教育部高等学校大学计算机课程教学指导委员会

中国大学生计算机设计大赛



软件开发类作品文档简要要求

作品编号：　　　　　　 61757

作品名称：　　基于人脸识别的智慧医疗预约挂号平台

作　　者：　　　 文杰、周文志、杨昊凡

版本编号：　　　　　　　V1.0

填写日期：　　　　　 2019/04/30

填写说明：

1. 本文档适用于**所有**涉及软件开发的作品，包括：软件应用与开发、大数据、人工智能、物联网应用；
2. 正文一律用五号宋体，一级标题为二号黑体，其他级别标题如有需要，可根据需要设置；
3. 本文档为简要文档，不宜长篇大论简明扼要为上；
4. 提交文档时，以PDF格式提交本文档；
5. 本文档内容是正式参赛内容组成部分，务必真实填写。如不属实，将导致奖项等级降低甚至终止本作品参加比赛。

# 1.需求分析

【填写说明：本部分内容建议不超过1000字，以300字以内为宜，简要说明为什么开发本作品，是否存在竞品，对标什么作品以及面向的用户、主要功能、主要性能等。建议有竞品分析表格，从多个维度分析本作品与竞品作品比较】

## 项目背景

随着互联网大潮的推进,互联网技术逐渐进入医疗领域,各大医院也在积极进行互联网相关的改造,切实解决患者的就医问题。为帮助患者进行就诊挂号，节省挂号时间，方便患者快速就诊，本项目从生物特征识别技术中最实用、应用最广泛的人脸识别技术入手,旨在开发一款基于人脸识别的智慧医疗预约挂号平台。

## 用户分析

* 预挂号病患；
* 科室医生；
* 医院后台管理人员；

## 主要功能

1、人脸识别信息注册。挂号时运用“人脸识别注册”通过身份证+人脸识别，这种验证系统精准、科学地防止“号贩子”恶意注册并占用挂号资源，真实有效的方便实际病患挂号需求；

2、精准信息检索。用户可通过点击相应科室、疾病或模糊搜索，实现预约挂号；

3、地图路径规划。页面显示医院精准定位，用户可根据当前定位选择合适的驾车、公交、步行路径规划方式。

## 竞品分析

**表1. 竞品分析表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **项目名称** | **功能特色** | **市场布局** | **数据情况** | **操作情况** | **界面情况** |
| 浙江在线预约诊疗服务平台 | 基本功能齐全 | 浙江省范围医院 | 数据齐全 | 查询复杂，页面刷新良好 | 页面信息繁杂 |
| 乌镇互联网医院 | 地图显示 | 全国部分医院 | 实际数据缺乏，部分查询无法显示 | 挂号流程较为繁琐 | 页面信息良好 |
| 绍兴市预约诊疗平台 | 基本功能齐全 | 绍兴市区医院 | 较为齐全 | 页面相应时间较长 | 页面体验感不佳 |

通过与表1所示三款同类型竞品分析得出，本作品具有以下优势：

1. 基于人脸识别，采用最新的生物识别技术；
2. 医院定位显示与路径规划；
3. 页面简洁，操作方便；
4. 应用对象范围偏小，为用户提供精准服务。

# 2.概要设计

【填写说明：将需求分析结果分解成功能模块以及模块的层次结构、调用关系、模块间接口以及人机界面等，建议用图体现内容，不宜全文字描述。建议图文总体不超过A4纸两页，以1页为宜。】

## 2.1模块设计及接口

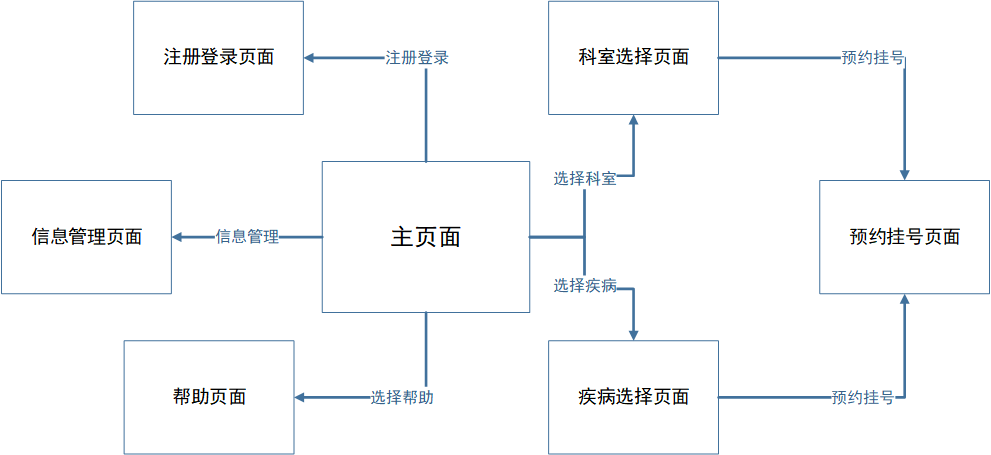
根据需求分析结果将项目内容分为7大页面模块，内容及其相应接口划分如表2所示：

**表2. 模块设计表**

|  |  |  |
| --- | --- | --- |
| **模块名称** | **模块功能简述** | **模块接口** |
| 主页面 | 显示科室，疾病，以及主要信息，提供其他页面入口链接。 | 无 |
| 注册登录页面 | 用户进行注册，登录 | 注册、登录接口 |
| 信息管理页面 | 用户，医生，管理员的信息管理 | 用户、医生、管理员信息管理接口 |
| 科室选择页面 | 用户选择挂号所需的科室 | 科室选择接口 |
| 疾病选择页面 | 用户选择自身的疾病，引导至相应科室进行挂号 | 疾病选择接口 |
| 预约挂号页面 | 用户进行预约挂号 | 预约挂号、医生列表、排班信息接口 |
| 帮助页面 | 显示常见问题，注册登录指南，预约指南 | 常见问题、注册指南、预约指南接口 |

## 2.2模块调用关系

根据所设计的功能模块，对各模块部分进行合理的逻辑流程划分，如图1所示：



**图1.模块逻辑流程图**

# 3.详细设计

【填写说明：包括但不限于：界面设计、数据库设计(如果有)、关键算法。界面设计建议用作品实际界面，建议包括典型使用流程；数据库设计建议用表格、ER图或UML方式，说明文字简明扼要，违背范式的设计建议说明理由；关键算法可以替换为关键技术、技术创新等。本部分不宜大篇幅铺陈，建议突出重点痛点难点特点。】

## 3.1界面设计

## 3.2数据库设计

### 3.2.1数据字典

本项目数据分为7大部分，其中具体数据项的数据字典如下表所示：

**表3.用户表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **表名** | hos\_user | | | |
| **列名** | **数据类型** | **空/非空** | **约束条件** | **中文名称** |
| usid | int | 非空 | 主键 | 用户编号 |
| uname | varchar（20） | 非空 |  | 姓名 |
| sex | varchar（20） | 非空 |  | 性别 |
| cardtype | varchar（20） | 非空 |  | 证件类型 |
| cardid | varchar（100） | 非空 |  | 证件号码 |
| pwd | varchar（100） | 非空 |  | 密码 |
| address | varchar（20） | 非空 |  | 地址 |
| tel | varchar（100） | 非空 |  | 手机号 |
| email | varchar（100） | 非空 | 唯一 | 邮箱 |
| photo | varchar（100） | 非空 |  | 照片 |

**表4.科室表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **表名** | hos\_ks | | | |
| **列名** | **数据类型** | **空/非空** | **约束条件** | **中文名称** |
| kid | int | 非空 | 主键 | 科室编号 |
| kname | varchar（20） | 非空 |  | 科室名称 |
| ktype | varchar（20） | 非空 |  | 所属大类 |
| hot | Int | 非空 |  | 热门度 |

**表5.疾病表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **表名** | hos\_jb | | | |
| **列名** | **数据类型** | **空/非空** | **约束条件** | **中文名称** |
| jid | int | 非空 | 主键 | 疾病编号 |
| kid | int | 非空 | 外键 | 科室编号 |
| jname | varchar（20） | 非空 |  | 疾病名称 |
| stype | varchar（20） | 非空 |  | 小部位 |
| btype | varchar（20） | 非空 |  | 大部位 |
| hot | Int | 非空 |  | 是否常见 |

**表6.医生表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **表名** | hos\_doctor | | | |
| **列名** | **数据类型** | **空/非空** | **约束条件** | **中文名称** |
| did | int | 非空 | 主键 | 医生编号 |
| kid | int | 非空 | 外键 | 科室编号 |
| msg | varchar（200） | 空 |  | 简介 |
| pic | varchar（100） | 非空 |  | 照片 |

**表7.日程表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **表名** | hos\_schedule | | | |
| **列名** | **数据类型** | **空/非空** | **约束条件** | **中文名称** |
| sid | int | 非空 | 主键 | 日程编号 |
| did | int | 非空 | 外键 | 医生编号 |
| sdate | date | 非空 |  | 日期 |
| timetype | varchar（20） | 非空 |  | 所属时间段 |
| rescount | Int | 非空 |  | 可预约总数 |

**表8.黑名单表**

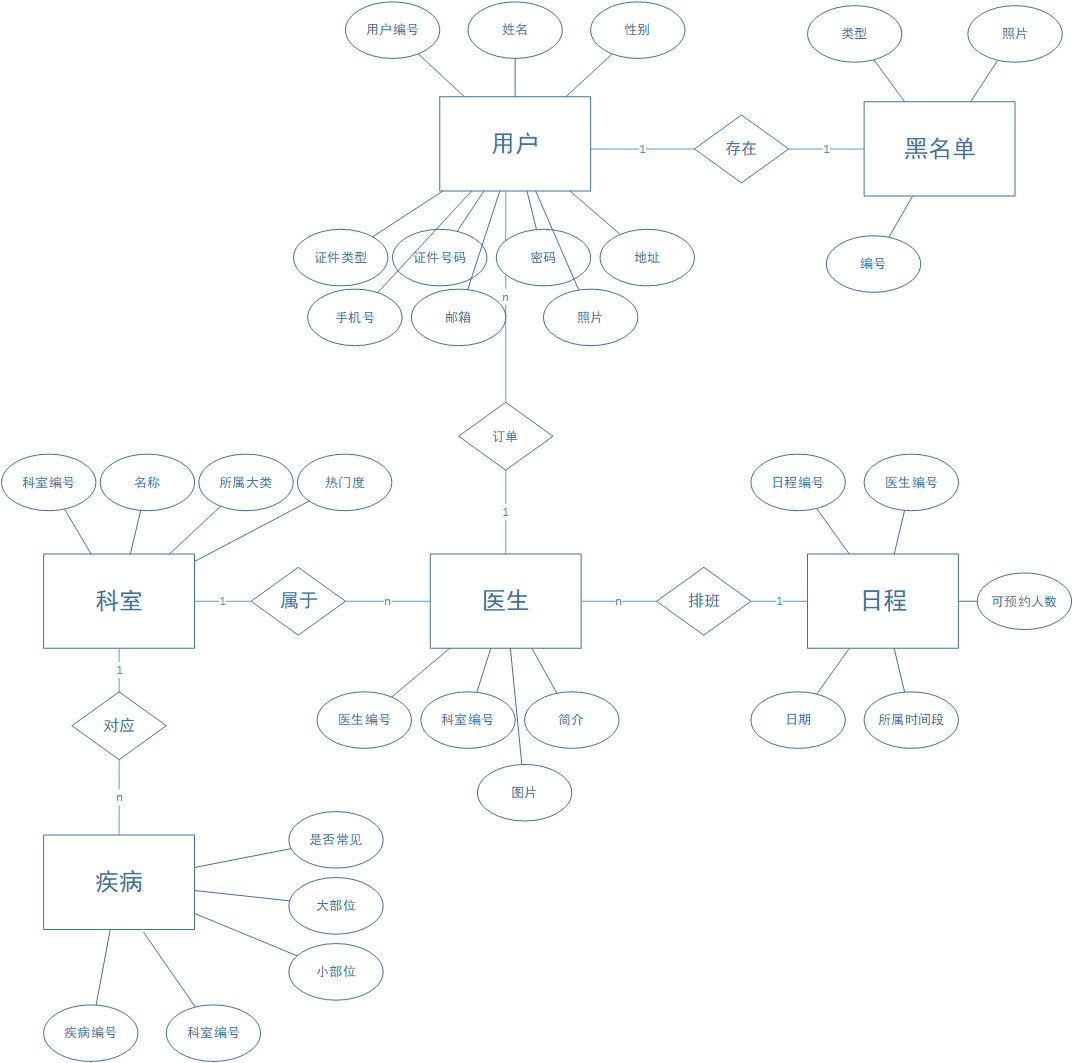
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **表名** | hos\_blacklist | | | |
| **列名** | **数据类型** | **空/非空** | **约束条件** | **中文名称** |
| bid | int | 非空 | 主键 | 黑名单编号 |
| photo | varchar（100） | 非空 |  | 照片 |
| btype | varchar（20） | 非空 |  | 类型 |

**表9.订单表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **表名** | hos\_order | | | |
| **列名** | **数据类型** | **空/非空** | **约束条件** | **中文名称** |
| oid | int | 非空 | 主键 | 订单编号 |
| uid | int | 非空 | 外键 | 用户编号 |
| sid | int | 非空 | 外键 | 日程编号 |

### 3.2.2概念模型设计

对网站使用用户所需的现实使用情况，通过对其中所处分类、聚类和概括，建立抽象的概念数据模型。网站全局ER图如图2所示：



**图2.全局ER图**

## 3.3关键技术

### 3.3.1 人脸信息比对

本项目使用dlib中提供的人脸检测方法（使用HOG特征或卷积神经网方法），并使用提供的深度残差网络（ResNet）实现实时人脸识别，不过本项目的目的不是构建深度残差网络，而是利用已经训练好的模型进行实时人脸识别，实时性要求一秒钟达到10帧以上的速率，并且保证不错的精度。

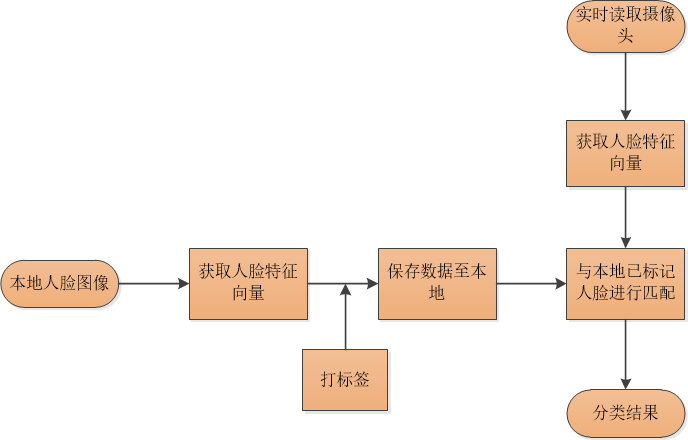
人脸识别分为人脸检测和识别两个阶段，人脸检测会找到人脸区域的矩形窗口，识别则通过ResNet返回人脸特征向量，并进行匹配。

1. 人脸检测阶段。人脸检测算法需要用大小位置不同的窗口在图像中进行滑动，然后判断窗口中是否存在人脸。dlib中使用的是HOG（histogram of oriented gradient）+ 回归树的方法，使用dlib训练好的模型进行检测效果要好很多。dlib也使用了卷积神经网络来进行人脸检测，效果好于HOG的集成学习方法。
2. 识别阶段。识别也就是我们常说的“分类”，摄像头采集到这个人脸时，让机器判断是张三还是其他人。分类分为两个部分：

• 特征向量抽取。本文用到的是dlib中已经训练好的ResNet模型的接口，此接口会返回一个128维的人脸特征向量。

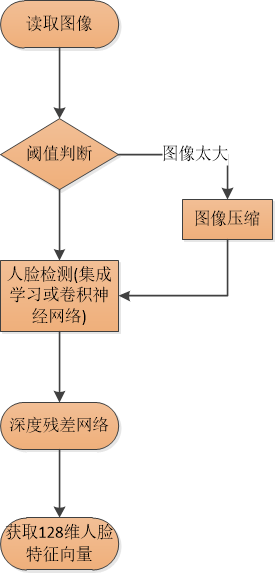
• 距离匹配。在获取特征向量之后可以使用欧式距离和本地的人脸特征向量进行匹配，使用最近邻分类器返回样本的标签。

根据上述内容，识别的大致过程如图x所示：



**图x.人脸识别分类过程**

对于图x中的获取人脸特征向量，其过程如图x所示：



**图x.获取人脸特征向量过程**

### 3.3.2地图路径规划

路径规划API是一套以HTTP形式提供的步行、公交、驾车查询及行驶距离计算接口，返回JSON 或 XML格式的查询数据，用于实现路径规划功能的开发。 由于道路/数据/算法的变更，很可能存在间隔一段时间后请求相同起终点的经纬度返回不同结果。

第一步，申请”Web服务API”密钥（Key）；

第二步，拼接HTTP请求URL，第一步申请的Key需作为必填参数一同发送；

第三步，接收HTTP请求返回的数据（JSON或XML格式），解析数据。

如无特殊声明，接口的输入参数和输出数据编码全部统一为UTF-8。

针对步行、公交、驾车这三种路径规划的接口，实现批量请求，使用批量请求接口（https://lbs.amap.com/api/webservice/guide/api/batchrequest）

### 3.3.3摄像头调用

### 3.3.4 ECharts可视化图表

ECharts一款基于HTML5的图形库，图形种类丰富，风格搭配美观。

引用ECharts首先需要下载ECharts.js文件。ECharts代码的编写基于JavaScript,本项目通过ajax异步请求获取数据库数据作为可视化图表的数据来源，从而创建可视化图表统计医院一些相关数据在某一段时间内的变化，同时当数据库中数据更新的时候，图表中数据也能随之动态变化。

## 3.4前后端设计

## 3.5重要功能函数说明

### 3.5.1 \_\_init\_\_

|  |  |  |  |
| --- | --- | --- | --- |
| \_\_init\_\_ | | | |
| **函数说明：**初始化模型配置环境 | | | |
| **参数** | 参数说明 | 返回值 | 返回值说明 |
| **self** | 函数本身参数 | NoneType | 无 |
| **predictor\_path** | 模型地址 | NoneType | 无 |
| **face\_rec\_model\_path** | 模型地址 | NoneType | 无 |

### 3.5.2 face\_detection

|  |  |  |  |
| --- | --- | --- | --- |
| face\_detection | | | |
| **函数说明：**人脸相似度对比 | | | |
| **参数** | 参数说明 | 返回值 | 返回值说明 |
| **self** | 函数本身参数 | NoneType | 无 |
| **url\_img\_1** | 图片1 | NoneType | 无 |
| **url\_img\_2** | 图片2 | NoneType | 无 |

# 4.测试报告

【填写说明：包括测试报告和技术指标。为了保证作品质量，建议多进行测试，并将测试过程、测试结果、修正过程或结果形成文档，也可以将本标题修改为主要测试，撰写主要测试过程结果及其修正；根据测试结果，形成多维度技术指标，包括：运行速度、安全性、扩展性、部署方便性和可用性等。本部分简要说明即可，减少常识性内容。】

# 5.安装及使用

【填写说明：简要说明安装环境要求、安装过程、主要流程等。建议包含默认安装和典型使用流程。】

## 5.1环境要求

|  |  |  |
| --- | --- | --- |
| 分类 | 名称 | 版本 |
| 开发工具 | pycharm | 5 |
| eclipse | 4.7 |
| 编译环境 | python | 3.6 |
| Java(TM) SE Runtime Environment | 1.8.0 |
| Tomcat | 9.0 |
| 重要库 | cmake | 3.14.3 |
| boost | 1.70.0 |
| dlib | 19.7.0 |
| numpy | 1.16.2 |
| opencv-python | 4.1.0.25 |
| 数据库平台 | mysql | 5.7.21 |

## 5.2 安装流程

### 5.2.1 Python端环境安装

1、安装python。Windows、Linux系统安装方式不同，具体方法如下所示：

①在Windows上安装Python。首先，根据Windows版本（64位还是32位）从Python的官方网站下载Python 3.7对应的64位安装程序或32位安装程序，然后，运行下载的EXE安装包。

②安装依赖包。首先安装gcc编译器，然后安装其他依赖包；下载python3.6.0源码，根据需求下载。

解压Python-3.6.0.tgz：tar -zxvf Python-3.6.0.tgz

建立一个空文件夹，用于存放python3程序：mkdir /usr/local/python3

执行配置文件，编译，编译安装：

cd Python-3.7.0

./configure --prefix=/usr/local/python3

make && make install

安装完成没有提示错误便安装成功。

1. 安装python重要依赖库。

①安装opencv-python：pip install opencv-contrib-python

②安装numpy：pip install numpy

③安装dlib：pip install dlib

④下载cmake，下载地址：<https://cmake.org/>

⑤下载boost，下载地址：https://www.boost.org/

1. Window环境下载pycharm软件，下载地址：http://www.jetbrains.com/pycharm/。Linux环境则无需安装，可直接编译相应文件。

### 5.2.2 Java端环境安装

1、安装JDK开发环境。下载网站：[http://www.oracle.com/](http://www.oracle.com/index.html)，根据电脑系统选择相应的安装版本。配置JAVA环境变量。

1. Eclipse下载安装。下载网站：[https://www.eclipse.org/,选择电脑系统匹配的32位或64位版本进行下载。](http://www.eclipse.org/downloads/，选择电脑系统匹配的32位或64位版本进行下载。)
2. tomcat下载安装与配置。下载网站：https://tomcat.apache.org/，下载相应zip文件，安装解压与环境变量配置。

### 5.2.3数据库导入

# **6.项目总结**

【填写说明：作品制作开发过程中的一些感悟和后续升级等，如：项目协调、任务分解、面对困难、水平提升、升级演进、商业推广等诸方面。建议部分篇幅不超过A4纸1页。】

## **6.1项目优化**

### **6.1.1后续升级**

本项目具有较大的潜在开发空间，其中可本项目已包含的部分进行二次开发与性能优化，如：

1. 人脸识别技术性能优化，提升图像对比速度，增强数据集测试。
2. 地图导航与路径规划，后续添加多因素，实现更优路径推荐。
3. 美化网站页面，缩短页面链接响应时间。

### **6.1.2功能扩展**

根据不同需求和应用环境及对象的变化，可进行功能扩展，实现功能适应性。若网站的业务范围扩大，实现多区域医院科室挂号，并且允许采集医院相关监控数据，则可进行下列功能添加：

1. 基于树莓派的视频端采集数据，通过网站后台行为分析算法计算研究，判断当前用户行为是否违规（如医闹行为，斗殴行为等），并进行相应警报提示。
2. 网站端添加基于自然语言处理的智能问答模块，将用户提问与标准问题匹配。
3. 智能人脸识别追踪数据库中的黑名单人员在医院中的出现情况，作出相应机制处理。

### **6.1.3商业推广**

通过线上线下结合模式，对本项目进行商业推广。线下推广主要采用医院宣传、团队人员管理等手段，与相关医院进行合作的前提下，依靠精炼有趣的平面广告或软文进行现场宣传，达到推广目的。线上推广主要借助网络平台进行高性价比且快速的推广方案，包括各种网络调研、网络广告、网络分销、网络服务等网络推广活动。

## 6.2 小组感悟

**任务分解：**本项目因存在核心创新技术点，所以任务点大致分为以下几点：项目创意设计、前端页面设计、核心算法设计与研究、前后端适配连接、信息采集与分析、网站部署与测试、文档编辑、视频制作、文档初审与复核。

**项目协调：**根据任务分解，将各任务点划分相应模块并分配给相应开发人员。负责人负责项目创意与内容设计以及项目全局进度把控，将前端、后台部分分配至两位技术人员，两人进行前后端适配连接以及测试。全体开发人员抓准主要矛盾，合理解决问题，切实有效的实现项目保质保量完工。

通过本项目的开发，我们对此项目的开发从开始迷茫到现在的熟悉，加深了对项目开发流程以及相应技术的理解。其中我们遇到了不少困难，如前端开发页面逻辑设计，生物设别技术使用等，经过查阅资料并认真思考解决了这些困难。同时我们进一步懂得了如何让大家高效协作,来一起完成我们艰巨的任务,如何协调组员之间观点，团队的力量是无量的,一个好的团队可以攻破不可思议的难题。