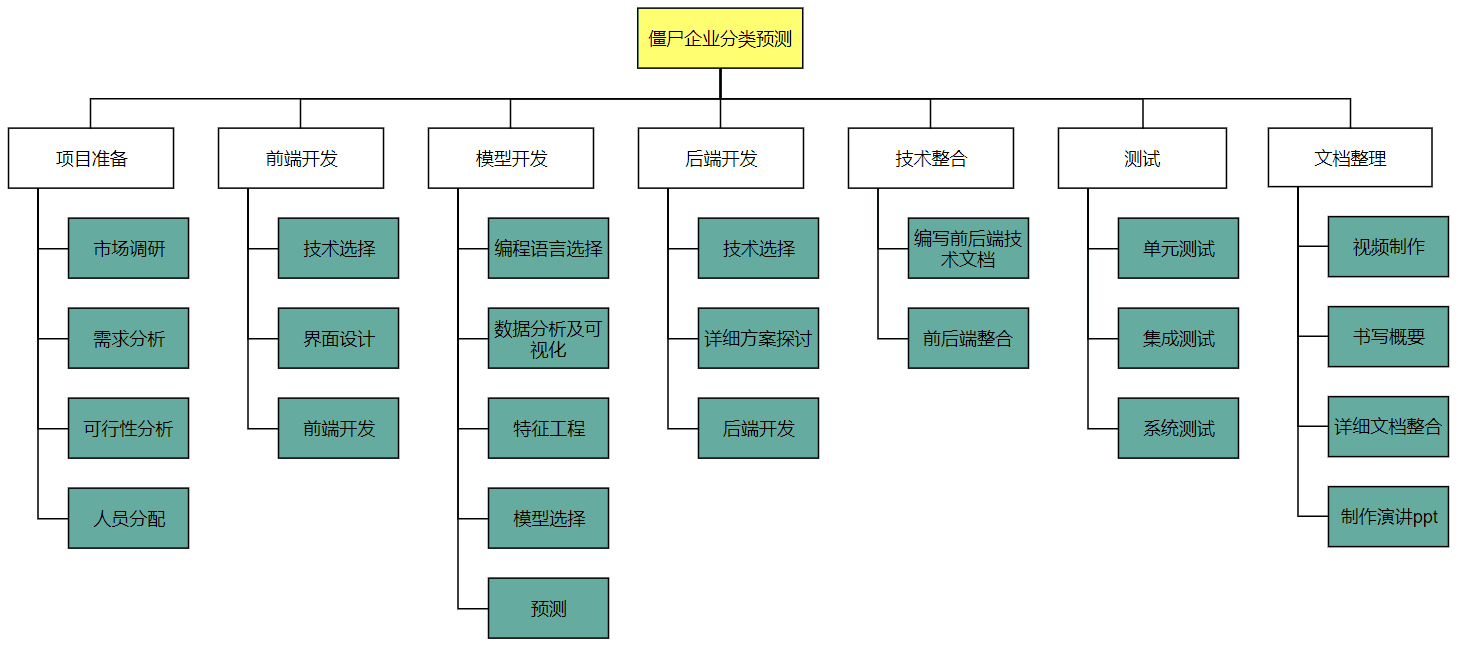
3.4 人员可行性

项目团队由计算机专业的同学组成，成员在校期间有数据挖掘和机器学习方面的学习实践经验，能够针对特定问题进行探索和研究。团队人员分工明确，各有所长，富有创新意识，能积极地参与到研究工作中来。此外，为了更好的分析项目中所用到的企业数据，我们专门挑选了对经济学有一定了解的同学加入团队，这对项目的特征工程和模型解释等工作提供了帮助。 项目由经验丰富的专业老师指导，实验室有项目实施所需的各类资源。这些条件为项目的顺利进展提供了充分的人员和软硬件支持。

四、组织管理与业务分析方案

4.1项目开发计划

4.1.1 WBS工作分解



图X 项目WBS工作分解图

该项目的工作分解结构如上图所示。整体上可分为项目准备、前端开发、模型开发、后端开发、技术整合、测试以及文档整理七个阶段。此外，我们详细设计了每个阶段的子任务：

项目准备阶段可分为市场调研，需求分析，可行性分析以及人员分配四个子任务。在这个阶段里，我们重点分析调研了市场上已有的算法框架，思考其优势与短板，并加以总结提炼与学习，主要查找了关于僵尸企业的一些重要特征，方便后续特征工程工作的进行，着重翻阅了半监督问题的文献资料，解决数据提供的半监督问题，为半监督问题提供一个良好的解决方案。

后端开发阶段可分为技术选择、详细方案研讨以及后端开发三个子任务。在这个阶段里，后端开发人员对后端分层进行了调查和处理，最终决定以python作为主要开发语言，开发过程中包括了服务器层，中间件层和应用程序层的开发。

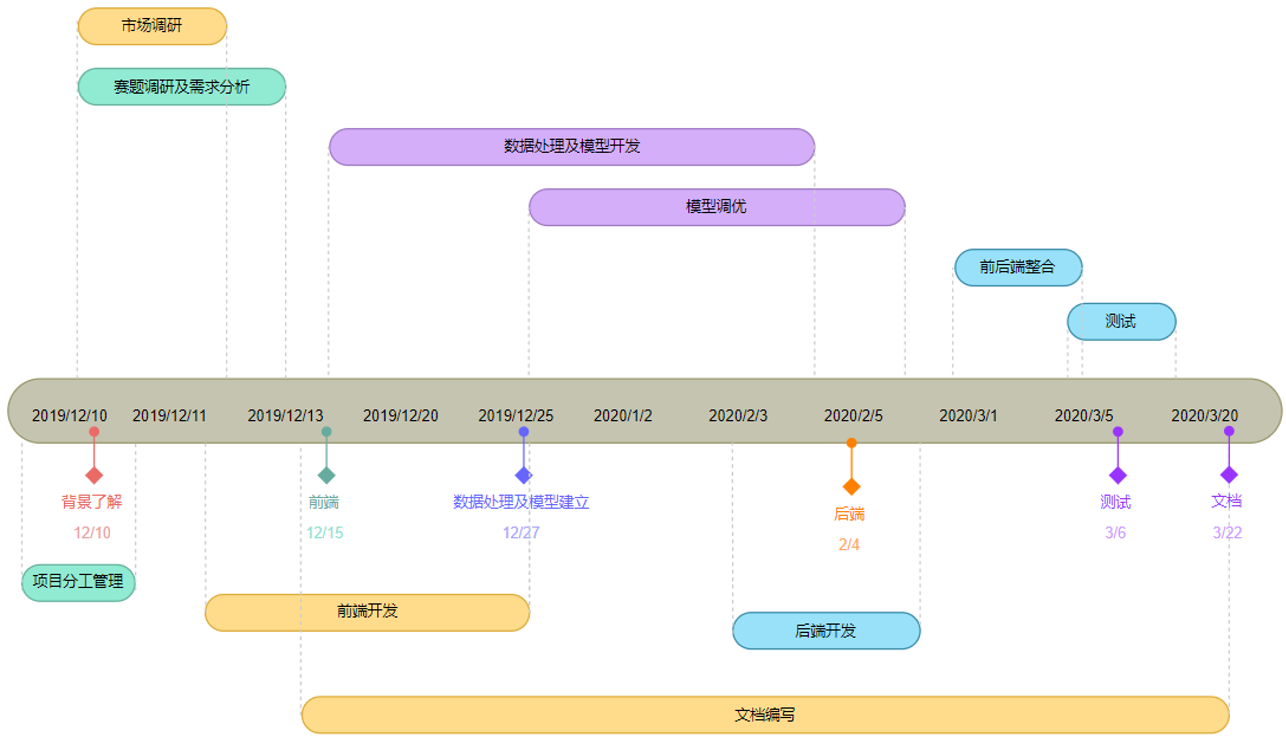
前端开发阶段可分为技术选择、界面设计、前端开发三个子任务。通过利用现有的前端框架Vue结合前端主流使用的HTML+CSS+JavaScript开发出具有用户交互功能的前端界面。

技术整合阶段则可分为前后端结合以及提交前后端技术文档两个子任务。

测试阶段可分为单元测试、集成测试、系统测试三部分子任务。在这个阶段里，我们根据测试结果不断地优化模型、修改细节。

文档整理阶段可分为视频制作、书写概要文档、详细文档整合、制作演讲PPT四个子任务。

4.1.2 时间线



图X 项目进展时间线

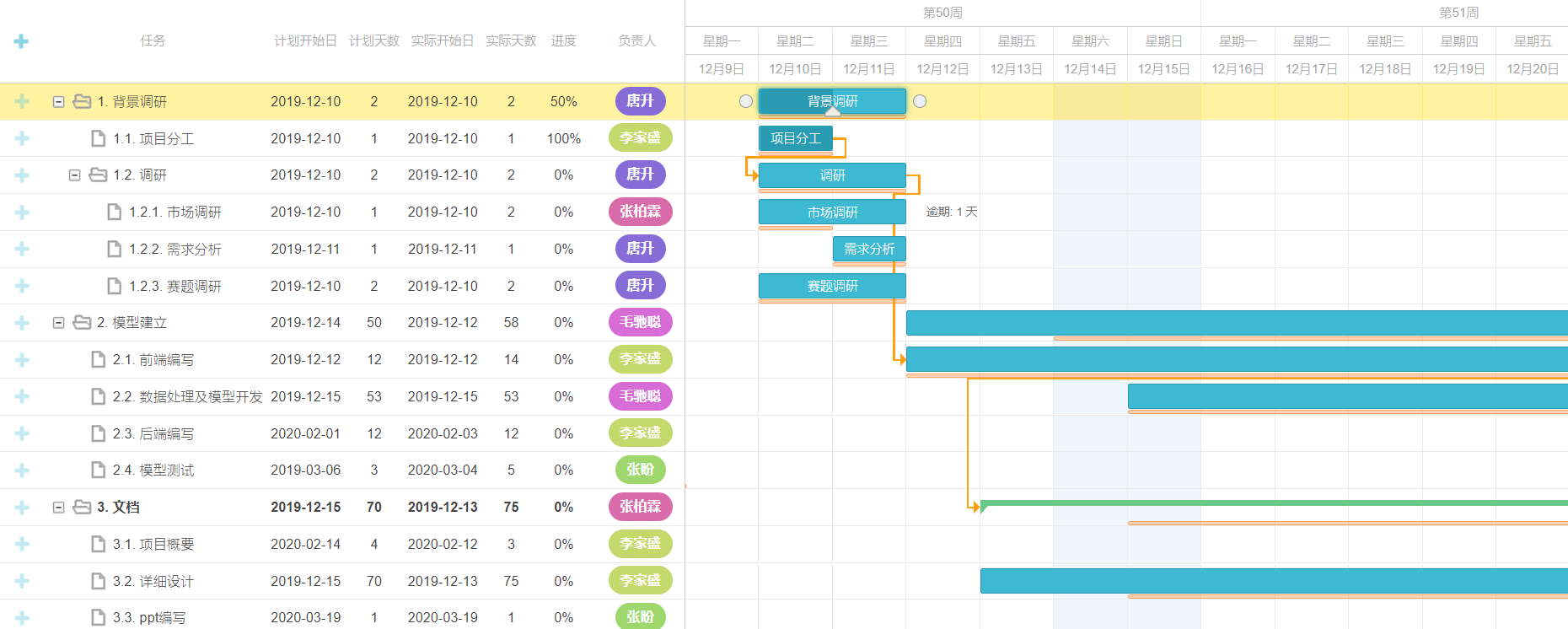
4.2 人员的组织及分工

|  |  |  |
| --- | --- | --- |
| 成员 | 角色 | 工作职能 |
| 李家盛 | 项目负责人 | 有效利用资源,组织实施项目,合理安排，完成项目开发，特征工程，前后端制作，概要制作。 |
| 唐升 | 数据分析师、市场分析师 | 市场分析，需求分析，部分文档编写，特征调研及分析，视频制作 |
| 毛驰聪 | 算法模型建立者 | 算法编写研究，核心技术及实现方案选择，特征工程及数据预处理，模型调参 |
| 张盼 | 测试工程师、文档审稿人 | 前后端测试，产品测试，文档编审，PPT制作 |
| 张柏霖 | 数据分析师、市场分析师 | 文献查找调研，需求分析，部分文档编写，算法调研及分析。 |

4.3项目管理

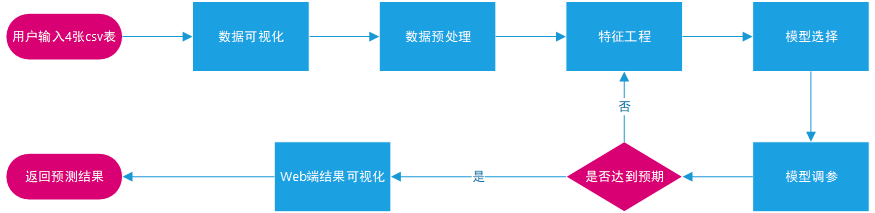
本项目的项目管理中应用了甘特图。

甘特图就是把项目分解细致，通过甘特图展示给管理者，管理者可以清晰详细的看到任务的进展，以及过程中延误的原因，以及延误者，可以及时解决延误的原因，从而达到项目结束的时间，保时保量的完成。项目一共持续进行了4个月左右。具体甘特图如下：



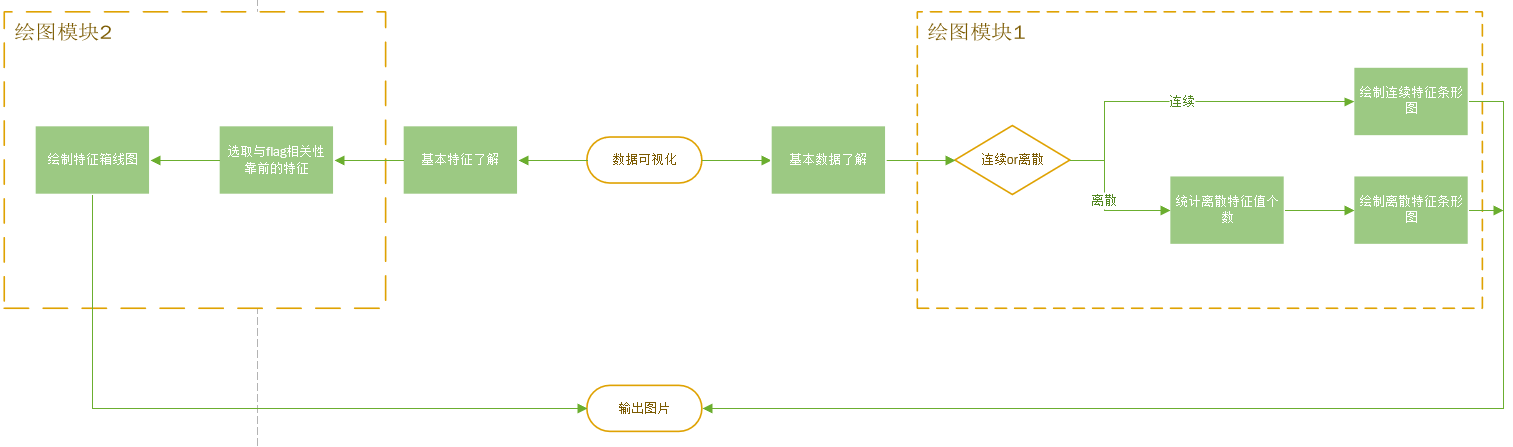
图X 项目管理中使用的甘特图（部分）

五、解决方案



图X 项目解决方案流程图

5.1 数据可视化



图X 数据可视化流程图

为了对赛题主办方提供的数据有一个基本的了解，我们对所有csv文件进行了数据可视化处理。首先通过pandas库中的head方法查看每个特征的数据类型，再用describe方法对每张表对特征进行一个基本刻画，发现每张表内的所有特征都有一定数量的NAN。

于是我们经过缺失值计数对每个csv文件中特征的缺失值进行了一个统计。得出了一个不但对于所有表内的所有特征的缺失值分布都十分的平均，同时对于flag值为1，0和NAN的限制条件下，特征的缺失值分布也十分平均。

接着对离散特征进行特征值的计数，发现每个特征的特征值分布亦是平均的，得出暂时不需要对特征进行删除，可能要做一些特征上的拆分合并结论。

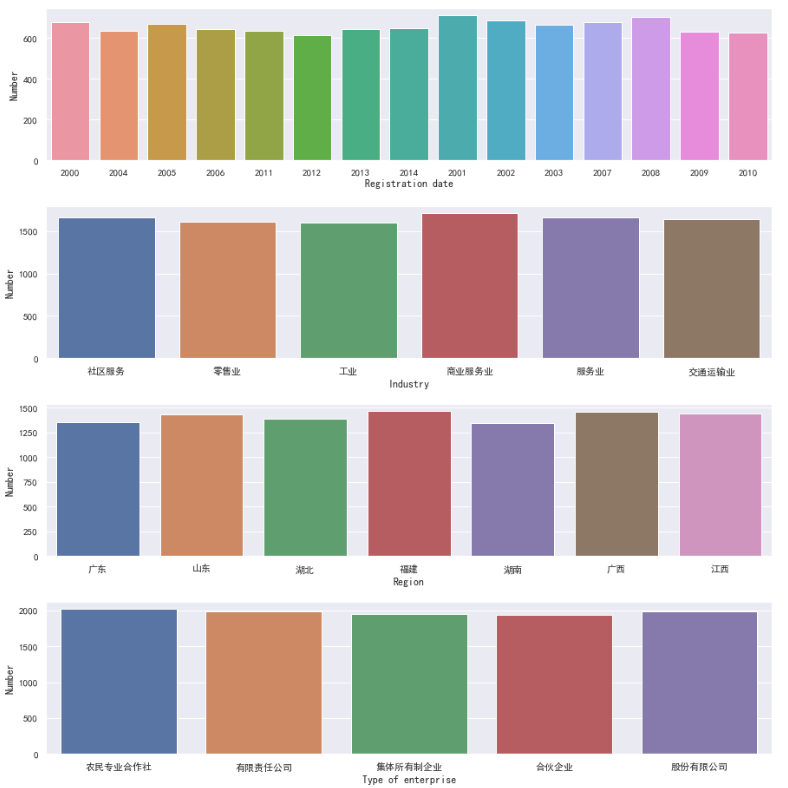
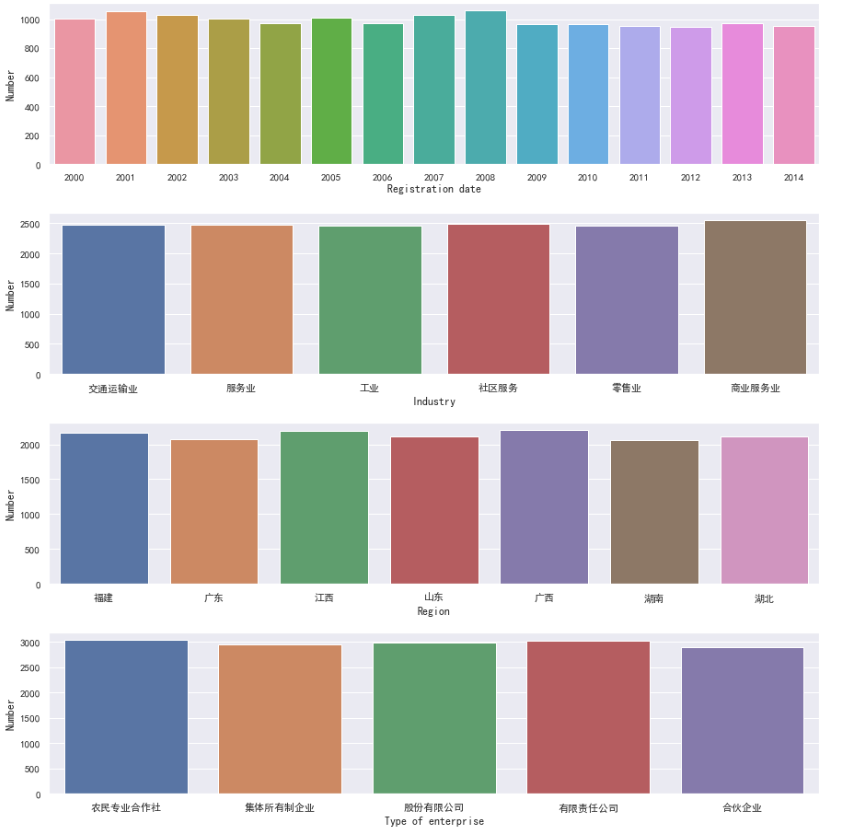
再经过对训练集和测试机各特征对flag的相关性刻画，降序输出特征对flag的相关性。对排序靠前的特征进行箱线图的可视化发现这些特征分布与均值有合理的偏差，可以作为理想的训练强特征。

最后我们还做了四张表内特征之间的相关性分析，发现对于融资表中的对应融资和负债的相关性为1，以及年表中营业收入和主营业收入相关性达到了0.95以上。所以我们在特征工程中可以对融资负债这8个特征进行合并成4个特征，并且对营业收入和主营业收入其中之一进行合并。

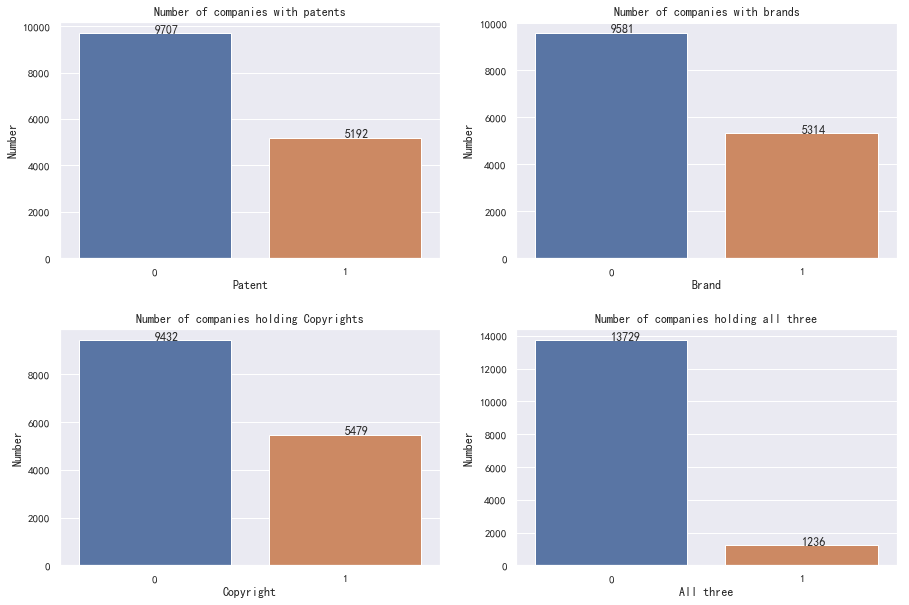
5.1.1 特征条形图

分别对离散型和连续型特征进行可视化，有助于特征工程的顺利实施。

* 离散型特征：在进行离散型数据绘图前，先对离散型特征值进行一个计数，若特征值类别十分的多，则不宜用条形图的形式表现出来，直接通过数字的形式观察。我们将离散型特征分所有企业，有标签企业，无标签企业对四张融资表，年报表切成三批做可视化。我们发现三批所有特征分布结果基本一致且十分平均，图中横坐标为特征值，纵坐标为对应特征值的数量。下面对部分结果进行展示。

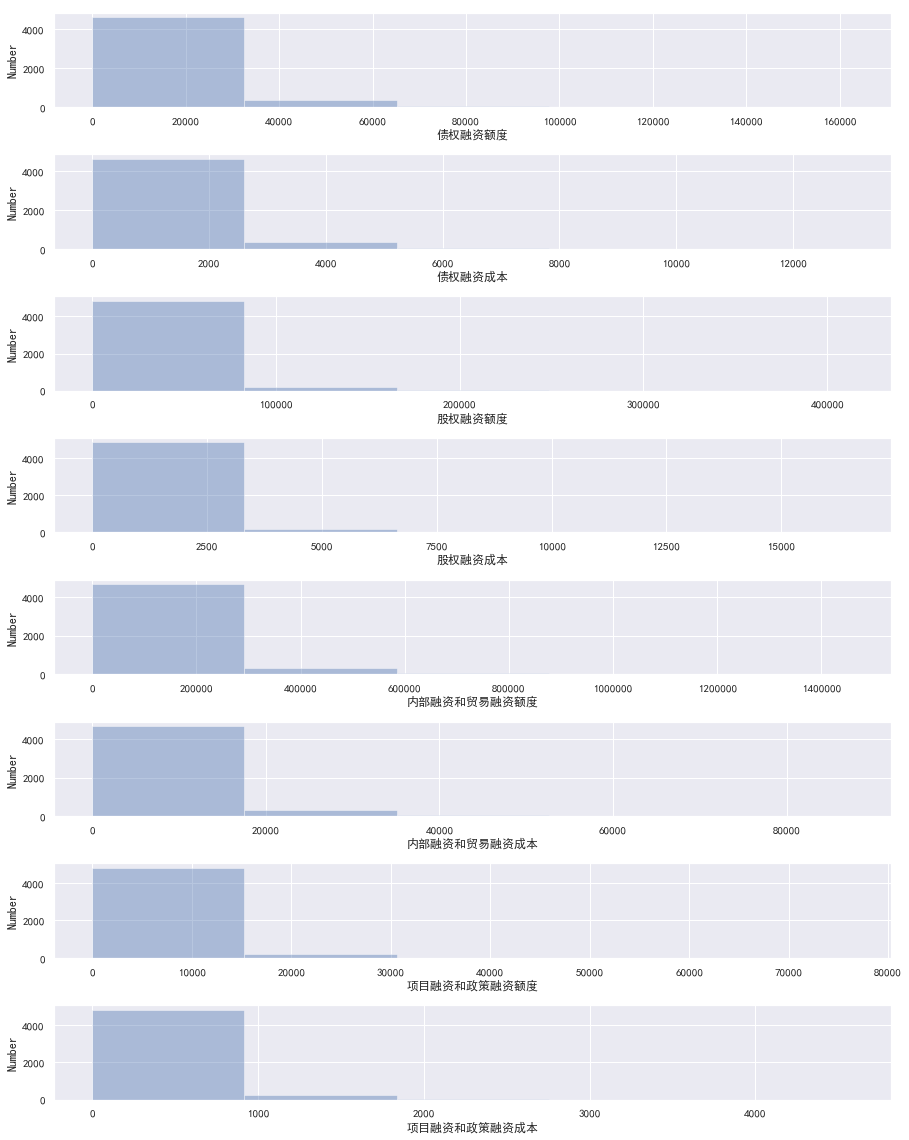


图X 训练集基本信息表特征图（左：正例样本，右：无标签样本）

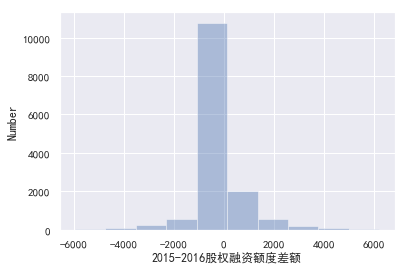
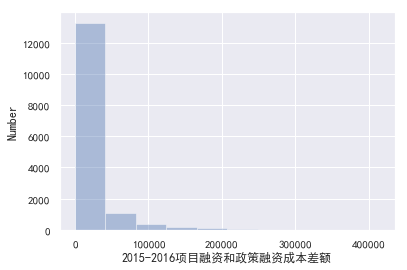


图X 训练集知识产权表特征图（仅包含正例样本）

* 连续型特征：将连续性特征分所有企业，有标签企业，无标签企业对四张融资表，年报表分成三批做可视化。图中横坐标为特征值，纵坐标为对应特征值的数量。



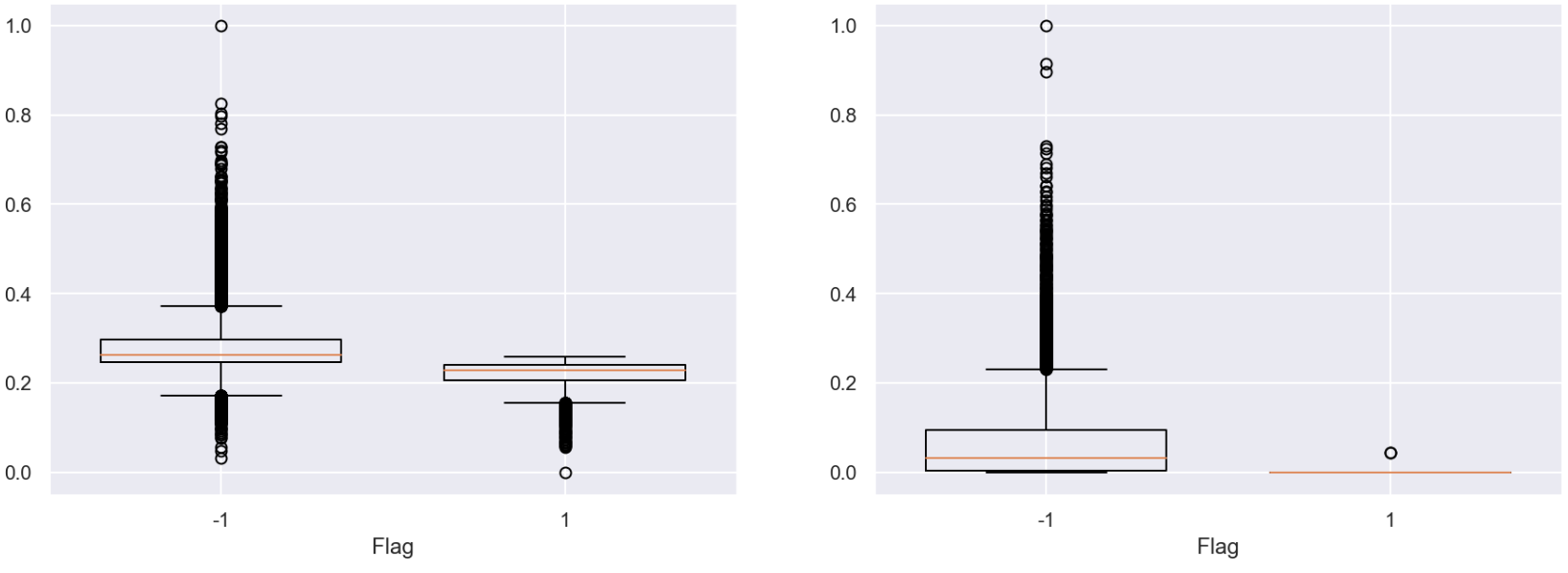
图X 训练集融资表特征图（仅包含正例样本）



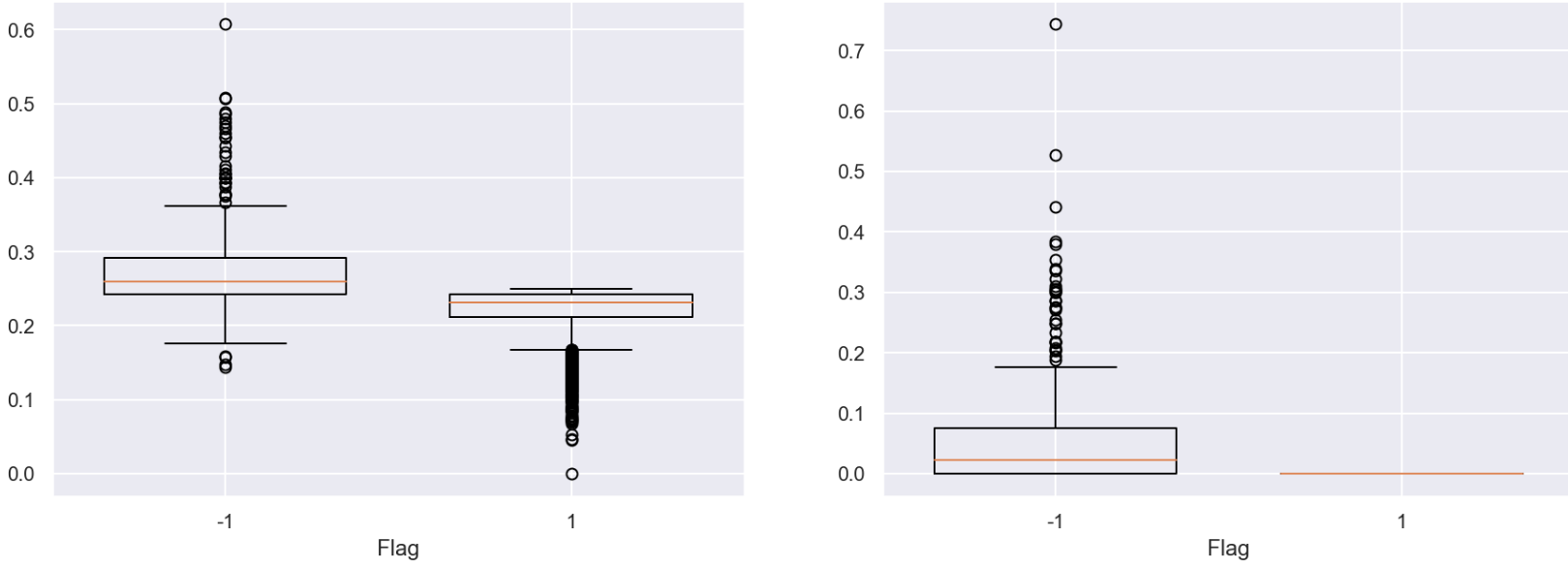
图X 训练集融资表特征差额图（仅包含正例样本）

5.1.2 箱线图

对每个特征与flag做相关性计算，做降序排序后把correlation值靠前的特征使用箱线图可视化，如下图所示：



图X 验证集年表中净利润特征图



图X 验证集年表中纳税总额特征图

5.2 数据预处理

在工程实践中，我们得到的数据会存在有缺失值、重复值等，在使用之前需要进行数据预处理。数据预处理没有标准的流程，通常针对不同的任务和数据集属性的不同而不同。数据预处理的常用流程为：去除唯一属性、处理缺失值、属性编码、数据标准化正则化、特征选择、主成分分析。在这个竞赛题目中也同样存在缺失值和重复值，并且为了更方便的使用Python进行模型训练，我们对数据进行了预处理，主要包括以下几个部分。

5.2.1 四表联合

该题目给出的训练集和验证集各有四个表格，每个表格的信息都不相同，前两个表格都是关于公司本身的属性信息，后两个表格都是关于公司经营情况的信息。为了方便后面对数据进行统一处理，我们将四个表格通过一定的处理合成一个表格。

在对数据可视化的过程中，我们发现前两张表每个企业只有一条数据，后两张表每个企业都有三条数据，分别表示不同年份公司的经营状况，通过网上查询资料以及小组内部讨论，我们首先将后两张表进行改造，把一个公司的三条记录转化为一条包含该公司2015和2016数据差值、2016和2017数据差值以及三年数据平均值这三个属性的记录，使得其与前两张表结构一致。然后把这张新构建的表和5两张表合成一个表格，这样每个公司就对应一条数据，方便训练模型。

5.2.2 重复值处理

如果数据中存在特征完全一致，并且标签也相同的数据，我们把它看作完全重复的数据，只保留一条数据；而对于那些特征完全一致，5标签不同的数据，我们把它看作有误的数据，因此删除所有此类数据。

5.2.3 缺失值处理

因为数据的缺失率在1%左右，我们决定对缺失值进行填补，对“注册时间”、“行业”、“区域”、“企业类型”等离散型数据填补众数，对其他特征填补平均值。此外，为5接下来程序方便处理数据，我们将缺失的flag标签值都赋值-1。

5.2.4 独热编码

独热码，也叫 One-Hot 编码, 直观来说就是有多少个状态就有多少比特，而且只有一个比特为1，其他全为0的一种码制。举例如下：假如有三种国家特征：中国、美国、法国。使用机器学习的算法一般需要对这个特征进行数字化或向量化。我们可以令中国=1，美国=2，法国=3，这就是最简单的标签编码，即给不同的类别分配一个数字标签，但是这样做会存在一个问题，“中国”和“美国”的距离比“中国”和法国的距离小，但这并不是我们想让机器学习到的知识。这时使用标签编码会产生一定的误差，需要对其进一步转化。因为有三个国家，所以就有三个比特，即中国：1 0 0，美国：0 1 0，法国：0 0 1。这样一来任何两个向量之间的距离都是相等的，就不会出现偏序性，不会影响基于向量空间度量算法的效果。

对于“行业”、“区域”、“企业类型”等离散类特征，它们之间的距离没有实质性的意义，为了避免对最后的预测结果产生误差，影响模型精度，我们对这些离散类特征进行了独热编码。

5.2.5 归一化处理

数据的标准化是将数据按比例缩放，使之落入一个小的特定区间。在某些比较和评价的指标处理中经常会用到，去除数据的单位限制，将其转化为无量纲的纯数值，便于不同单位或量级的指标能够进行比较和加权。

目前数据标准化方法有多种，归结起来可以分为直线型方法(如极值法、标准差法)、折线型方法(如三折线法)、曲线型方法(如半正态性分布)。不同的标准化方法，对系统的评价结果会产生不同的影响。常用的归一化方法有以下两种，0-1标准化和Z-score标准化。

0-1标准化通过遍历特征向量里的每一个数据，将最大值和最小值记录下来，并通过Max-Min作为基数（即Min=0，Max=1）进行数据的归一化处理。

Z-score标准化给予原始数据的均值和标准差进行数据的标准化。经过处理的数据符合标准正态分布，即均值为0，标准差为1，这里的关键在于复合标准正态分布，这在一定程度上改变了特征的分布。

为了消除数据特征之间的量纲影响，我们同样需要对该数据特征进行归一化处理，使不同指标之间具有可比性。在本次问题中，我们采用了0-1标准化的方法。

5.2.6 缺失标签值处理

在现实生活中，有许多只有正样本和大量未标记样本的例子。这是因为负类样本的一些特点使得获取负样本较为困难。比如：

* 负类数据不易获取。
* 负类数据太过多样化。
* 负类数据动态变化。

因此，有研究人员开始关注PU-Learning，即在只有正类数据和无标记数据的情况下训练分类器。由于标准机器学习问题（利用大量正面和负面样本用于训练模型）有了充分的发展，因此在监督学习的基础上，有许多方法经过巧妙调整可以来进行PU-learning。

2.1 直接利用标准分类方法

直接利用标准分类方法是这样的：将正样本和未标记样本分别看作是positive samples和negative samples, 然后利用这些数据训练一个标准分类器。分类器将为每个物品打一个分数（概率值）。通常正样本分数高于负样本的分数。因此对于那些未标记的物品，分数较高的最有可能为positive。

这种朴素的方法在文献Learning classifiers from only positive and unlabeled data KDD 2018中有介绍。该论文的核心结果是，在某些基本假设下（虽然对于现实生活目的而言可能稍微不合理),合理利用正例和未贴标签数据进行训练得到的标准分类器应该能够给出与实际正确分数成正比的分数。

2.2 PU bagging

一个更加复杂的方法是bagging的变种：通过将所有正样本和未标记样本进行随机组合来创建训练集。利用这个“bootstrap”样本来构建分类器，分别将正样本和未标记样本视为positive和negative。将分类器应用于不在训练集中的未标记样本 - OOB（“out of bag”）- 并记录其分数。重复上述三个步骤，最后为每个样本的分数为OOB分数的平均值。这是一种bootstrap的方法，可以理解为之前我们会想到随机抽取一部分未标记样本U作为负样本来训练，在这里会设置迭代次数T，根据正样本的个数，每次都随机可重复地从U中选取和P数量相同的样本作为负样本N，并打上标签，每次迭代都重复进行取样->建模->预测的过程，最后的预测概率使用T次迭代的平均值作为最终预测的概率。

参考文献：A bagging SVM to learn from positive and unlabeled examples ，PRL 2014

2.3 两步法

大部分的PU learning策略属于 “two-step approaches”。该方法的思想也很直观：识别可以百分之百标记为negative的未标记样本子集（这些样本称为“reliable negatives”。）所谓的百分之百只是一个夸张的说法，通常我们可以用正样本和未标记样本训练一个模型，然后对未标记样本进行预测，按照概率排序，选取前面的样本作为reliable negatives。使用正负样本来训练标准分类器并将其应用于剩余的未标记样本。通常，会将第二步的结果返回到第一步并重复上述步骤。即每一轮循环都会找出那些所谓百分之百的正样本和负样本，加入到训练集里，重新预测剩余的未标记样本，直到满足停止条件为止。

在本项目给的数据集中，同样只有少许正例样本和大量未标记样本，因为模型学习必须需要标签值，未标记样本的真实标签未知，所以我们小组采用pulearning来确定未标记样本的标签。根据论文结果显示，随着已知正样本比例的减少，PU bagging最好，Standard最差，两步法居中。其中两步法效果不错，并且训练过程相较PU bagging简便，所以我们采用了两步法来实现缺失标签的填补。

5.3 特征工程

5.3.1 基本特征处理

基本特征指可以从数据表中直接获取的特征。在本项目中，4张数据表中的特征以连续型为主。对于基本信息表中的几个离散型的中文特征，我们在数据预处理时对其进行One-Hot编码，使之转化为数值类型特征。对于融资表和年表，每张表中都包含了企业在2015、2016和2017三年内的数据，我们分别计算三年数据的均值、2015-2016年数据的差额和2016-2017年数据的差额作为各个企业的基本特征。最后，我们将分别经过处理后的4张表融合保存为一张基本特征表。

5.3.2 交叉特征

交叉特征是在特征工程中十分重要的一个方法，它将两个或以上的特征通过数学计算组合成一个。在模型中使用这些组合生成的特征往往会比使用原始特征的效果更好。在进行特征交叉时，常见的数学运算除了加减乘除，还有针对离散型数据的笛卡尔积，针对字符数据的拼接操作等。

虽然交叉特征在实践中通常能取得显著的效果，但是如果只是通过简单的排列组合后来生成特征，也会导致数据维度爆炸的问题。这样不仅产生了大量的冗余无效特征，也会极大的降低模型运行效率。所以，在进行交叉特征的实验时，也十分需要主观经验上的指导和后期特征选择的工作。

在本项目的交叉特征处理中，我们首先选用了4个在经济学和企业经营状况分析中常用的指标：

* 销售净利率

计算公式：销售净利率 = 净利润 / 营业总收入

销售净利率反映每一元销售收入带来净利润的多少，也反映投资者从销售收入中获得收益的比率。销售净利率低说明企业管理当局未能创造足够的销售收入或未能控制好成本和费用。

在生成这个特征之后，我们分别查看其在正类和负类数据上的分布情况：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 均值 | 标准差 | 最小值 | 25% | 50% | 75% | 最大值 |
| 僵尸 | -0.100 | 0.013 | -0.618 | -0.100 | -0.100 | -0.100 | -0.003 |
| 非僵尸 | 0.152 | 0.133 | -0.334 | 0.062 | 0.150 | 0.239 | 6.962 |

正负例样本在这个特征上的分布差异非常明显，僵尸企业的销售净利率几乎都为负，这与僵尸企业长期亏损的状况有很大的关联；而非僵尸企业的销售净利率除了极少数为负，大多数为正，这也符合我们对正常企业盈利的认知。

* 资产净利率

计算公式：资产净利率 = 净利润 / 资产总额

资产净利率反映企业资产的利用效果，指标越高，表明资产的利用效果越好，说明企业在增收节支和加速资金周转方面取得了良好效果。

在生成这个特征之后，我们分别查看其在正类和负类数据上的分布情况：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 均值 | 标准差 | 最小值 | 25% | 50% | 75% | 最大值 |
| 僵尸 | -0.255 | 0.093 | -1.036 | -0.320 | -0.253 | -0.187 | -0.007 |
| 非僵尸 | 0.386 | 0.344 | -0.407 | 0.129 | 0.338 | 0.598 | 2.457 |

与销售净利率相似，正负例样本在这个特征上的分布差异也非常明显，从数据上看，这是由于绝大部分僵尸企业净利润都为负导致的，但也能充分地反映出僵尸企业在生产经营和资金周转上存在的问题。

* 资产负债率

计算公式：资产负债率 = 负债总额 / 资产总额

资产负债率反映企业全部资产中有多大比重是通过借贷获取的。对投资者而言，负债对总资产的比率越小，表明所有者权益的比率越大，则企业的资金力量越强，资金收益率低。若负债比率过高，而企业状况良好，当然可通过财务杠杆作用使投资者获得较高的报酬率；但若企业状况不佳，利息费用将使之不堪重负，企业会有破产的危险。

在生成这个特征之后，我们分别查看其在正类和负类数据上的分布情况：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 均值 | 标准差 | 最小值 | 25% | 50% | 75% | 最大值 |
| 僵尸 | 1.173 | 0.353 | 0.250 | 0.913 | 1.169 | 1.427 | 3.547 |
| 非僵尸 | 1.183 | 0.356 | 0.188 | 0.924 | 1.179 | 1.437 | 5.083 |

正负例样本在这个特征上的分布差异不明显，从经济学上来解释，资产负债率多用来描述一个企业的投资和经营风险情况，尽管从人们的常识上来看，僵尸企业应该伴随着大量的负债和极高的风险，但在正常的企业中也不乏通过大量借贷抬高杠杆，以获取高额利润的存在。因此，两类企业在这个特征上的差异是不显著的。

* 资金收益率

计算公式：资金收益率 = 净利润 / 所有者权益 （只当所有者权益为正时有意义）

资金收益率反映企业运用资本获得收益的能力。资金收益率越高，说明企业自有投资的经济效益越好，投资者的风险越少，值得投资和继续投资。

在生成这个特征之后，我们分别查看其在正类和负类数据上的分布情况：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 均值 | 标准差 | 最小值 | 25% | 50% | 75% | 最大值 |
| 僵尸 | -3.059 | 6.863 | -129.500 | -2.653 | -1.302 | -0.738 | -0.063 |
| 非僵尸 | 4.843 | 13.854 | -30.280 | 0.588 | 1.735 | 4.250 | 502.500 |

正负例样本在这个特征上的分布差异也十分明显，这个特征同样与企业净利润有很大的关联，可以通过数值的正负性看出僵尸企业和非僵尸企业之间的差别。

除了上述4个已经被广泛使用的指标外，我们还根据相关资料人工生成了下面几个特征：

* 创新能力

将知识产权表中的三个特征相加得到的离散特征，可能的取值有0、1、2、3四种。

* 融资成本总和

将融资表中的所有融资成本相加得到的连续特征。

* 企业规模

参照国家统计局发布的《统计上大中小微型企业划分办法》，我们根据企业的行业、从业人数、营业收入和资产总额进行企业规模划分，划分标准如下：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 行业 | 划分指标 | 计量单位 | 大型 | 中型 | 小型 | 微型 |
| 交通运输业 | 从业人数(X)  营业收入(Y) | 人  万元 | X≥1000  Y≥30000 | 300≤X<1000  3000≤Y<30000 | 20≤X<300  200≤Y<3000 | X<20  Y<200 |
| 工业 | 从业人数(X)  营业收入(Y) | 人  万元 | X≥1000  Y≥40000 | 300≤X<1000  2000≤Y<40000 | 20≤X<300  300≤Y<2000 | X<20  Y<300 |
| 零售业 | 从业人数(X)  营业收入(Y) | 人  万元 | X≥300  Y≥20000 | 50≤X<300  500≤Y<20000 | 10≤X<50  100≤Y<500 | X<10  Y<100 |
| 服务业 | 从业人数(X)  资产总额(Z) | 人  万元 | X≥300  Z≥120000 | 100≤X<300  8000≤Z<120000 | 10≤X<100  100≤Z<8000 | X<10  Z<100 |
| 其他行业 | 从业人数(X) | 人 | X≥300 | 100≤X<300 | 10≤X<100 | X<10 |

在根据实践经验和相关资料生成上述几个特征后，我们也对在4张表中的15个数值型特征进行逐一组合来生成交叉特征。从特征表示的经济学意义上看，我们认为对这些特征进行除法计算得到的比率值更有价值，因此，我们对这15个特征进行两两相除，共得到210个生成特征。

5.3.3 偏离值特征

偏离值特征指单个样本相对于分组内所有样本的某个统计量之间的偏离距离。在计算偏离值特征时，首先根据某个标准进行分类，之后计算各个类别中所有样本的均值、累加和等统计量，最后该类别中的各样本对这些统计量进行相减或相除运算，来得到样本相对于该统计量的偏离值。偏离值特征可以很好的反映单个样本相对于该类别的所有样本来说，是偏高还是偏低，也是一种针对不同类别进行的无量纲化处理。

在本项目中，我们根据企业的行业、区域、企业类型、控制人类型、企业规模这5个离散特征进行分类，之后计算所有数值型特征相对于类别总体均值、累加和、最小值和最大值的偏离量，共能得到420个生成特征。

5.3.4 特征选择

特征选择算法可以被视为搜索技术和评价指标的结合。前者提供候选的新特征子集，后者为不同的特征子集打分。特征选择能够有效地从原始特征中找出对模型预测最有效的特征，通过使用这些特征来达到降低数据维度，简化模型的目的。与此同时，对数据关键特征的选取也能起到提高模型通用型、降低过拟合的效果。特征选择的常用方法可以分为过滤类方法(Filter)、包装类方法(Wrapper)和嵌入类方法(Embedded)三类。在本项目的特征选择实验中，我们根据数据特点和自身的实践经验，在这三种方法的基础上进行了一些综合与改进，我们首先在数据的基本特征集上使用过滤类和包装类方法，之后再结合生成特征集使用嵌入类方法来得到特征选择结果。

5.3.4.1 过滤类方法(Filter)

过滤类方法选用代理指标，并据此对数据的各个特征进行评分，设定阀值以选择特征。常用的代理指标有互信息、皮尔逊系数等。过滤类方法的计算时间相对较少，但由于选择结果通过代理指标直接产生，即使是针对在不同假设下的预测模型，过滤类方法也只能产生相同的选择结果，这会导致其预测性能相比另外两种方法较为低下。因此，在本项目中，我们只参考过滤类方法所提供的特征排名，将其作为样本特征和标签之间关系的一种度量，而不直接使用过滤类方法来选择特征子集。

* 方差选择

方差是衡量一个特征是否发散的重要指标，如果一个特征的方差接近于0，也就说明样本在这个特征上几乎没有差异，那么这个特征对于样本区分的作用就很小了。我们对数据集中的所有连续型特征进行了方差计算，各特征的方差值普遍较大，并未发现特征不发散的情况。

* 相关系数选择

相关系数是用来衡量特征与标签之间相关性的指标，相关系数越大，说明该特征对于标签的影响越高，对模型的预测也越重要。在统计学上最常用的相关系数是皮尔逊系数，其被定义为两个变量之间的协方差与标准差的商，计算公式如下：

皮尔逊系数的取值范围为-1到1，用来描述两个变量的线性相关性。皮尔逊系数为1表明两变量之间存在线性关系，且随着增加而增加；皮尔逊系数为-1表明两变量之间存在线性关系，且随着增加而减少。

我们计算了数据集中所有特征的皮尔逊系数，并根据其绝对值大小进行了排序，筛选出了与标签相关性最高的10个特征：

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 特征 | 净利润平均值 | 纳税总额平均值 | 2016-2017净利润差额 | 2015-2016纳税总额差额 | 2016-2017纳税总额差额 | 2015-2016净利润差额 | 著作权 | 2015-2016项目融资和政策融资额度差额 | 2015-2016项目融资和政策融资成本差额 |
| 皮尔逊系数 | -0.490 | -0.408 | -0.032 | -0.030 | -0.030 | -0.029 | -0.019 | 0.017 | 0.017 |

从表中可以看出，净利润和纳税总额这两个特征与标签之间的相关性极高，其皮尔逊系数是其他特征的10倍以上。负相关性表明净利润和纳税总额越高，标签值越小，越有可能为负类(非僵尸企业)，这与我们的主观认知相吻合。

5.3.4.2 包装类方法(Wrapper)

包装类方法使用预测模型给特征子集打分。每个新子集都被用来训练一个模型，然后用验证数据集来测试。通过计算模型在验证数据集上的错误率给特征子集评分。由于对所有的特征子集都进行了遍历，包装类方法往往能为特定模型找到性能最好的特征集。但是，对每一个特征子集，包装类方法都要训练一个新模型，这样做的计算开销非常大。因此，在本项目中我们根据之前过滤类方法得到的特征重要性排序来设置特征子集，先将重要性最高的特征加入到子集中，之后采取贪心策略逐步从备选集中选择特征。

在实验中，我们首先将通过过滤类方法得到的相关性最高的两个特征：净利润平均值和纳税总额平均值加入特征子集中，然后使用默认参数训练XGBoost模型，在验证集上检验得到模型准确率为99.62%，召回率为99.96%，F-score为99.35%。实验结果表明仅仅通过净利润和纳税这两个特征组成的特征子集已经能够几乎完美的实现模型预测。

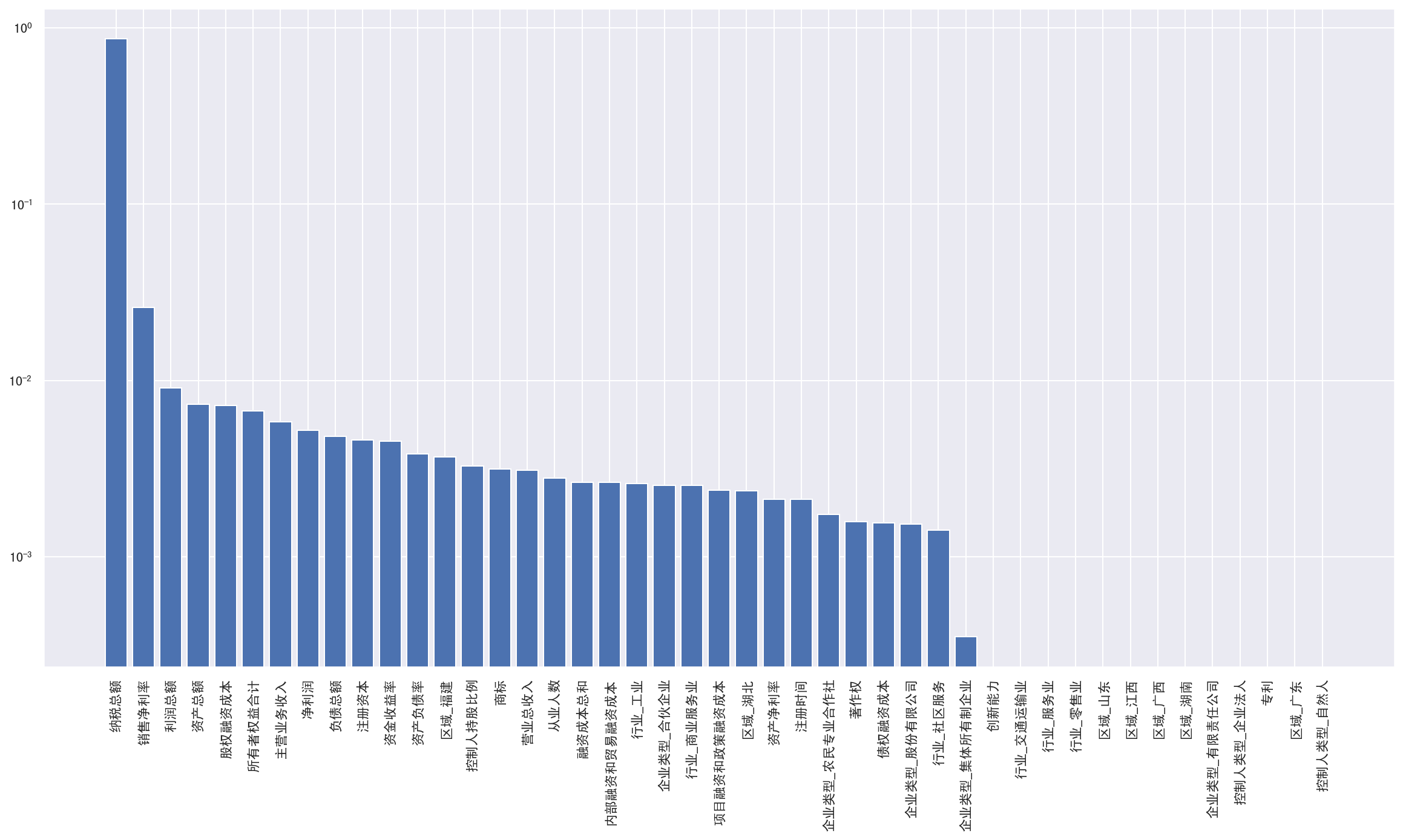
之后我们根据之前得到的特征相关性排序，逐步向子集中添加新的特征，直到排在前10的特征全部加入到子集中。我们观察到，在此过程中随着特征不断加入，模型在验证集上的效果并没有明显的提升，这表明除了净利润和纳税以外的其他特征对模型预测的影响很小。之后，我们使用所有的特征训练模型，在验证集上检验得到模型准确率为99.63%，召回率为100.00%，F-score为99.37%。使用所有特征的模型与只使用净利润和纳税这两个特征的模型在结果上值提高了万分之二，这一结果再次佐证了之前得出的结论。

5.3.4.3 嵌入类方法(Embedded)

嵌入类方法先选择模型进行训练，然后通过模型得到各个特征的权值系数或重要性分数，据此排序从大到小选择特征。这与过滤类方法有些类似，但嵌入类方法是通过训练模型来选定特征而非代理指标，且能够根据特定的模型调整特征选择。嵌入类方法需要进行模型训练，但不需要遍历特征子集，因此计算复杂度往往在过滤类方法和包装类方法之间。在本项目中，我们在之前两种方法的基础之上使用嵌入类方法，实验中均使用默认参数的XGBoost作为训练模型。

* 基本特征集的重要性分析

针对基本特征进行的选择已经使用上面两种方法进行了实验，在这里我们将带有经济学意义的4个交叉特征(销售净利率、资产净利率、资产负债率和资金收益率)加入到特征集中训练XGBoost模型，得到的特征重要性分数如下图所示(横轴为特征名，纵轴为特征重要性分数的对数)：



图X 基本特征集的重要性图

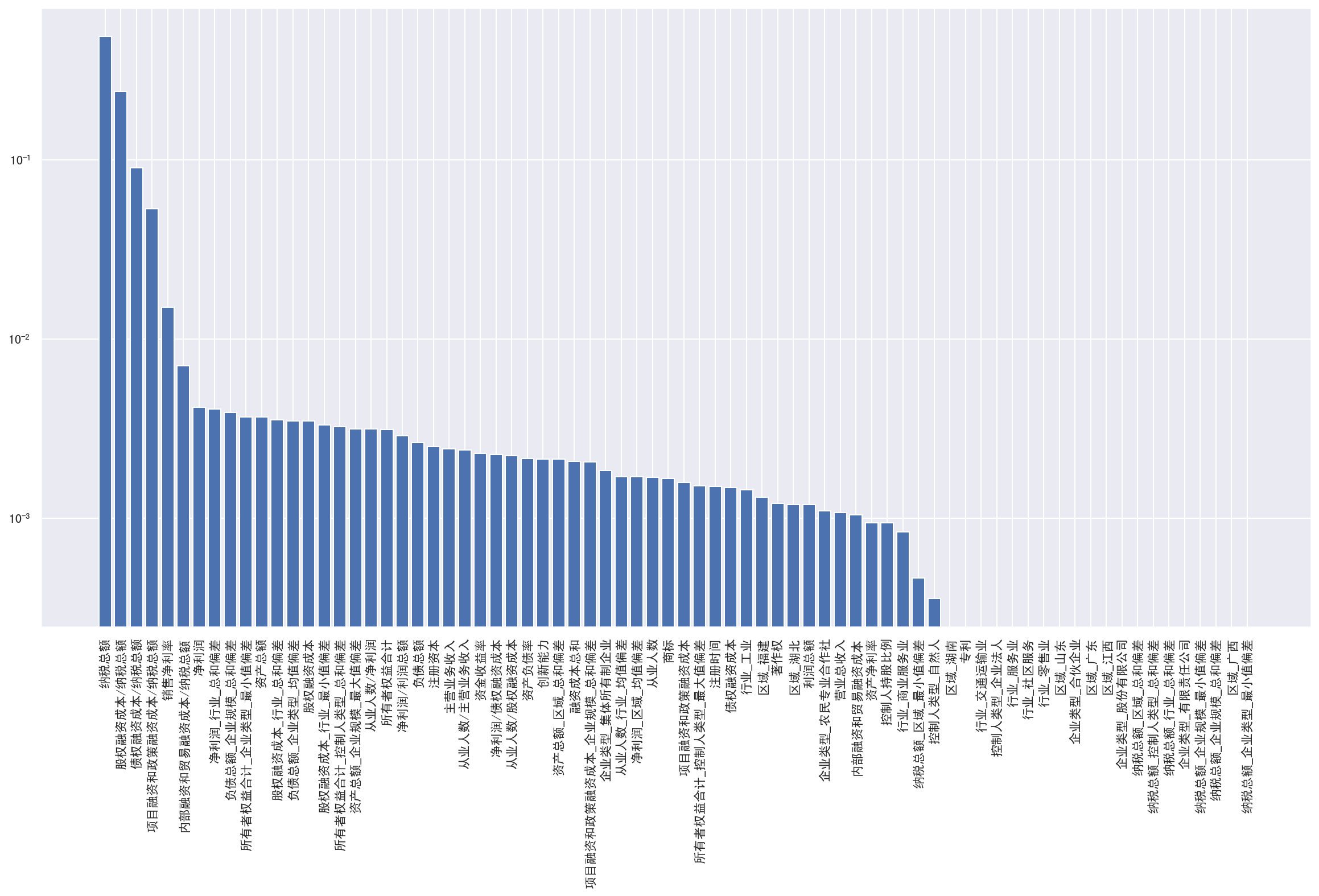
从图中可以看出，纳税总额这一特征的重要性分数在取对数后仍然远高于其他特征。而重要性分数相对较高的多为从年表中的直接获得或交叉生成的数值型特征，大部分离散型特征的重要性分数几乎为0。

* 生成特征集的重要性分析

在将基本特征和生成特征合并为完整特征集进行训练之前，我们使用一种两步走的方法来降低训练开销。针对生成特征，我们将其分为交叉特征(210个)和偏离值特征(420个)两个特征子集，在这两个子集上分别采用用嵌入类方法进行特征选择，挑选出最重要的若干特征与基本特征合并为最终的完整特征集。在实验中，我们从交叉特征集中挑选了重要性分数最高的10个特征，从偏离值特征集中挑选了重要性分数最高的20个特征。

* 完整特征集的重要性分析

通过上面的步骤得到完整特征集后，我们同样使用默认参数训练XGBoost模型，得到的特征重要性分数如下图所示：



图X 完整特征集的重要性图

从图中可以看到，纳税总额这一特征对模型预测结果的影响仍热最大，同时三个与纳税总额有关的生成特征也得到了较高的重要性分数，这一结果再次凸显了纳税总额特征对于僵尸企业分类模型的巨大作用。而在重要性分数为0的特征中，仍然以离散型特征为主。

除此以外值得一提的是，在使用嵌入类方法进行的实验中，采用不同特征子集训练的模型在预测效果上没有显著的差异，在验证集上的F-score均能达到99%以上。在这种情况下，进行特征生成和特征选择时的关注点会更多的从模型预测效果向模型预测时间转移。结合赛题方所强调的模型运行效率来考虑，在使用少量特征就能达到非常好的预测效果时，我们倾向于在排除冗余特征的同时，尽可能地减少特征生成运算来提高模型的运行效率。因此，在本项目中最终只选用了4张数据表中所包含的有效数值型特征和之前提到的4个有明确经济学意义的生成特征。

5.4 模型选择

5.4.1 单个模型

5.4.1.1 Logistic Regression

逻辑回归是一个分类算法，他可以处理二元分类以及多元分类。虽然是有回归两个字，但是却不是回归算法。线性回归的模型是求输出特征向量和输入样本矩阵之间的线性关系系数，满足。此时是连续的，所以是回归模型。如果变成离散的，一个办法是对这个再做一次函数变换，变为.如果令的值在某个实数区间的时候是类别A，在另一个实数区间的时候就是类别B，以此类推就是一个分类模型了。如果分类的类别只有两种就是一个二元分类模型，就是本项目的僵尸企业分类，一个是僵尸企业，一个类别是非僵尸企业。

这个函数在逻辑回归中我们一般取为sigmoid函数，形式如下：

它有一个非常好的性质，即当趋于正无穷时，趋于1，而当趋于负无穷时，趋于0，这非常适合于我们的分类概率模型。另外，它还有一个很好的导数性质：

这个通过函数对求导很容易得到，后面我们会用到这个式子。如果我们令中的为：，这样就得到了二元逻辑回归模型的一般形式：

其中为样本输入，为模型输出，可以理解为某一分类的概率大小。而为分类模型的要求出的模型参数。对于模型输出，我们让它和我们的二元样本输出（假设为0和1）有这样的对应关系，如果，即, 则为1。如果，即, 则为0。是临界情况，此时，从逻辑回归模型本身无法确定分类。的值越小，而分类为0的的概率越高，反之，值越大的话分类为1的的概率越高。如果靠近临界点，则分类准确率会下降。

逻辑回归尤其是二元逻辑回归是非常常见的模型，训练速度很快，虽然使用起来没有支持向量机（SVM）那么占主流，但是解决普通的分类问题是足够了，训练速度也比起SVM要快不少。

5.4.1.2 Ramdom Forest

随机森林(Ramdom Forest, 以下简称RF)。它是Bagging算法的进化版，也就是说思想任然是bagging，但是进行了独有的改进。

首先，RF使用了CART决策树作为弱学习器。第二，在使用决策树的基础上，RF对决策树的建立做了改进，对于普通的决策树，在节点上所有的个样本特征中选择一个最优的特征来做决策树的左右子树划分，但是RF通过随机选择节点上的一部分样本特征，这个数字小于，假设为，然后在这些随机选择的个样本特征中，选择一个最优的特征来做决策树的左右子树划分。这样进一步增强了模型的泛化能力。

如果，则此时RF的CART决策树和普通的CART决策树没有区别。越小，则模型约健壮，当然此时对于训练集的拟合程度会变差。也就是说越小，模型的方差会减小，但是偏倚会增大。在实际案例中，一般会通过交叉验证调参获取一个合适的的值。

除了上面两点，RF和普通的bagging算法没有什么不同， 下面简单总结下RF的算法。

输入为样本集，弱分类器迭代次数。

输出为最终的强分类器

1）对于:

a)对训练集进行第次随机采样，共采集次，得到包含个样本的采样集

b)用采样集训练第个决策树模型，在训练决策树模型的节点的时候， 在节点上所有的样本特征中选择一部分样本特征， 在这些随机选择的部分样本特征中选择一个最优的特征来做决策树的左右子树划分

2) 如果是分类算法预测，则个弱学习器投出最多票数的类别或者类别之一为最终类别。如果是回归算法，个弱学习器得到的回归结果进行算术平均得到的值为最终的模型输出。

RF的主要优点有：

1） 训练可以高度并行化，对于大数据时代的大样本训练速度有优势。

2） 由于可以随机选择决策树节点划分特征，这样在样本特征维度很高的时候，仍然能高效的训练模型。

3） 在训练后，可以给出各个特征对于输出的重要性

4） 由于采用了随机采样，训练出的模型的方差小，泛化能力强。

5） 相对于Boosting系列的Adaboost和GBDT， RF实现比较简单。

6） 对部分特征缺失不敏感。

RF的主要缺点有：

1）在某些噪音比较大的样本集上，RF模型容易陷入过拟合。

2) 取值划分比较多的特征容易对RF的决策产生更大的影响，从而影响拟合的模型的效果。

5.4.1.3 GBDT

梯度提升树(Gradient Boosting Decison Tree, 以下简称GBDT)。GBDT也是集成学习Boosting家族的成员，但是和传统的Adaboost有很大的不同。Adaboost是利用前一轮迭代弱学习器的误差率来更新训练集的权重，这样一轮轮的迭代下去。GBDT也是迭代，使用了前向分布算法，但是弱学习器限定了只能使用CART回归树模型，同时迭代思路和Ababoost也有所不同。

在GBDT的迭代中，假设我们前一轮迭代得到的强学习器是, 损失函数是, 我们本轮迭代的目标是找到一个CART回归树模型的弱学习器，让本轮的损失函数最小。也就是说，本轮迭代找到决策树，要让样本的损失尽量变得更小。

GBDT算法的具体流程如下：

输入为训练集样本，最大迭代次数，损失函数。

输出是强学习器。

1. 初始化弱学习器
2. 对迭代轮数有：
3. 对样本，计算负梯度
4. 利用，拟合一棵CART回归树，得到第棵回归树，其对应的叶子节点区域为，其中为回归树的叶子节点个数。
5. 对叶子区域，计算最佳拟合值
6. 更新强学习器
7. 得到强学习器的表达式

GBDT主要的优点有：

1) 可以灵活处理各种类型的数据，包括连续值和离散值。

2) 在相对少的调参时间情况下，预测的准确率也可以比较高。

3）使用一些健壮的损失函数，对异常值的鲁棒性非常强。比如 Huber损失函数和Quantile损失函数。

GBDT的主要缺点在于，由于弱学习器之间存在依赖关系，难以并行训练数据。不过可以通过自采样的SGBT来达到部分并行。

5.4.1.4 XGBoost

作为GBDT的高效实现，XGBoost是一个上限特别高的算法，因此在算法竞赛中比较受欢迎。简单来说，对比原算法GBDT，XGBoost主要从下面三个方面做了优化：

一是算法本身的优化：在算法的弱学习器模型选择上，对比GBDT只支持决策树，还可以直接很多其他的弱学习器。在算法的损失函数上，除了本身的损失，还加上了正则化部分。在算法的优化方式上，GBDT的损失函数只对误差部分做负梯度（一阶泰勒）展开，而XGBoost损失函数对误差部分做二阶泰勒展开，更加准确。

二是算法运行效率的优化：对每个弱学习器，比如决策树建立的过程做并行选择，找到合适的子树分裂特征和特征值。在并行选择之前，先对所有的特征的值进行排序分组，方便前面说的并行选择。对分组的特征，选择合适的分组大小，使用CPU缓存进行读取加速。将各个分组保存到多个硬盘以提高IO速度。

三是算法健壮性的优化：对于缺失值的特征，通过枚举所有缺失值在当前节点是进入左子树还是右子树来决定缺失值的处理方式。算法本身加入了L1和L2正则化项，可以防止过拟合，泛化能力更强。

XGBoost算法主要流程：

输入是训练集样本， 最大迭代次数, 损失函数， 正则化系数。

输出是强学习器。

对迭代轮数有：

1) 计算第个样本在当前轮损失函数基于的一阶导数，二阶导数,计算所有样本的一阶导数和,二阶导数和。

2) 基于当前节点尝试分裂决策树，默认分数score为0，和为当前需要分裂的节点的一阶、二阶导数只和。

对特征序号：

1. 令；
2. 将样本按特征从小到大排序，依次取出第个样本，计算将这个样本放入左子树或右子树后的一阶、二阶导数和：
3. 尝试更新最大的分数：

3) 基于最大score对应的划分特征和特征值分裂子树。

4) 如果最大score为0，则当前决策树建立完毕，计算所有叶子区域的, 得到弱学习器，更新强学习器,进入下一轮弱学习器迭代。如果最大score不是0，则转到第2)步继续尝试分裂决策树。

5.4.2 单模调参

在对之前提到的多个模型进行实验和比较后，我们最终选择XGBoost作为本项目的分类模型，XGBoost算法预测能力强，灵活性高，可调节参数多。我们通过机器学习库sklearn中的网格搜索方法配合五折交叉验证进行参数选择，使用F-score作为评价指标。接下来我们将详细介绍对XGBoost模型的各个参数进行调整的过程。

* max\_depth + min\_child\_weight

max\_depth参数表示模型中树的最大深度，模型中的各棵树在达到此深度后会停止继续分裂。随着树深度的增加，模型能够学习到更具体、更局部的样本，预测能力会变得更强，但也越容易导致过拟合。因此，这个参数的设置不宜过大，通常在3-10之间。

min\_child\_weight参数表示各个叶子节点下所有样本的最小权重和，当树在一次分裂中，某个叶子节点下所有样本的权重和小于这个值，树就会停止分裂。从直观上看，这个参数表示了树的每个叶子节点至少要包含的样本的带权数目，当树的叶子节点上包含的样本过少时，更容易出现过拟合。在具体实现上，某个样本的权重即为损失函数对该样本在上一轮迭代中的预测值的二阶导数。这样做的合理性在于，我们可以将模型在每一轮优化时的损失函数重写为如下形式：

这时目标函数可被视为模型预测值与标签之间的均方误差损失，权重为。

这两个参数对模型的预测结果有很大的影响，所以我们首先对它们进行调整。我们首先设置较大跨度的参数集：

|  |  |
| --- | --- |
| max\_depth | 2, 4, 6, 8, 10 |
| min\_child\_weight | 1, 3, 5, 7, 9 |

参数集包含25种排列组合，在训练集上使用五折交叉验证共需125次训练。训练完成后F-score排在前5的参数组合如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 训练时间均值 | max\_depth | min\_child\_weight | 评价指标 |
| 5.066 | 2 | 3 | 99.295(±1.360) |
| 4.964 | 2 | 9 | 99.295(±1.360) |
| 4.999 | 2 | 5 | 99.295(±1.360) |
| 4.945 | 2 | 7 | 99.295(±1.360) |
| 5.302 | 2 | 1 | 99.295(±1.360) |

从表中可以看出，当树的最大深度为2时，模型的预测效果最佳，而叶子节点最小权重和这一参数对模型的结果没有太大影响。接下来我们参考刚才得到的最优参数，缩小选取范围和跨度，设置第二组参数集：

|  |  |
| --- | --- |
| max\_depth | 1, 2, 3 |
| min\_child\_weight | 1, 3, 5 |

第二组参数集包含9种排列组合，在训练集上使用五折交叉验证共需45次训练。训练完成后F-score排在前5的参数组合如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 训练时间均值 | max\_depth | min\_child\_weight | 评价指标 |
| 3.478 | 1 | 1 | 99.315(±1.370) |
| 3.624 | 1 | 3 | 99.315(±1.370) |
| 3.542 | 1 | 5 | 99.305(±1.365) |
| 5.116 | 2 | 3 | 99.295(±1.360) |
| 5.176 | 2 | 5 | 99.295(±1.361) |

从表中可以看出，深度为1的树仅可达到最优的预测效果。正如之前在特征工程中所发现的，除了纳税和净利润等少数几个关键特征，绝大多数特征与标签的关联性不高，这是树结构如此简单的直接原因。而对于一棵深度为1的二叉树，其只有两个叶子节点，这种情况下叶子节点的最小权重和参数就意义不大了，因为当将数以万计的样本分裂到两个叶子节点上时，很少会出现样本全部集中在一个节点而另一节点权重和极低的状况。

* gamma

参数主要用于模型正则化，这个参数在损失函数的惩罚项中出现，也被用作判断树是否分裂的阀值。在XGBoost算法中，惩罚项可被展开为

其中为树的叶子节点数目，为第个叶子节点的权重。展开式的第二项为常用的L2正则项，而第一项用于限制叶子节点的数目。表示每增加一个叶子节点所需的代价，通过增加的值，可以有效抑制节点分裂，从而起到防止过拟合的效果。

同时，参数还在叶子节点的分裂损失函数中作为阀值出现。在默认情况下，当一个节点即将分裂时，需要分别计算分裂后左右节点的损失函数值，若其和大于未分裂时的损失函数值，则进行分裂，否则不分裂。而当使用参数时，需要分裂后左右节点的损失函数和与未分裂时损失函数的差值大于才进行分裂。这样做同样起到抑制节点分裂，防止过拟合的效果。

我们首先设置第一组参数集：

|  |  |
| --- | --- |
| gamma | 0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0 |

参数集包含11种候选值，在训练集上使用五折交叉验证共需55次训练。通过训练后我们发现，对于gamma参数的11个取值，我们得到了完全一样的训练结果。为了判定所选取的gamma值是否过小，我们设置了第二组跨度较大的参数集：

|  |  |
| --- | --- |
| gamma | 0, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000 |

第二组参数集包含11种候选值，在训练集上使用五折交叉验证共需55次训练。而这组参数的11个取值在训练后同样得到了和之前一样的训练结果。这表明该参数对本项目的分类模型影响不大。这个结果一方面表明在XGBoost模型中树的叶子节点数目很少，另一方面也表明树中节点的每一次分裂所带来的增益极高。

* subsample + colsample\_bytree

这里的两个参数是在各个树模型中都被广泛使用的参数。对于一张数据表，通常每行代表一个样本，而每列代表一个特征。这两个参数就是在训练时对样本和特征进行采样时使用的。

subsample为行采样参数，即对于模型的每棵树，在训练时对样本进行随机采样的比例。如设置subsamle为0.5时，训练每棵树时都只选用50%的样本，这样做可以避免过拟合。但当这个参数过小时，训练样本太少，也会导致模型欠拟合，因此，这个参数通常设置在0.5-1.0之间。

colsample\_bytree为列采样参数，即对于模型的每棵树，在训练时对特征进行随机采样的比例。如设置colsample\_bytree为0.5时，训练每棵树时都只选择50%的特征来计算分裂增益，这样可以避免某个特征在模型中的权重过高，从而防止过拟合。与subsample参数一样，这个参数通常也设置在0.5-1.0之间。

这两个参数的可调整范围较小，我们设置下面的参数集：

|  |  |
| --- | --- |
| subsample | 0.6, 0.7, 0.8, 0.9, 1.0 |
| colsample\_bytree | 0.6, 0.7, 0.8, 0.9, 1.0 |

参数集包含25种排列组合，在训练集上使用五折交叉验证共需125次训练。训练完成后F-score排在前5的参数组合如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 训练时间均值 | subsample | colsample\_bytree | 评价指标 |
| 3.940 | 0.8 | 0.6 | 99.305(±1.370) |
| 4.279 | 0.8 | 0.7 | 99.305(±1.370) |
| 4.449 | 0.8 | 0.8 | 99.305(±1.370) |
| 4.678 | 0.8 | 0.9 | 99.305(±1.370) |
| 5.152 | 0.9 | 1 | 99.305(±1.370) |

从表中可以看出，行采样的比率在0.8左右时效果最优，而列采样对模型的影响不大。这也再次验证了我们通过特征工程分析得到的结果，即除去少数几个特征以外，绝大部分特征与标签的相关性小，对模型的预测几乎没有贡献。

* learning\_rate

学习率learning\_rate是在几乎所有机器学习算法中通用的参数，通过改变学习率来控制梯度下降法权重更新的幅度，可以有效控制模型训练的速度和精度。这个参数的取值范围较大，但通常选取在0.001-1.0之间，太小的学习率会导致模型收敛过慢，太大的学习率会导致模型损失函数下降太快，越过了极小值点，而出现震荡现象。

我们首先设置第一组参数集：

|  |  |
| --- | --- |
| learning\_rate | 0.001, 0.01, 0.1, 1.0 |

参数集包含4种候选值，在训练集上使用五折交叉验证共需20次训练。训练后得到的结果如下：

|  |  |  |
| --- | --- | --- |
| 训练时间均值 | learning\_rate | 评价指标 |
| 4.018 | 0.001 | 99.305(±1.365) |
| 4.057 | 0.01 | 99.295(±1.360) |
| 3.883 | 0.1 | 99.305(±1.365) |
| 3.566 | 1.0 | 99.137(±1.285) |

接下来我们尝试在0.001附近进行调整，设置了第二组参数集：

|  |  |
| --- | --- |
| learning\_rate | 0.0001, 0.0002, 0.0005, 0.0008, 0.001, 0.002, 0.005 |

第二组参数集包含7种候选值，在训练集上使用五折交叉验证共需35次训练。训练完成后我们发现学习率小于等于0.001的5次训练得到了相同的预测分数。最终我们选择学习率0.001作为最优学习率。

* n\_estimators

n\_estimators参数表示模型中使用的弱学习器的数目(本项目中为决策树)，增加树的数目能够显著的增强模型的拟合能力，但模型容易变得过于复杂，导致运行效率的较低。因此，我们先设置一组值较小的参数集：

|  |  |
| --- | --- |
| n\_estimators | 5, 10, 50, 100, 500 |

参数集包含5种候选值，在训练集上使用五折交叉验证共需25次训练，训练后得到的结果如下：

|  |  |  |
| --- | --- | --- |
| 训练时间均值 | n\_estimators | 评价指标 |
| 0.290 | 5 | 99.295(±1.360) |
| 0.470 | 10 | 99.295(±1.360) |
| 2.020 | 50 | 99.295(±1.360) |
| 4.091 | 100 | 99.305(±1.365) |
| 17.373 | 500 | 99.315(±1.370) |

从训练结果看，添加树的数目对模型有一定的影响，而训练所需的时间也明显变长了。接下来我们设置第二组参数集来观察树的数目继续增加对模型的影响：

|  |  |
| --- | --- |
| n\_estimators | 100, 300, 500, 700, 900 |

第二组参数集包含5种候选值，在训练集上使用五折交叉验证共需25次训练，训练后得到的结果如下：

|  |  |  |
| --- | --- | --- |
| 训练时间均值 | n\_estimators | 评价指标 |
| 4.288 | 100 | 99.305(±1.365) |
| 11.698 | 300 | 99.305(±1.365) |
| 20.197 | 500 | 99.315(±1.370) |
| 31.262 | 700 | 99.315(±1.370) |
| 32.955 | 900 | 99.315(±1.370) |

可以看到在树的数目达到500以后继续增加数目不再提升模型效果。接下来我们设置第三组参数集寻找在300-500之间的最优参数：

|  |  |
| --- | --- |
| n\_estimators | 300, 350, 400, 450, 500, 550, 600 |

第三组参数集包含7种候选值，在训练集上使用五折交叉验证共需35次训练。训练完成后我们发现当树的数目为350时模型评价分数达到最高值，之后继续增加数目模型分数不再提升，故我们最终选择n\_estimators为350作为最优参数。

* 综合调整

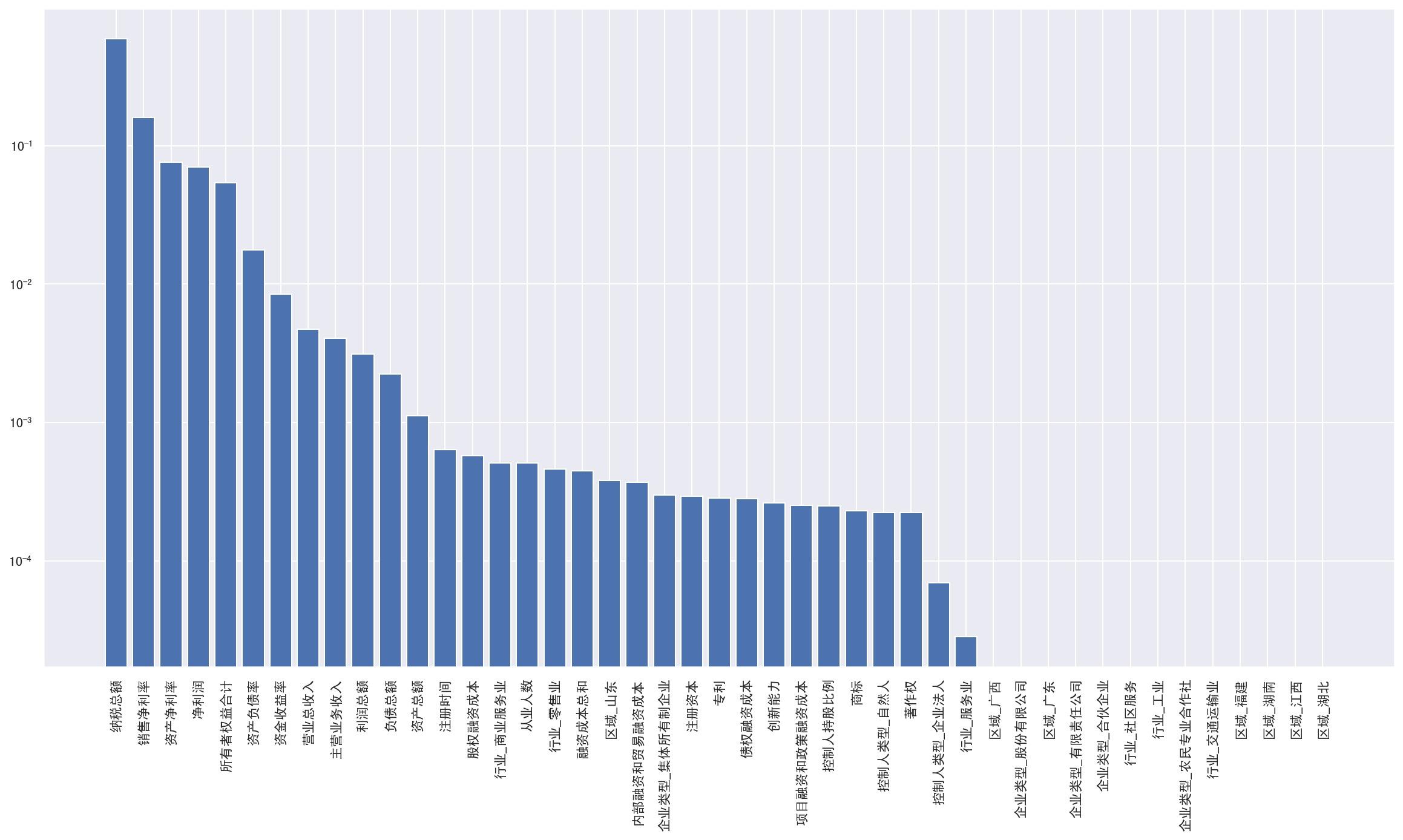
在对各个参数分别进行调整后，我们根据之前得到的最优参数值，设置了一个用于模型整体调整的参数集：

|  |  |
| --- | --- |
| max\_depth | 1, 2, 3 |
| min\_child\_weight | 1, 3, 5 |
| subsample | 0.7, 0.8 |
| colsample\_bytree | 0.5, 0.6, 0.7 |
| learning\_rate | 0.0005, 0.001 |
| n\_estimators | 100, 350, 500 |

参数集包含324种排列组合，在训练集上使用五折交叉验证共需1620次训练。训练完成后的最优参数如下：

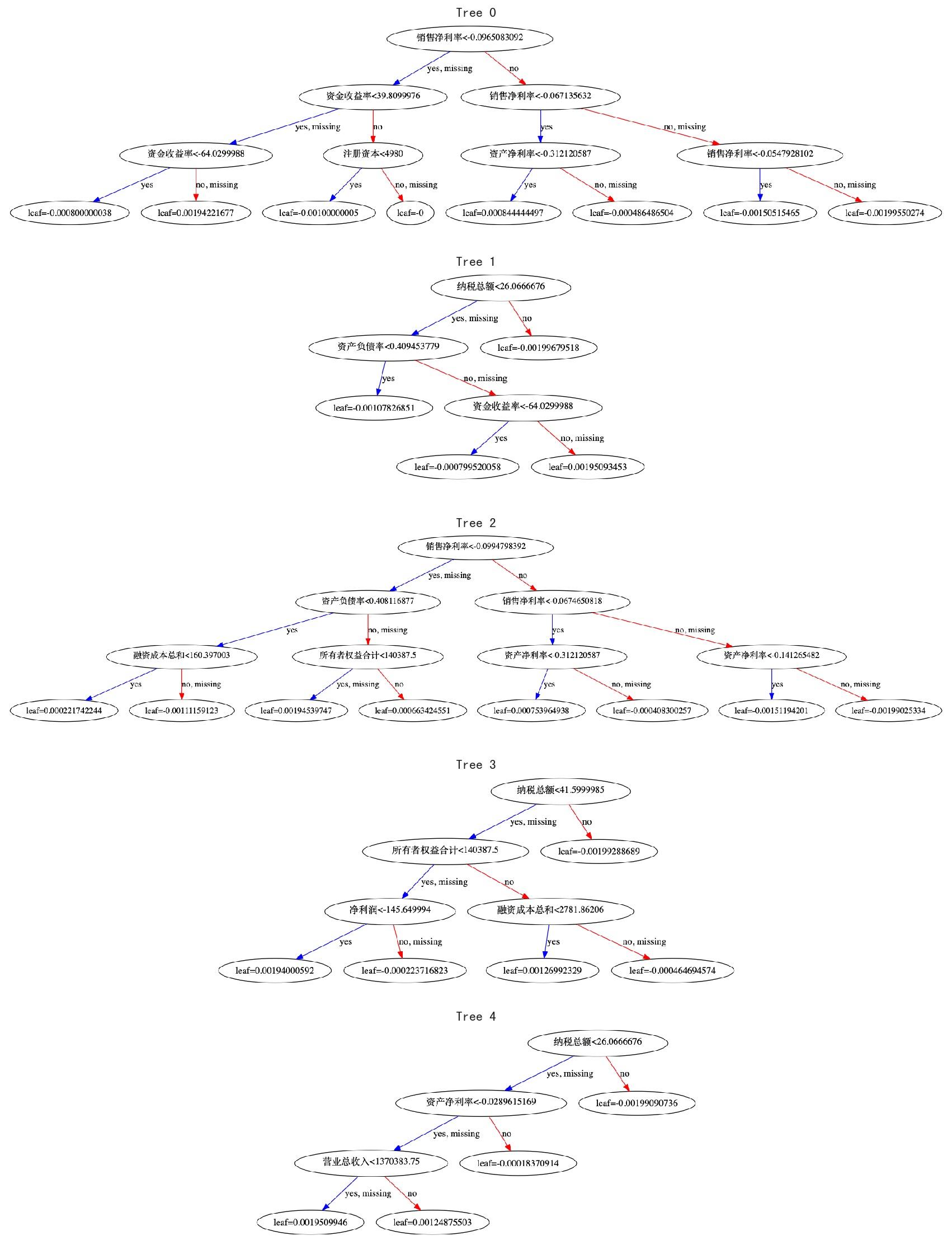
|  |  |
| --- | --- |
| 训练时间均值 | 25.634 |
| max\_depth | 3 |
| min\_child\_weight | 1 |
| subsample | 0.8 |
| colsample\_bytree | 0.5 |
| learning\_rate | 0.001 |
| n\_estimators | 500 |
| 评价指标 | 99.342(±1.292) |

使用通过参数调整得到的最优模型在验证集上评估，得到准确率为99.93%，召回率为99.87%，F-score为99.88%，AUC为99.91。查看该模型的各特征重要性分数，绘制图表如下：



图X 最优模型的特征重要性图

此外，我们还对最优模型的前5棵决策树进行了可视化，如下图所示：



图X 最优模型的前5棵决策树图

从上面的两张图可以看出，纳税总额这一特征对于模型分类效果的影响是最大的，前5棵决策树中有3棵将其作为根节点进行分裂。

六、系统设计

6.1 Web前端设计

Web页面主要用于项目成果展示，系统除了实现僵尸企业分类的基本功能，还支持通过ID进行企业画像标签查询、同类型企业数据对比分析等功能。Web页面主要由文件上传和画像分析两个区域构成，用户在文件上传区域选择符合标准的4份csv文件(基本信息表、知识产权表、融资表和年表)，服务器端将运行预训练好的模型预测僵尸企业分类结果，用户可通过页面的下载窗口将预测结果保存到本地。在生成分类结果后，服务器端还将生成各企业的画像标签信息，用户可以在画像分析区域的搜索窗口输入企业ID进行查询。在进行一次查询后，Web页面的画像分析区域将显示企业基本信息、企业画像标签和企业数据与总体均值比较图，用户可以直观的查看企业的经营状况，并与同类型企业的数据进行对比。画像分析区域效果图如下：



图X Web前端画像分析区界面截图

由于本项目中涉及的Web前端为简单的单页面类型，我们在编写代码时尽可能选用了轻量级的开发工具，网页也针对用户的主要需求进行功能设置，网页整体遵循简明易用、直观高效的设计原则。在Web开发中，我们以前端主流使用的HTML+CSS+JavaScript为基础，使用Vue框架进行方便的数据绑定与更新，并实现视图层与模型的交互。此外，我们使用Element框架来美化Web界面的UI效果，通过Chart.js框架来获得简洁优美、交互性强的动态图表。

6.2 Web后端设计

Web后端负责接受来自前端的请求并进行处理，在本项目中的请求可以分为文件传输、模型预测和画像分析三类，我们在后端设置了如下表所示的多个API接口以处理不同种类的请求。

|  |  |  |  |
| --- | --- | --- | --- |
| API | 请求方法 | 请求参数 | 备注 |
| upload | POST | key, file | 上传csv文件 |
| remove | POST | key, fileName | 从服务器删除csv文件 |
| predict | POST | key | 进行僵尸企业分类预测 |
| search | GET | key, id | 查询指定ID的企业画像 |
| chart | GET | key, id, class | 按照类别更新指定企业对比图 |

本项目中Web后端主要由Python语言实现，分为服务器层、中间件层和应用程序层三个部分，主要使用的技术如下：

* 服务器层——Nginx

在传统的客户端-服务端架构中，服务端主要负责接受从客户端发送的HTTP请求并进行处理，然后向客户端返回响应。本项目中选用了支持负载均衡的高性能服务器Nginx，与另一常用的服务器Apache相比，Nginx部署简单，内存消耗少，运行成本低，非常适合用于本项目中的单页面轻量级Web开发。

* 中间件层——WSGI

在项目的后端开发中，服务器层直接与用户端进行交互和数据传递，在服务器接收到用户请求后，需要交给应用程序进行处理，应用程序处理完后将结果返回给服务器，服务器再向各户端返回响应。而中间件层就是用于服务器和应用程序之间交互的一种规范，在本项目中我们使用的是Python语言专用的WSGI规范。WSGI全称是Web Server Gateway Interface，也就是Web 服务器网关接口，它是由Python语言定义出来的服务器和应用程序之间的简单而通用的接口，基于现存的 CGI 标准设计。

* 应用程序层——Flask

应用程序层负责具体实现Web应用的功能，我们将僵尸企业分类、企业画像分析等功能封装为多个应用程序，供服务器层调用。本项目中选用了轻量级框架Flask，Flask是最精致、功能最丰富的微框架之一，不仅部署配置简单高效，且通过模块化的设计得到了更高的灵活性和可扩展性，是开发小型Web应用的不二选择。