****

**毕 业 论 文**

题 目 英语写作智能辅助系统

的设计与实现

姓 名 康宇辰

学 号 14080004

指导教师 于学军

日 期

**独 创 性 声 明**

本人声明所呈交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京工业大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

签名： 日期：

**关于论文使用授权的说明**

本人完全了解北京工业大学有关保留、使用学位论文的规定，即：学校有权保留送交论文的复印件，允许论文被查阅和借阅；学校可以公布论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存论文。

（保密的论文在解密后应遵守此规定）

签名： 导师签名： 日期：

**摘要**

文化交融的今天，大量的用户有英语写作需求但缺少足够写作能力。就自然语言处理的发展现状而言，我们已经有能力实现一些功能为用户提供英语写作时的查错、纠正、智能提示等辅助功能，帮助用户更好地进行英语写作，并提高英语写作能力。

英语写作智能辅助系统是包括Android移动端和Web后台管理端两方面的在线系统。Android移动端为用户提供方便快捷的英语写作、展示、浏览平台，并在写作时提供实时智能提示功能。Web后台管理端为平台管理员提供实时管理平台人员、文章等信息的平台，并提供不同权限实现分别管理。本系统后台搭建主要使用Spring + Spring MVC + MyBatis框架，并通过集成其他工具实现自然语言分析等核心业务功能。本系统功能较为全面，界面简洁美观，操作友好，可拓展性强，具有实际使用价值，基本完成了既定需求。

本文主要介绍英语写作智能辅助系统的设计和实现过程，包括系统分析、总体设计、设计实现和系统测试等。

关键词：英语写作 ；软件工程 ；自然语言处理 ；安卓

**Abstract**

Today, with cultural blending, a large number of users need English writing but lack enough writing ability. As far as Natural Language Processing is concerned, we have the ability to provide some functions for the user to provide some auxiliary functions, such as error checking, correction, and intelligent hint, to help the users to write better in English and to improve their English writing ability.

The English Assistant is an online system that includes two aspects: the Android mobile terminal and the Web backstage management terminal. Android mobile terminal provides users with a convenient and efficient platform for writing, displaying and browsing English, and provides real-time intelligent prompting in writing. The Web backstage management side provides platform managers with real-time management platform personnel, articles and other information platform, and provides different permissions to achieve separate management. The background building of the system mainly uses Spring + Spring MVC + MyBatis framework, and integrates other tools to realize the core business functions such as natural language analysis. The system has comprehensive functions, simple and beautiful interface, friendly operation, expansibility, practical value, and basically completed the established demand.

This paper mainly introduces the design and implementation of intelligent assistant system for English writing, including system analysis, overall design, design and implementation, and system testing.

**Keywords：**English writing; Software engineering; Natural Language Processing; Android.

**目录**

[**摘要** I](#_Toc514226423)

[**Abstract** II](#_Toc514226424)

[1. **绪论** 1](#_Toc514226425)

[**1.1** **自然语言处理的发展情况** 1](#_Toc514226426)

[**1.2** **英语写作智能辅助系统的概述** 1](#_Toc514226427)

[**2.** **系统分析** 2](#_Toc514226428)

[**2.1** **需求概述** 2](#_Toc514226429)

[**2.2** **功能需求** 2](#_Toc514226430)

[**2.2.1** **移动客户端功能需求** 2](#_Toc514226431)

[**2.2.2** **Web管理端功能需求** 2](#_Toc514226432)

[**2.3** **对涉及工具和技术的相关说明** 2](#_Toc514226433)

[**2.3.1** **Spring + Spring MVC + MyBatis框架** 2](#_Toc514226434)

[**2.3.2** **Bootstrap前端框架** 5](#_Toc514226435)

[**2.3.3** **Android端集成工具** 5](#_Toc514226436)

[**2.3.4** **Languagetool语法检查工具** 6](#_Toc514226437)

[**2.3.5** **Stanford CoreNLP工具** 7](#_Toc514226438)

[**2.3.6** **GloVe词向量模型** 8](#_Toc514226439)

[**2.4** **本章小结** 8](#_Toc514226440)

[3. **总体设计** 9](#_Toc514226441)

[**3.1** **总体功能** 9](#_Toc514226442)

[**3.1.1** **移动客户端总体功能** 9](#_Toc514226443)

[**3.1.2** **Web管理端总体功能** 10](#_Toc514226444)

[**3.2** **数据库模型** 10](#_Toc514226445)

[**3.2.1** **管理员表** 10](#_Toc514226446)

[**3.2.2** **点击关联表** 11](#_Toc514226447)

[**3.2.3** **收藏关联表** 11](#_Toc514226448)

[**3.2.4** **积分变动表** 11](#_Toc514226449)

[**3.2.5** **日志表** 12](#_Toc514226450)

[**3.2.6** **分享关联表** 12](#_Toc514226451)

[**3.2.7** **文章表** 13](#_Toc514226452)

[**3.2.8** **用户表** 14](#_Toc514226453)

[**3.2.9** **首句提示表** 15](#_Toc514226454)

[**3.2.10** **免费提示表** 15](#_Toc514226455)

[**3.2.11** **权限地址对应表** 16](#_Toc514226456)

[**3.2.12** **地址码表** 16](#_Toc514226457)

[**3.2.13** **在线统计表** 16](#_Toc514226458)

[**3.2.14** **文章统计表** 17](#_Toc514226459)

[**3.3** **本章小结** 17](#_Toc514226460)

[**4.** **设计实现** 18](#_Toc514226461)

[**4.1** **Android端设计实现** 18](#_Toc514226462)

[**4.1.1** **登录** 18](#_Toc514226463)

[**4.1.2** **文章** 19](#_Toc514226464)

[**4.1.3** **发现** 20](#_Toc514226465)

[**4.1.4** **写作** 21](#_Toc514226466)

[**4.1.5** **排行榜** 27](#_Toc514226467)

[**4.1.6** **我的** 28](#_Toc514226468)

[**4.2** **Web端设计实现** 29](#_Toc514226469)

[**4.2.1** **登录** 29](#_Toc514226470)

[**4.2.2** **用户列表** 31](#_Toc514226471)

[**4.2.3** **文章列表** 31](#_Toc514226472)

[**4.2.4** **平台文章** 32](#_Toc514226473)

[**4.2.5** **在线统计** 33](#_Toc514226474)

[**4.2.6** **文章统计** 34](#_Toc514226475)

[**4.2.7** **日志管理** 35](#_Toc514226476)

[**4.2.8** **管理员列表** 35](#_Toc514226477)

[**4.2.9** **注册审核** 36](#_Toc514226478)

[**4.2.10** **修改密码** 37](#_Toc514226479)

[**4.2.11** **登出** 37](#_Toc514226480)

[**4.3** **本章小结** 37](#_Toc514226481)

[**5.** **系统测试** 39](#_Toc514226482)

[**5.1** **测试环境搭建** 39](#_Toc514226483)

[**5.1.1** **JDK环境配置** 39](#_Toc514226484)

[**5.1.2** **Tomcat安装** 39](#_Toc514226485)

[**5.1.3** **Android模拟器Genmotion安装** 39](#_Toc514226486)

[**5.1.4** **Mysql数据库安装** 39](#_Toc514226487)

[**5.2** **项目部署和准备** 39](#_Toc514226488)

[**5.2.1** **服务器端项目部署** 39](#_Toc514226489)

[**5.2.2** **Android项目测试准备** 39](#_Toc514226490)

[**5.2.3** **Web项目测试准备** 40](#_Toc514226491)

[**5.3** **系统测试** 40](#_Toc514226492)

[**5.3.1** **Android端测试** 40](#_Toc514226493)

[**5.3.2** **Web端测试** 41](#_Toc514226494)

[**5.4** **本章小结** 43](#_Toc514226495)

[**结论** 44](#_Toc514226496)

[**参考文献** 45](#_Toc514226497)

[**致谢** 46](#_Toc514226498)

# 绪论

* 1. **自然语言处理的发展情况**

随着社会的日益信息化,计算机学科也在逐步发展成熟,人们也越来越强烈地希望能更好地同计算机进行交流。而伴随信息爆炸而来的是更为广阔的信息检索需求、日益严峻的信息安全问题以及潜力十足的机器翻译市场。如何让计算机实现自动的语言处理功能、挖掘和有效利用海量语言信息显得尤为重要,也对于人工智能研究领域提出了更新的要求。

自然语言处理的研究历史虽然不是很长,但就目前已经取得的成果和地位足以显示它的重要性和广阔的应用前景。就目前的研究成果来看,部分问题已经得到了解决,可以为人们提供可靠的辅助性帮助,比如专业领域文档翻译,电子词典,搜索引擎,文字录入等。但随着信息服务、通讯、网络内容管理、情报处理、国家安全等社会需求的日益迫切,很多基础问题的研究仍然任重道远:比如语义表示和计算、比如高质量的自动翻译。目前所进行的研究中,暴露和存在的问题还有很多很多,研究中仍然缺乏坚实的理论基础;词汇句法方面的问题尚未解决,已开始挑战语义、知识等深层次课题;语音识别中采用的统计语言模型推动了发展,于是目前的统计模型在迅速向语言深层发展等等等等。所以,许多技术离真正实用其实还有不少距离,若干理论问题也有待更深入的研究。我们可以从现有的模型和方法改进入手,在不成熟技术的基础上开发出更实用的人机系统,我们也有理由期待更有效和更健全地理论体系。【1】

* 1. **英语写作智能辅助系统的概述**

同样的随着文化交融和发展，无论是日常英文写作还是留学推荐信等，英文写作的需求普遍存在，但更多的时候我们的写作能力可能有所欠缺，所以本系统旨在为有英语写作需求的用户，提供英语写作时的查错、纠正、智能提示等辅助功能，帮助用户更好地进行英语写作，并提高英语写作能力。

**系统分析**

* 1. **需求概述**

1.根据实际业务需求，定制整体解决方案，完成整体平台设计与开发。

2.核心业务层应完成对自然语言（英语）的语法分析，分词、词性标注、专名识别等功能，并通过依存句法分析、词向量、DNN语言模型、词义相似度、短文相似度、评论客观抽取、情感倾向分析等方法，完成英语写作中的实时智能提示功能。

3.依据实际业务需求，设计并实现Web平台管理端。

4.依据实际业务需求，设计并实现移动应用端。

* 1. **功能需求**

### **移动客户端功能需求**

该系统移动客户端要求用户需要注册才可以登录，登陆后可以浏览文章、发现文章、写作、查看排行榜、查看和修改个人信息。文章需要分为陈述和推荐信两种类别；浏览文章具有关键字搜索功能；发现文章具有随机推荐功能；对文章可以进行收藏、分享、点赞等操作；写作时可以使用拼写检查、免费提示、vip提示等功能，写作完成可以保存并上传；排行榜显示提分排行，可以点击头像查看其他用户信息；查看个人信息同时可以修改包括头像、昵称等信息；可以查看个人主页、我的收藏、我的分享、浏览记录等。

### **Web管理端功能需求**

该系统Web管理端要求管理员需要注册并通过审核才可以登录，登陆后不同权限管理员可以使用不同模块功能，包括用户管理、文章管理、统计信息、后台管理等。用户管理可以查看用户列表、搜索、排序、编辑、启用停用；文章管理可以查看文章列表、搜索、排序、编辑、删除或恢复、设置是否公开、发布平台文章等；统计信息可以对在线数据和文章数据等数据进行统计并展示；后台管理包括日志管理、管理员列表和管理员注册审核，日志管理应该可以清晰地展示后台管理员动作、结果和发生时间，管理员列表可以查看所有管理员信息和状态等，管理员注册审核应具备审核通过和驳回功能，并实现权限分配。

* 1. **对涉及工具和技术的相关说明**

### **Spring + Spring MVC + MyBatis框架**

SSM框架，是spring + Spring MVC + MyBatis的缩写，是继SSH之后，目前比较主流的Java EE企业级框架，适用于搭建各种大型的企业级应用系统。

Spring 是一个为了解决企业应用程序开发复杂性而创建的开源框架。框架的主要优势之一就是其分层架构，分层架构允许选择使用哪一个组件，同时为 Java EE应用程序开发提供集成的框架。其模块化方便了我们整合需要的工具，同时保持低耦合。简单来说，Spring是一个轻量级的控制反转（IoC）和面向切面（AOP）的容器框架。

IoC是一种用于解决组件（实际上也可以是简单的Java类）之间的依赖关系、配置及生命周期的设计模式，其中对组件依赖关系的处理是IoC的精华部分。IoC的实际意义就是把组件之间的依赖关系提取出来，由容器来具体配置。这样，各个组件之间就不存在直接的关联，任何组件都可以最大程度的得到重用。运用了IoC模式后我们不需要再自己管理组件之间的依赖关系，只需要声明由容器去实现这种依赖关系。就好比把对组件之间依赖关系的控制进行了倒置，不再有组件自己来建立这种依赖关系而交给容器。【2】

采用面向对象进行编程（OOP）时,其封装技术极大地提高了代码的复用能力和系统的设计能力。然而有时候OOP无法避免一些重复冗余的代码,例如在处理核心业务逻辑时,还要实现日志记录,安全检查及事务管理等功能,这些代码通常并不属于业务组件本身,是一个个独立的模块,但是为了保证系统的健壮性,就要在每个业务方法里都反反复复地编写这些代码,造成了冗余,从系统角度来看这些代码通常分散于各处,和具体的业务代码有横切关系,具有横切性,利用OOP很难将这些代码剔除,而AOP可以很好地解决这个问题。一般而言,关注点可划分为两类:核心关注点(core con-cern)和横切关注点(crosscutting concern)。其中,核心关注点主要关注系统的业务逻辑;而横切关注点主要关注的是系统级的公共服务,具有横切性的代码,比如日志记录。在OOP处理这些横切代码能力表现不足时,AOP为开发者提供了一种模块化横切关注点的机制,并能够将横切关注点以某种方式织入到面向对象的软件系统中,从而完成横向功能的模块化,避免代码复用,提高系统的可维护性。【3】

在使用Spring进行WEB开发时常常会选用框架提供的MVC模块Spring MVC框架。Spring MVC框架主要由DispatcherServlet、处理器映射、处理器(控制器)、视图解析器、视图组成。它实现了MVC的核心概念,为控制器和处理程序提供了大量与此模式相关的功能。并且当向MVC添加反转控制时,它使应用程序高度解耦,提供了通过简单的配置更改即可动态更改组件的灵活性。【4】Spring MVC 分离了控制器、模型对象、分派器以及处理程序对象的角色，这种分离让它们更容易进行定制，也方便了我们的开发工作。

MyBatis是一个基于JAVA的支持定制化 SQL、存储过程以及高级映射的优秀的持久层框架。MyBatis 避免了几乎所有的 JDBC 代码和手工设置参数以及抽取结果集。MyBatis 使用简单的 XML 或注解来配置和映射基本体，将接口和 Java 的 POJOs(Plain Old Java Objects,普通的 Java对象)映射成数据库中的记录。虽然MyBatis需要生产较多的model类、mapping.xml文件和dao接口，但是我们可以通过生成工具直接从数据库生成这些文件。具体的使用方法为：在在新文件夹下拷入mybatis-3.2.8.jar、mybatis-generator-core-1.3.2.jar、mysql-connector-java-5.1.39.jar这三个jar包，并创建configuration.xml，内容大概如下：

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE generatorConfiguration

PUBLIC "-//mybatis.org//DTD MyBatis Generator Configuration 1.0//EN"

"http://mybatis.org/dtd/mybatis-generator-config\_1\_0.dtd">

<generatorConfiguration>

<!-- 数据库驱动-->

<classPathEntry location="mysql-connector-java-5.1.39.jar"/>

<context id="DB2Entity" targetRuntime="MyBatis3">

<commentGenerator>

<property name="suppressDate" value="true"/>

<!-- 是否去除自动生成的注释 true：是 ： false:否 -->

<property name="suppressAllComments" value="true"/>

</commentGenerator>

<!--数据库URL，用户名、密码 -->

<jdbcConnection driverClass="com.mysql.jdbc.Driver" connectionURL="jdbc:mysql://localhost/test" userId="root" password="root">

</jdbcConnection>

<javaTypeResolver>

<property name="forceBigDecimals" value="false"/>

</javaTypeResolver>

<!-- 生成实体的包名和位置-->

<javaModelGenerator targetPackage="test.model" targetProject="MyProject\src">

　　　　　　<property name="enableSubPackages" value="true" />

　　　　　　<property name="trimStrings" value="true" />

　　　　 </javaModelGenerator>

<!-- 生成映射文件的包名和位置-->

<sqlMapGenerator targetPackage="test.mapping" targetProject="MyProject\src">

<property name="enableSubPackages" value="true"/>

</sqlMapGenerator>

<!-- 生成DAO的包名和位置-->

<javaClientGenerator type="XMLMAPPER" targetPackage="test.dao" targetProject="MyProject\src">

<property name="enableSubPackages" value="true"/>

</javaClientGenerator>

<!-- 要生成的表 tableName是数据库中的表名或视图名 domainObjectName是实体类名 -->

<table tableName="person" domainObjectName="Person" enableCountByExample="false" enableUpdateByExample="false"

　　　　　　　　 enableDeleteByExample="false" enableSelectByExample="false" selectByExampleQueryId="false"></table>

</context>

</generatorConfiguration>

在当前目录下打开控制台，输入java -jar mybatis-generator-core-1.3.2.jar -configfile configuration.xml –overwrite就可以自动生成对应的文件，再将生成的所需文件拷入项目中，这样就完成了MyBatis所需文件的自动生成。

### **Bootstrap前端框架**

Bootstrap是Twitter推出的一个用于前端开发的开源工具包，使得Web开发更加快捷。它由Twitter的设计师Mark Otto和Jacob Thornton合作开发，是一个CSS/HTML框架。同时它提供了些基于其开发的优质工具例如bootstrap-datetimepicker和bootstrap-fileinput等。Bootstrap可以让我们的Web开发更迅速、简单。使用的方法为：从<http://www.bootcss.com/>，Bootstrap中文网上下载Bootstrap 的预编译版本，并将其导入本项目中。在jsp页面上使用类似如下代码：

<link rel="stylesheet" href="static/lib/bootstrap/css/bootstrap.css">

<script src="static/lib/jquery-1.11.1.min.js"></script>

<script src="static/lib/bootstrap/js/bootstrap.js"></script>

让页面使用 Bootstrap 的模板。

### **Android端集成工具**

(1) volley

volley 是 Goole I/O 2013上发布的网络通信库，使网络通信更快、更简单、更健壮。Volley既可以访问网络取得数据，也可以加载图片，并且在性能方面也进行了大幅度的调整，它的设计目标就是非常适合去进行数据量不大，但通信频繁的网络操作，而对于大数据量的网络操作，比如说下载文件等，Volley的表现就会非常糟糕。所以在该项目中我们选择volley进行基本的网络通信。

(2) xUtils

xUtils是基于Afinal开发的目前功能比较完善的一个Android开源框架，包含了很多实用的android工具。我们主要是用xUtils的HttpUtils模块补充volley对于大数据量网络通信的缺陷，该项目中主要是用于用户头像的上传下载。

(3) AMap\_Location

AMap\_Location 高德定位服务包，用于得到当前的所在地点。

(4) Universal-Image-Loader

Universal-Image-Loader是一个开源的Android图片加载框架，只需要简单的配置就能完成缓存，图片下载，显示图片这些工作。

(5) Gson

Gson 是google解析Json的一个开源框架，该项目Android端使用Gson解析网络请求中的Json数据。

### **Languagetool语法检查工具**

LanguageTool是一个用于英语，法语，德语，波兰语，俄语和其他20多种语言的开源校对软件。它能够发现简单的拼写检查器无法检测到的许多错误,例如动词第三人称单数甚至标点符号和空格错误等。我们使用LanguageTool实现写作中的拼写检查功能。

安装及使用：访问<https://github.com/languagetool-org/languagetool>，在branch/tags处找到tags下v4.0版本，点击Clone or download下载源码，选择zip格式，下载后可以修改源码中language-module.properties配置文件去掉不需要的语言并删除相应文件，在本项目中我们只选择留下English、AmericanEnglish和BritishEnglish，然后导入项目中（导入源码或者打包jar），使用类似如下代码：

static private JLanguageTool AlangTool = new JLanguageTool(new AmericanEnglish());

List<RuleMatch> matches = null;

matches = AlangTool.check(str);

来调用LanguageTool实现拼写检查功能。

### **Stanford CoreNLP工具**

Stanford CoreNLP是一个集成的框架，是一种可扩展的管道模型，可提供核心自然语言分析。这个工具包在NLP研究社区以及开源NLP技术的商业和政府用户中都有相当广泛的应用。其设计者使其遵循简单，平易近人的设计，直观的界面，包含强大和高质量的分析组件，并且不需要使用大量关联包。【5】框架的目标是使得应用一大堆语言分析工具分析大量的文本变得简单。Stanford CoreNLP工具可以仅仅通过两行命令执行大量的文本分析工作。框架设计的初衷就是高度灵活的可扩展性的。通过一个单独的命令就可以选择某个工具的开启和关闭。Stanford CoreNLP集成了许多斯坦福的NLP工具，包括：词性标记（POS），命名实体识别（NER），语法，参数分析系统，情绪分析，自举模式学习，和开放信息提取工具。该项目中我们将在写作中使用Stanford CoreNLP工具分析前文内容从而提取感情或关键词等。

安装和使用：从<http://nlp.stanford.edu/software/stanford-corenlp-full-2016-10-31.zip>下载Stanford CoreNLP工具包，解压后把所有jar包导入项目中，使用类似如下代码：

Properties props = new Properties();

props.setProperty("annotators", "tokenize, ssplit, pos, lemma, ner, parse, dcoref, sentiment");

StanfordCoreNLP pipeline = new StanfordCoreNLP(props);

Annotation document = new Annotation(str);

pipeline.annotate(document);

for (CoreMap sentence : sentences) {

// traversing the words in the current sentence

// a CoreLabel is a CoreMap with additional

// token-specific methods

for (CoreLabel token: sentence.get(CoreAnnotations.TokensAnnotation.class)) {

// this is the text of the token

String word = token.get(CoreAnnotations.TextAnnotation.class);

// this is the POS tag of the token

String pos = token.get(CoreAnnotations.PartOfSpeechAnnotation.class);

// 获取词性识别结果

// this is the NER label of the token

String ne = token.get(CoreAnnotations.NamedEntityTagAnnotation.class); // 获取命名实体识别结果

String lemma = token.get(LemmaAnnotation.class);

// 获取词形还原结果

System.out.println(word+"\t"+pos+"\t"+lemma+"\t"+ne);

}

// this is the parse tree of the current sentence

Tree tree = sentence.get(TreeCoreAnnotations.TreeAnnotation.class);

System.out.println("语法树：");

System.out.println(tree.toString());

// this is the Stanford dependency graph of the current sentence

SemanticGraph dependencies = sentence.get(SemanticGraphCoreAnnotations.CollapsedCCProcessedDependenciesAnnotation.class);

System.out.println("依存句法：");

System.out.println(dependencies.toString());

}

此处仅演示了直接输出段落内全部单词、词性、命名实体、原型，以及段落中每个句子的语法书和依存句法。

### **GloVe词向量模型**

GloVe（Global vectors for word representation）是一个新的全局对数双线性回归模型，用于无监督地进行单词表示的学习，该模型在词类比，单词相似性和命名实体识别任务方面优于其他模型。【6】同时GloVe也是现在较为常见的基于矩阵的分布式表示的词向量模型。【7】在该项目中我们只需要一个预训练好的GloVe词向量模型来实现得到关键词的相关词汇，其语料库为Wikipedia+Gigaword 5。

下载地址：<http://nlp.stanford.edu/data/glove.6B.zip>。出于项目大小考虑选择使用了最小维度的50维的模型。

* 1. **本章小结**

本章我们明确了英语写作智能辅助系统的总体需求和功能需求，以及涉及到的工具和技术，并对这些工具和技术有了简要的说明介绍。

**总体设计**

* 1. **总体功能**

### **移动客户端总体功能**

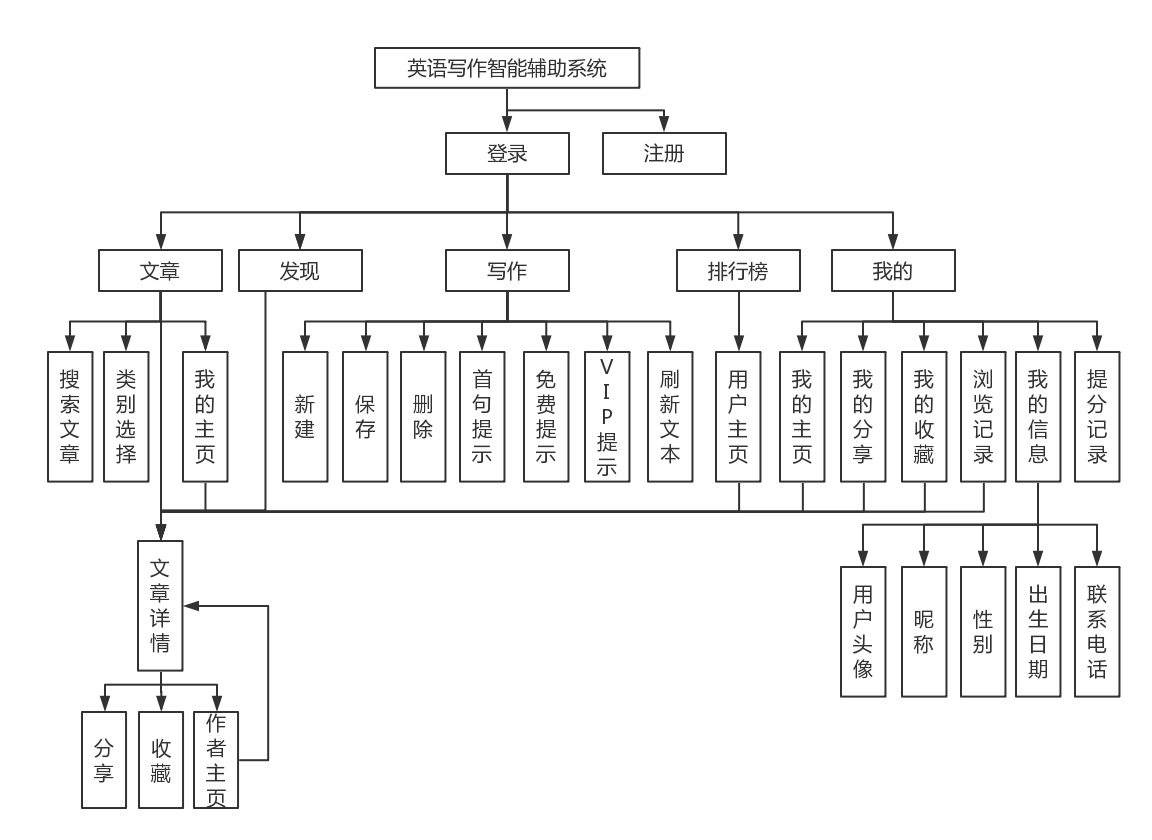


图1 移动客户端总体功能

### **Web管理端总体功能**

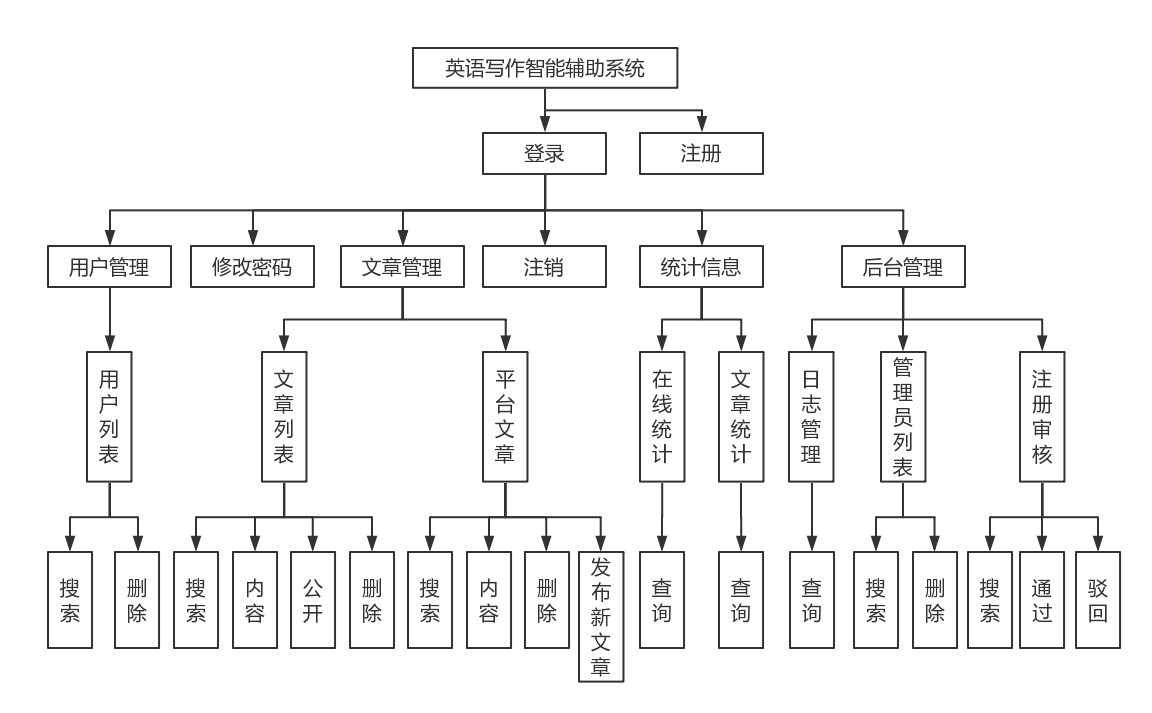


图2 Web管理端总体功能

* 1. **数据库模型**

### **管理员表**

该表中各字段设置如表1所示，id字段为主键自增，其中role字段-1为待审核，0为全部权限，1为综合管理员，注册后的管理员账号默认为待审核状态，待注册审核通过后由进行操作的管理员分配新的权限。isDeleted字段0代表正常，1代表逻辑删除。

表1 管理员表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **代码** | **名称** | **数据类型** | **主要的** | **强制** | **注释** |
| id | 管理员ID | bigint | TRUE | TRUE |  |
| phone | 电话号码 | varchar | FALSE | FALSE |  |
| loginName | 用户名 | varchar | FALSE | FALSE |  |
| password | 密码 | varchar | FALSE | FALSE |  |
| role | 权限 | int | FALSE | FALSE | -1为待审核/0为全部权限/1为综合管理员 |
| isDeleted | 是否删除 | tinyint | FALSE | FALSE | 0正常/1删除 |

### **点击关联表**

该表中各字段设置如表2所示，user\_ID字段为逻辑外键（用户表id），text\_ID字段为逻辑外键（文章表id），user\_ID字段和text\_ID字段为双主键，该表用于存储用户在某一时刻点击并浏览文章的记录。

表2 点击关联表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **代码** | **名称** | **数据类型** | **主要的** | **强制** | **注释** |
| user\_ID | 用户ID | bigint | TRUE | TRUE |  |
| text\_ID | 文章ID | bigint | TRUE | TRUE |  |
| createTime | 创建时间 | timestamp | FALSE | TRUE |  |

### **收藏关联表**

该表中各字段设置如表3所示，user\_ID字段为逻辑外键（用户表id），text\_ID字段为逻辑外键（文章表id），user\_ID字段和text\_ID字段为双主键，该表用于存储用户在某一时刻收藏文章的记录。

表3 收藏关联表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **代码** | **名称** | **数据类型** | **主要的** | **强制** | **注释** |
| user\_ID | 用户ID | bigint | TRUE | TRUE |  |
| text\_ID | 文章ID | bigint | TRUE | TRUE |  |
| createTime | 创建时间 | timestamp | FALSE | TRUE |  |

### **积分变动表**

该表中各字段设置如表4所示，id字段为主键自增，user\_ID字段为逻辑外键（用户表id），text\_ID字段为逻辑外键（文章表id）。

表4 积分变动表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **代码** | **名称** | **数据类型** | **主要的** | **强制** | **注释** |
| id | 积分变动ID | bigint | TRUE | TRUE |  |
| uesr\_ID | 用户ID | bigint | FALSE | FALSE |  |
| text\_ID | 文章ID | bigint | FALSE | FALSE |  |
| integral | 积分变动值 | int | FALSE | FALSE |  |
| cause | 原因 | varchar | FALSE | FALSE |  |
| createTime | 创建时间 | timestamp | FALSE | TRUE |  |

### **日志表**

该表中各字段设置如表5所示，id字段为主键自增，operation用于区分操作类型，0代表登录，1代表登出，2代表注册，3代表修改密码，4代表修改（各种状态或数据），5代表上传，6代表审核。具体结果或内容会存储在result字段中，例如：可能的情况为admin\_ID字段值为1，operation字段值为4，result字段值为“文章23公开状态成功”，这表示该条记录存储了一次由管理员id为1的管理员对文章id为23的文章的公开状态的修改且结果为成功。

表5日志表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **代码** | **名称** | **数据类型** | **主要的** | **强制** | **注释** |
| id | 日志ID | bigint | TRUE | TRUE |  |
| admin\_ID | 管理员ID | bigint | FALSE | FALSE |  |
| ip | IP地址 | varchar | FALSE | FALSE |  |
| operation | 操作 | int | FALSE | FALSE | 0登录/1登出/2注册/3修改密码/4修改/5上传/6审核 |
| result | 结果 | varchar | FALSE | FALSE |  |
| createTime | 创建时间 | timestamp | FALSE | TRUE |  |

### **分享关联表**

该表中各字段设置如表6所示，user\_ID字段为逻辑外键（用户表id），text\_ID字段为逻辑外键（文章表id），user\_ID字段和text\_ID字段为双主键，该表用于存储用户在某一时刻分享文章的记录。

表6 分享关联表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **代码** | **名称** | **数据类型** | **主要的** | **强制** | **注释** |
| user\_ID | 用户ID | bigint | TRUE | TRUE |  |
| text\_ID | 文章ID | bigint | TRUE | TRUE |  |
| createTime | 创建时间 | timestamp | FALSE | TRUE |  |

### **文章表**

该表中各字段设置如表7所示，id字段为主键自增，afflatus字段，languageSense字段和mood字段均为-1代表差，0代表一般，1代表好。Type字段0代表陈述，1代表推荐信。isPrivate字段0代表公开，1代表私有。isPlatform字段0代表用户文章，1代表平台推荐文章。isDeleted字段0代表正常，1代表逻辑删除。

表7 文章表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **代码** | **名称** | **数据类型** | **主要的** | **强制** | **注释** |
| id | 文章ID | bigint | TRUE | TRUE |  |
| title | 文章标题 | varchar | FALSE | FALSE |  |
| content | 文章内容 | mediumtext | FALSE | FALSE |  |
| address | 地址 | varchar | FALSE | FALSE |  |
| afflatus | 灵感 | int | FALSE | FALSE | 暂定0一般 |
| languageSense | 语感 | int | FALSE | FALSE | 暂定0一般 |
| mood | 心情 | int | FALSE | FALSE | 暂定0一般 |
| type | 文章类型 | int | FALSE | FALSE | 0陈述/ 1推荐信 |
| clickNumber | 点击量 | int | FALSE | FALSE |  |
| collectionNumber | 收藏量 | int | FALSE | FALSE |  |
| shareNumber | 分享量 | int | FALSE | FALSE |  |
| createTime | 创建时间 | timestamp | FALSE | TRUE |  |
| editTime | 修改时间 | timestamp | FALSE | TRUE |  |
| user\_ID | 创建人ID | bigint | FALSE | FALSE |  |
| isPrivate | 是否私有 | tinyint | FALSE | FALSE | 0公开/1私有 |
| isDeleted | 是否删除 | tinyint | FALSE | FALSE | 0正常/1删除 |
| isPlatform | 是否平台推荐 | tinyint | FALSE | FALSE | 0正常/1平台推荐 |

### **用户表**

该表中各字段设置如表8所示，id字段为主键自增。photo字段仅存储服务器端地址的文件名，具体路径由项目内配置。isVIP字段0代表非VIP用户，1代表VIP用户。isOnline字段0代表用户不在线，1代表用户在线。isDeleted字段0代表正常，1代表逻辑删除。

表8 用户表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **代码** | **名称** | **数据类型** | **主要的** | **强制** | **注释** |
| id | 用户ID | bigint | TRUE | TRUE |  |
| phone | 电话号码 | varchar | FALSE | FALSE |  |
| name | 用户名 | varchar | FALSE | FALSE |  |
| loginName | 账号 | varchar | FALSE | FALSE |  |
| password | 密码 | varchar | FALSE | FALSE |  |
| photo | 头像地址 | varchar | FALSE | FALSE |  |
| sex | 性别 | int | FALSE | FALSE |  |
| birthday | 生日 | timestamp | FALSE | FALSE |  |
| firstSentenceAll | 首句提示总量 | int | FALSE | FALSE |  |
| firstSentenceUse | 首句提示用量 | int | FALSE | FALSE |  |
| freeHintAll | 免费提示总量 | int | FALSE | FALSE |  |
| freeHintUse | 免费提示用量 | int | FALSE | FALSE |  |
| vipHintAll | VIP提示总量 | int | FALSE | FALSE |  |
| vipHintUse | VIP提示用量 | int | FALSE | FALSE |  |
| integral | 积分 | int | FALSE | FALSE |  |
| isVIP | 是否VIP | tinyint | FALSE | FALSE | 0否/1是 |
| isDeleted | 是否删除 | tinyint | FALSE | FALSE | 0正常/1删除 |
| isOnline | 是否在线 | tinyint | FALSE | FALSE | 0否/1是 |

### **首句提示表**

该表中各字段设置如表9所示，id字段为主键自增。isDeleted字段0代表正常，1代表逻辑删除。该表用于存储首句提示的全部提示数据。

表9 首句提示表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **代码** | **名称** | **数据类型** | **主要的** | **强制** | **注释** |
| id | ID | bigint | TRUE | TRUE |  |
| content | 提示内容 | text | FALSE | FALSE |  |
| sentiment | 情感值 | double | FALSE | FALSE |  |
| usep | 正情感使用次数 | bigint | FALSE | FALSE |  |
| usen | 负情感使用次数 | bigint | FALSE | FALSE |  |
| isDeleted | 是否删除 | tinyint | FALSE | FALSE | 0正常/1删除 |

### **免费提示表**

该表中各字段设置如表10所示，id字段为主键自增。isDeleted字段0代表正常，1代表逻辑删除。该表用于存储免费提示的全部提示数据。

表10 免费提示表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **代码** | **名称** | **数据类型** | **主要的** | **强制** | **注释** |
| id | ID | bigint | TRUE | TRUE |  |
| content | 提示内容 | text | FALSE | FALSE |  |
| sentiment | 情感值 | double | FALSE | FALSE |  |
| usep | 正情感使用次数 | bigint | FALSE | FALSE |  |
| usen | 负情感使用次数 | bigint | FALSE | FALSE |  |
| isDeleted | 是否删除 | tinyint | FALSE | FALSE | 0正常/1删除 |

### **权限地址对应表**

该表中各字段设置如表11所示。role字段为逻辑外键（管理员表role），urlid字段为逻辑外键（地址码表id），role字段和urlid字段为双主键，该表用于存储管理员权限对应的可访问地址。

表11 权限地址对应表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **代码** | **名称** | **数据类型** | **主要的** | **强制** | **注释** |
| role | 权限ID | bigint | TRUE | TRUE |  |
| urlid | 地址ID | bigint | TRUE | TRUE |  |

### **地址码表**

该表中各字段设置如表12所示，id字段为主键自增。url字段存储实际的访问路径例如：“/user/userlist”（用户列表）。该表用于可访问地址id对应的实际地址。

表12地址码表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **代码** | **名称** | **数据类型** | **主要的** | **强制** | **注释** |
| id | ID | bigint | TRUE | TRUE |  |
| url | 地址 | varchar | FALSE | FALSE |  |

### **在线统计表**

该表中各字段设置如表13所示，id字段为主键自增。该表用于存储定时任务执行时统计的在线人数记录。

表13 在线统计表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **代码** | **名称** | **数据类型** | **主要的** | **强制** | **注释** |
| id | ID | bigint | TRUE | TRUE |  |
| num | 在线人数 | int | FALSE | FALSE |  |
| createTime | 创建时间 | timestamp | FALSE | FALSE |  |

### **文章统计表**

该表中各字段设置如表14所示，id字段为主键自增。该表用于存储定时任务执行时统计的首句提示使用率、免费提示使用率和VIP提示使用率记录。

表14 文章统计表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **代码** | **名称** | **数据类型** | **主要的** | **强制** | **注释** |
| id | ID | bigint | TRUE | TRUE |  |
| firstsentencerate | 首句提示使用率 | double | FALSE | FALSE |  |
| freehintrate | 免费提示使用率 | double | FALSE | FALSE |  |
| viphintrate | VIP提示使用率 | double | FALSE | FALSE |  |
| createTime | 创建时间 | timestamp | FALSE | FALSE |  |

* 1. **本章小结**

本章我们设计了英语写作智能辅助系统的客户端和管理端的总体功能，并根据总体功能设计，利用数据库的设计思想完成了数据库的详细设计以及数据库搭建，为接下来的设计实现做好了准备工作。

**设计实现**

* 1. **Android端设计实现**

### **登录**

打开应用，显示如图3页面，已登录用户自动跳转文章页面，详见4.1.2文章，否则跳转登录页面。



图3 Logo页面

本系统的登录页面如图4所示，输入正确的账号/手机号以及正确的密码后点击登录可以成功登录系统，否则会反馈相应错误信息。登录状态通过服务器端session存储以及cookie携带Jsessionid保持，并且会更新数据库该用户在线状态字段为真。服务器端session过期时间为24小时，session过期时由SessionListener监听捕捉并更新数据库该用户在线状态字段为假。所以登录成功后24小时内不进行任何访问网络操作将会自动注销登录，届时再次使用时会提示“登录已过期，请重新登陆”并跳转至登录页面。



图4 登录页面

点击“没有账号，点击注册”链接处可以跳转注册页面，如图5，正确输入手机号格式的手机号后点击获取按钮可以获取验证码（并未实现，验证码暂不做校验），继续输入其他信息并且格式正确后点击注册可以成功注册一个账号，否则会反馈相应错误信息。账号和手机号不能与已存在账号和手机号重复，否则会反馈相应错误信息。



图5 注册页面

### **文章**

登录成功后将跳转至文章页面如图6，列表部分默认显示5个最新的文章，下拉列表刷新，上拉至底部加载更多5个文章，每个列表项包括文章标题、发布时间、用小图标代表的心情、灵感、语感、作者昵称和文章的开头节选。可以输入关键词进行模糊搜索，点击右上角菜单可以选择浏览陈述或推荐信类别。点击左上角图标进入个人主页页面，详见4.1.6我的。点击下边栏可以切换至其他页面，所在页面标签会有颜色填充。

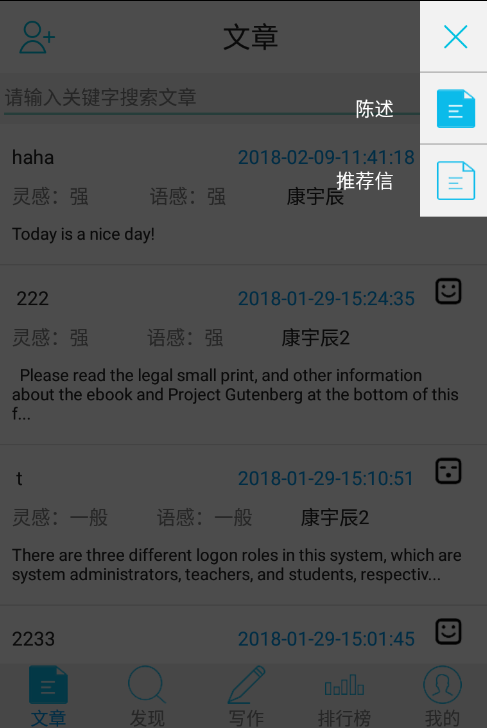
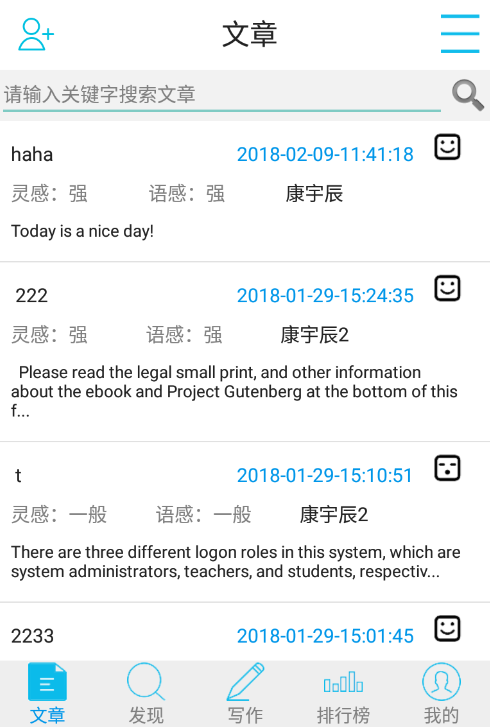


图6 文章页面

点击列表项可以打开文章详情页面，如图7，可以查看文章内容和浏览量、心情等信息，点击信息栏处可以切换显示，点击作者头像处可以打开作者的个人主页，点击右上角菜单可以看到收藏（已收藏不可见）、分享、修改（仅自己的文章可见）按钮，可进行相应操作。修改界面同4.14写作。

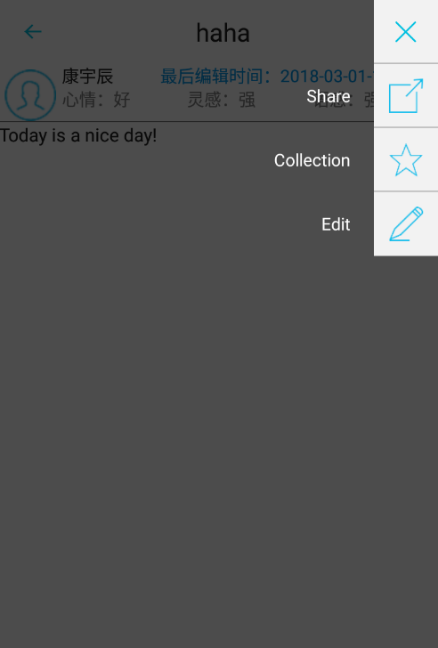
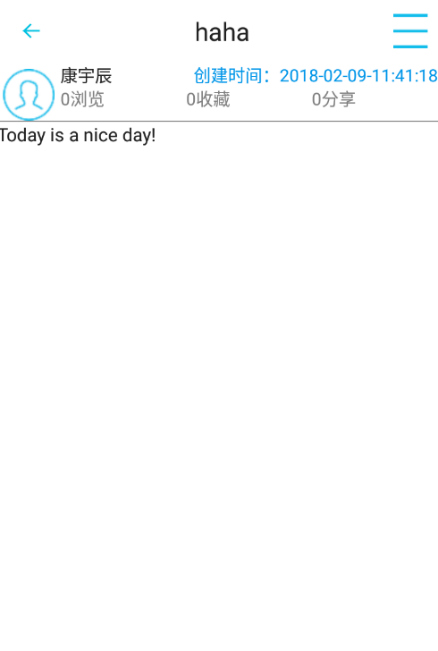


图7 文章详情页面

### **发现**

点击下边栏可以切换至发现页面，如图8，列表部分默认显示4个随机的用户文章和1个平台推荐文章（作者昵称为蓝色的“平台推荐”），下拉列表刷新，上拉至底部加载更多5个文章，每个列表项包括文章标题、发布时间、浏览量、收藏量、分享量、作者昵称和文章的开头节选。点击列表项进入文章详情。

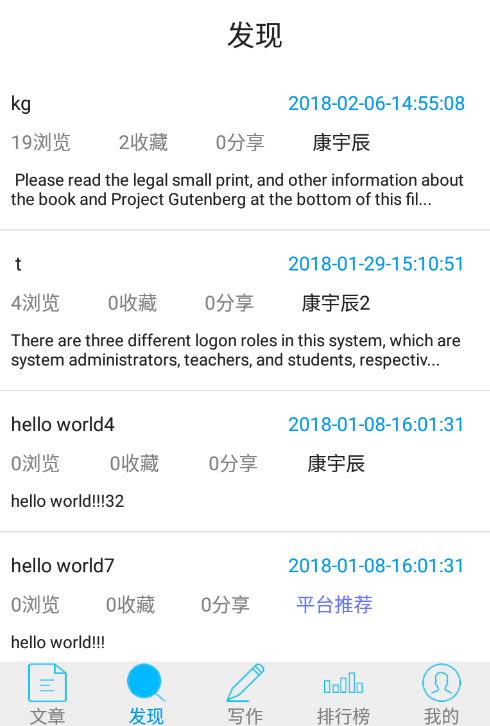


图8 发现页面

### **写作**

点击下边栏可以切换至发现页面，如图9，首先新建文章需要先选择一个类别（当前类别区分对其他功能没有影响）。写作界面中上方可以输入所在地点，下边灵感、语感、心情三个选项可以选择，而心情选项会影响下面某些提示内容。右上菜单可以进行保存和删除操作。



图9 写作页面

点击首句提示将会根据已选择的心情和以写文章内容获取首句提示，如图10。具体算法为：由Stanford CoreNLP工具取文章内容中情感有效的句子得到句情感，计算总字数（情感有效句子），总字数大于10时，

否则，

具体实现类似如下代码：

Annotation document = new Annotation(str);

pipeline.annotate(document);

List<CoreMap> sentences = document.get(CoreAnnotations.SentencesAnnotation.class);

double mainSentiment = 0;

int textl = 0;

for (CoreMap sentence : sentences) {

Tree tree = sentence.get(SentimentAnnotatedTree.class);

int sentiment = RNNCoreAnnotations.getPredictedClass(tree);

if (sentiment > 0 && sentiment < 4) {

String partText = sentence.toString();

mainSentiment += sentiment \* partText.length();

textl += partText.length();

}

}

double r = 0;

if (textl > 10)

r = mainSentiment / textl;

else

r = Integer.parseInt(selectSentiment);

取数据库中首句sentiment-平均情感绝对值最小的两个首句数据，再取其外3个随机首句数据合计5条数据随机排序构成首句提示。

具体实现类似如下代码：

lf：

SELECT id,content FROM firstsentence where isDeleted=0 order by abs(sentiment-#{r}) LIMIT 0,2;

lf2：

SELECT id,content FROM firstsentence where isDeleted=0 order by rand() LIMIT 0,5;

Set< Firstsentence > rs = new HashSet< Firstsentence >(); //首句提示

rs.addAll(lf);

for (int i = lf2.size(); i > 0; i--) { //去重

for (Firstsentence f : lf) {

if (f.getId() == lf2.get(i - 1).getId()) {

lf2.remove(lf2.get(i - 1));

break;

}

}

}

rs.add(lf2.get(0));

rs.add(lf2.get(1));

rs.add(lf2.get(2));

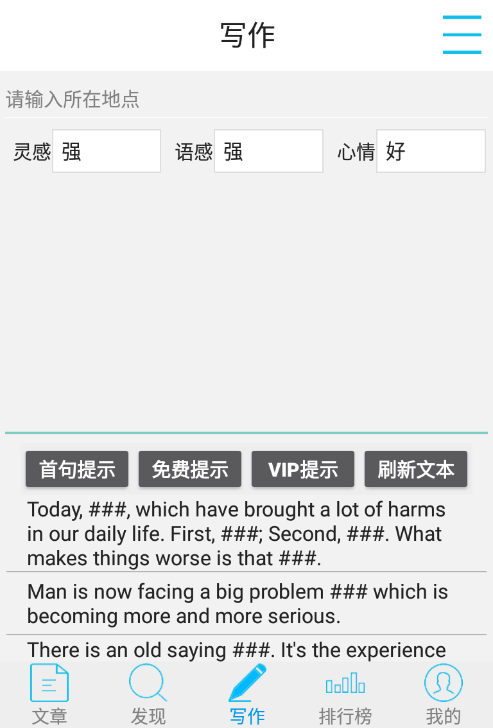


图10 首句提示

点击免费提示将会根据以写文章内容获取免费提示，如图11。具体算法为：由Stanford CoreNLP工具取文章内容情感有效句子得到句情感，计算总字数（情感有效句子），总字数大于10时，

具体实现类似如下代码：

Annotation document = new Annotation(str);

pipeline.annotate(document);

List<CoreMap> sentences = document.get(CoreAnnotations.SentencesAnnotation.class);

double mainSentiment = 0;

int textl = 0;

for (CoreMap sentence : sentences) {

Tree tree = sentence.get(SentimentAnnotatedTree.class);

int sentiment = RNNCoreAnnotations.getPredictedClass(tree);

if (sentiment > 0 && sentiment < 4) {

String partText = sentence.toString();

mainSentiment += sentiment \* partText.length();

textl += partText.length();

}

}

double r = 0;

if (textl > 10)

r = mainSentiment / textl;

取数据库中免费提示sentiment-平均情感绝对值最小的两个免费提示数据，再取其外3个随机免费提示数据合计5条数据随机排序构成免费提示。

否则，取5个随机免费提示数据构成首句提示。

具体实现类似如下代码：

lf：

SELECT id,content FROM freehint where isDeleted=0 order by abs(sentiment-#{r}) LIMIT 0,2;

lf2：

SELECT id,content FROM freehint where isDeleted=0 order by rand() LIMIT 0,5;

Set<Freehint> rs = new HashSet<Freehint>(); // 免费提示

if (textl > 10) {

rs.addAll(lf);

for (int i = lf2.size(); i > 0; i--) { //去重

for (Freehint f : lf) {

if (f.getId() == lf2.get(i - 1).getId()) {

lf2.remove(lf2.get(i - 1));

break;

}

}

}

rs.add(lf2.get(0));

rs.add(lf2.get(1));

rs.add(lf2.get(2));

}

else {

flist.addAll(lf2);

}

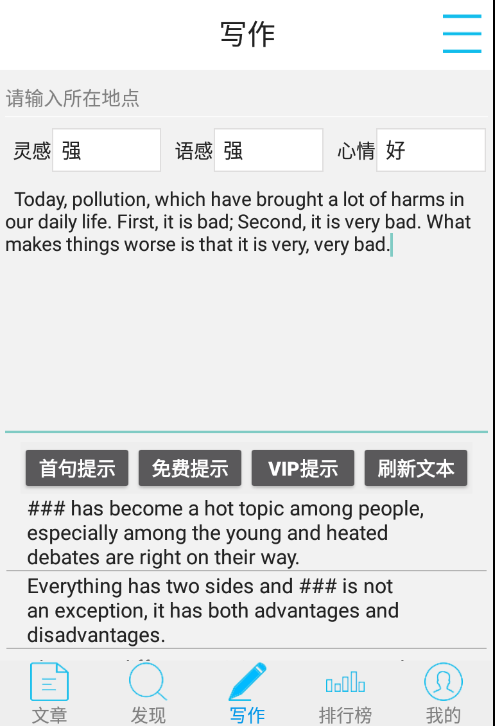


图11 免费提示

Stanford CoreNLP工具的情感分析使用一种基于深度学习的树的组合模型（Socher et al.，2013）。每个句子的二值化树的节点，特别是包括每个句子的根节点，都被赋予情感评分。【5】分析结果从0-4分别对应Very positive，Positive，Neutral，Negative,Very negative。

关于首句提示和免费提示数据中的sentiment，

对于数据的每次使用，心情为好时usep+1，心情为差时usen+1，心情为一般时usep和usen各+1。这样提示数据的sentiment值受正负情感下使用次数影响在区间（-1,1）内浮动。

每当用户数输入完一句话或者点击刷新文本将会当前文本进行拼写检查，如图12，“Haelth”明显拼写错误，点击下方正确单词纠正为“Health”，而纠正后单词前仍缺少一个空格，继续点击下方正确单词，继续纠正后没有其他错误。

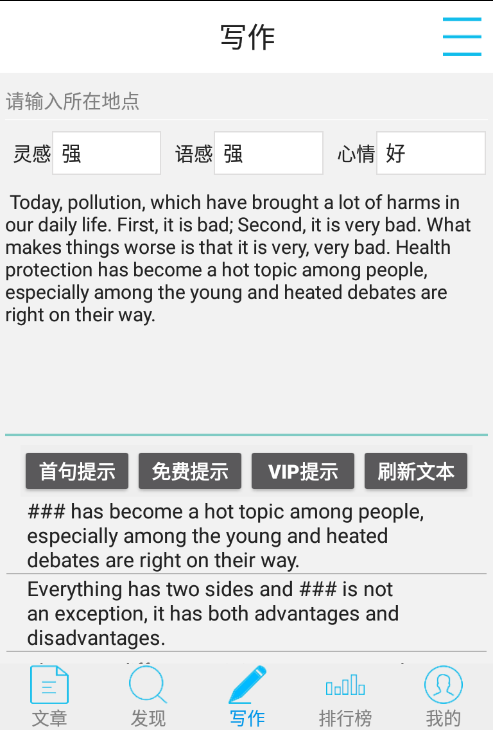
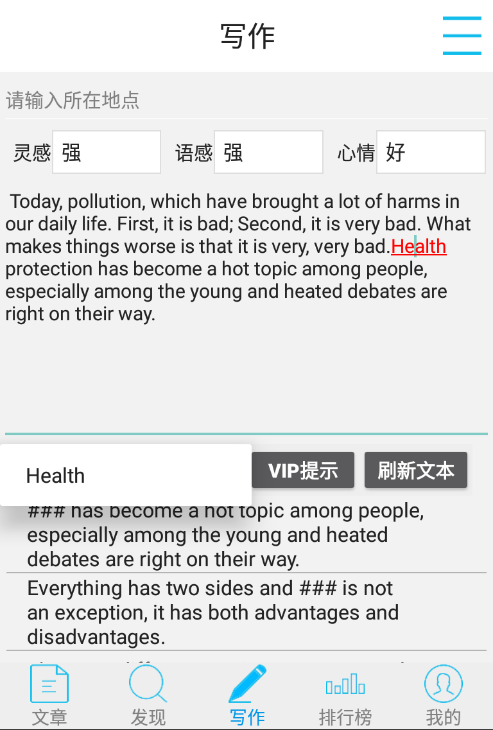
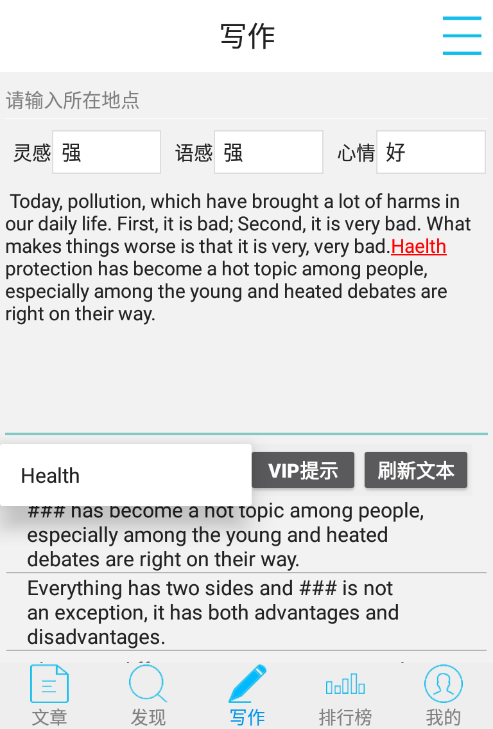


图12 拼写检查

点击VIP提示将会根据当前文章内容提取关键词，并根据预训练的GloVe词向量模型得到相关词汇并给出提示，如图13。由于前文中我写了很多“bad”，所以可以看到提示了“worse”、“unfortunately”等词。

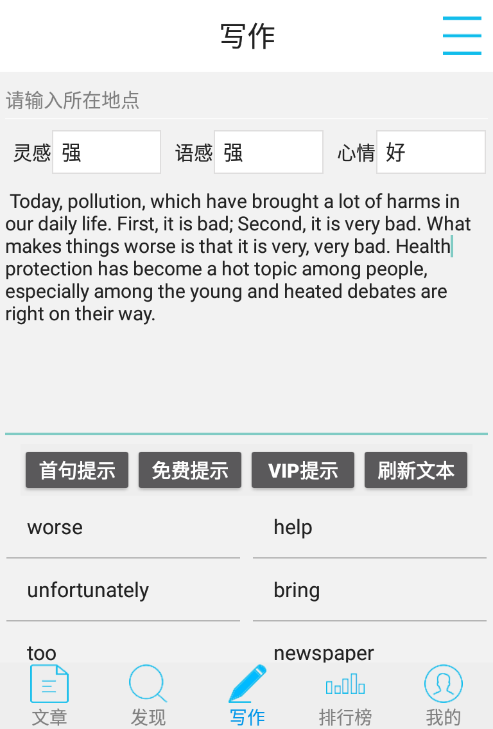


图13 VIP提示

关键词提取基于一个简单的算法：先使用一个英文停用词表stoplist筛去无意义单词（例如“yes”）再使用Stanford CoreNLP工具提取单词词根和词性，只判断名词、动词、形容词的词根，取出现次数最多的三个词根。相关词汇则是根据两个单词词向量的余弦值大小确定的，我们知道向量的余弦计算，

我们当前使用的词向量模型的维度是50，多维向量的余弦计算公式为，

将向量用double[]类型表示，则我们可以用如下代码实现：

public double test(double[] a, double[] b) {

// 计算相似度

double vector1Modulo = 0.00;// 向量1的模

double vector2Modulo = 0.00;// 向量2的模

double vectorProduct = 0.00; // 向量积

for (int i = 0; i < 50; i++) {

vector1Modulo += a[i] \* a[i];

vector2Modulo += b[i] \* b[i];

vectorProduct += a[i] \* b[i];

}

vector1Modulo = Math.sqrt(vector1Modulo);

vector2Modulo = Math.sqrt(vector2Modulo);

return (vectorProduct / (vector1Modulo \* vector2Modulo));

}

对关键词和模型中其他单词求余弦，取与第一关键词的余弦值最大的三个单词和与第二、第三关键词的余弦值最大的单词以及三个关键词本身共八个单词组成VIP提示列表。

点击右上菜单的保存，弹出保存窗口，如图14，输入标题并选择是否仅我可见后点击保存可以保存并上传。

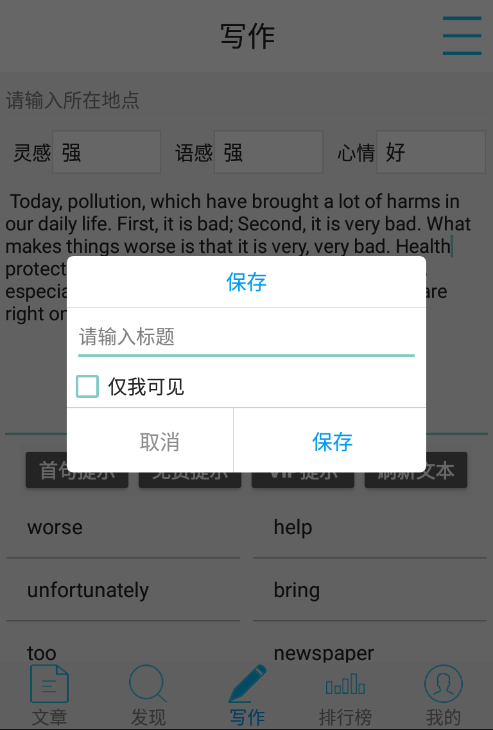


图14 保存文章

### **排行榜**

点击下边栏可以切换至排行榜页面，如图15，显示我的排名和提分以及前十名的排名和提分，该排行榜我的排名的计算按照同提分同名次且向后占位的规则，例如，排行榜上top4的用户提分和top3的用户相同，所以虽然现实在榜上为top4，但是该用户显示我的排名时仍会是3。点击用户头像处可以打开用户的个人主页。



图15 排行榜页面

### **我的**

点击下边栏可以切换至我的页面，如图16，显示昵称和积分以及我的主页等按钮。点击昵称栏可以查看我的信息。在我的信息页面点击用户头像栏可以更换头像，更换头像可以从本地上传或现在拍照，操作完成后将上传图片至服务器端存储，若成功则成功更改头像，否则将返回上传失败，这里上传的图片类型限制为BMP、JPG、JPEG、PNG和GIF，大小为1MB以下，条件不符合的将返回相应提示。其余的昵称、性别、出生日期、联系电话也可以随时修改。



图16 我的页面

点击我的主页可以打开我的主页，如图17，显示当前用户发布的文章，列表部分默认显示5个最新的文章，下拉列表刷新，上拉至底部加载更多5个文章，点击列表项可以打开文章详情页面。我的分享、我的收藏、浏览记录页面类似，分别显示当前用户分享过的文章、收藏过的文章和浏览过的文章。提分记录暂未实现。

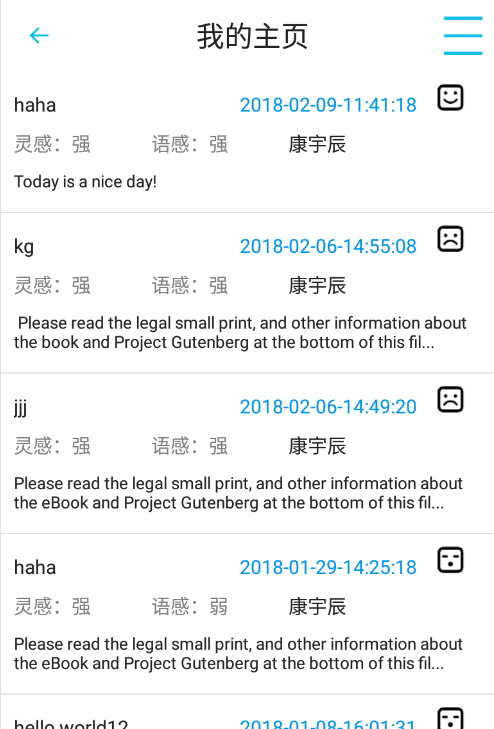


图17 我的主页页面

点击退出登录弹出确认窗口，如图18，点击确认会向服务器端确认销毁session，成功则可以退出当前登录用户，返回登录界面，否则会返回退出登录失败。



图18 退出登录

* 1. **Web端设计实现**

### **登录**

浏览器访问后台管理平台地址，默认跳转登录页面，如图19，如果未登录状态下访问其他内部地址，同样会跳转至登录界面。勾选“记住我”可以在浏览器localStorage中存储本次登陆的用户名和密码，若不勾选并登陆会删除已存储的信息。后面下拉菜单中选择“每次需要登入”则关闭浏览器后将需要重新登录，选择10天内或30天内则在对应时间内存储用户名和密码至Cookie，并在期间内无须重新登陆。



图19 登录页面

没有管理员账号可以点击注册按钮进入注册页面，如图20，按照格式填写所有信息后点击注册并等待审核通过则注册成功。信息格式错误或用户名与已有用户名重复会返回错误信息。



图20 注册页面

登录成功后进入后台管理平台，如图21，左侧菜单根据不同权限会显示不同内容，当前是总管理员权限可以看到所有功能。右上角显示已登录管理员的姓名，点击后打开下拉菜单可以进行修改密码（4.2.10）和登出（4.2.1）操作。

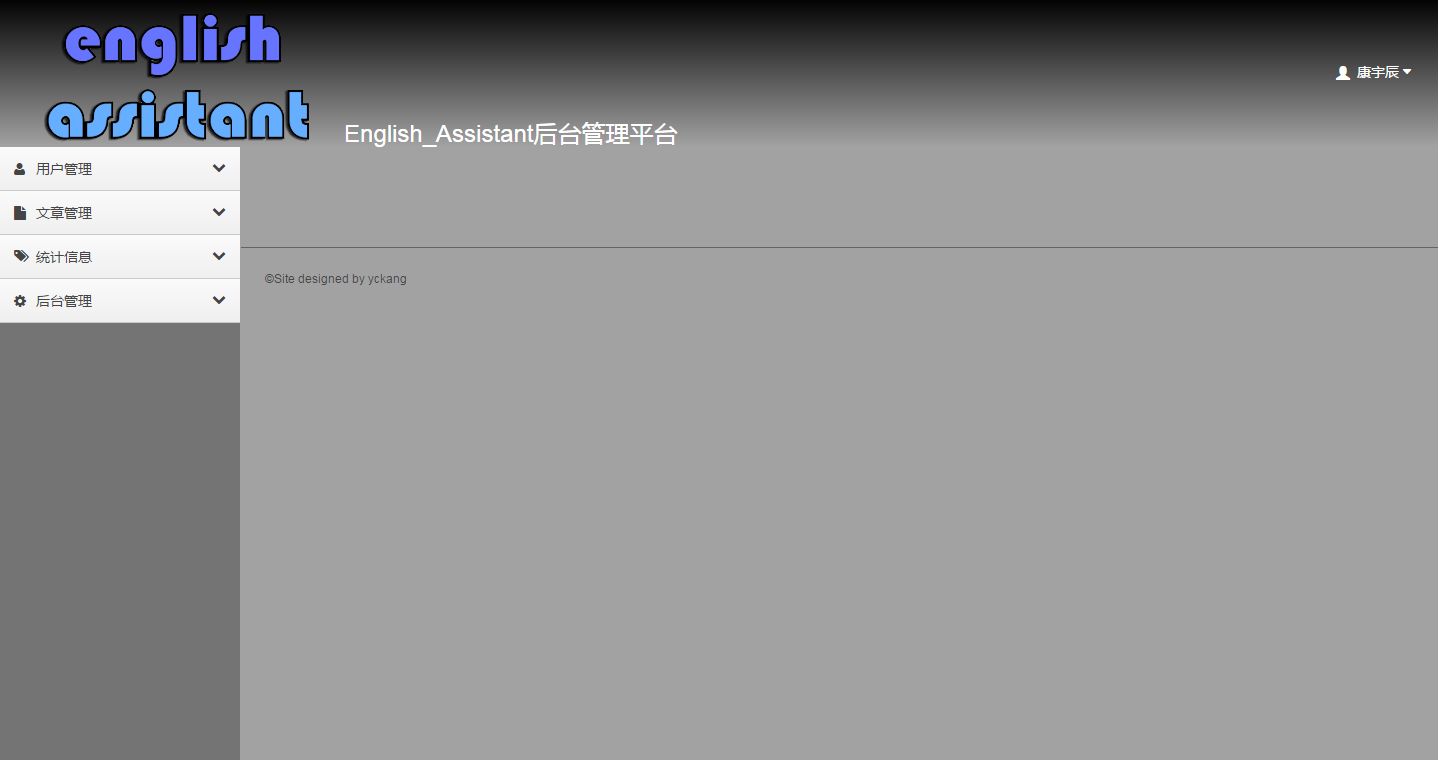


图21 首页页面

### **用户列表**

点击左侧菜单进入用户列表页面，如图22，列表显示当前系统所有用户的账号、昵称、电话号码、生日、是否VIP、是否在线等信息，输入搜索条件点击查询按钮可以进行搜索，点击列表表头可以按照对应信息进行正反排序，点击状态列的按钮可以对用户进行删除或恢复操作。

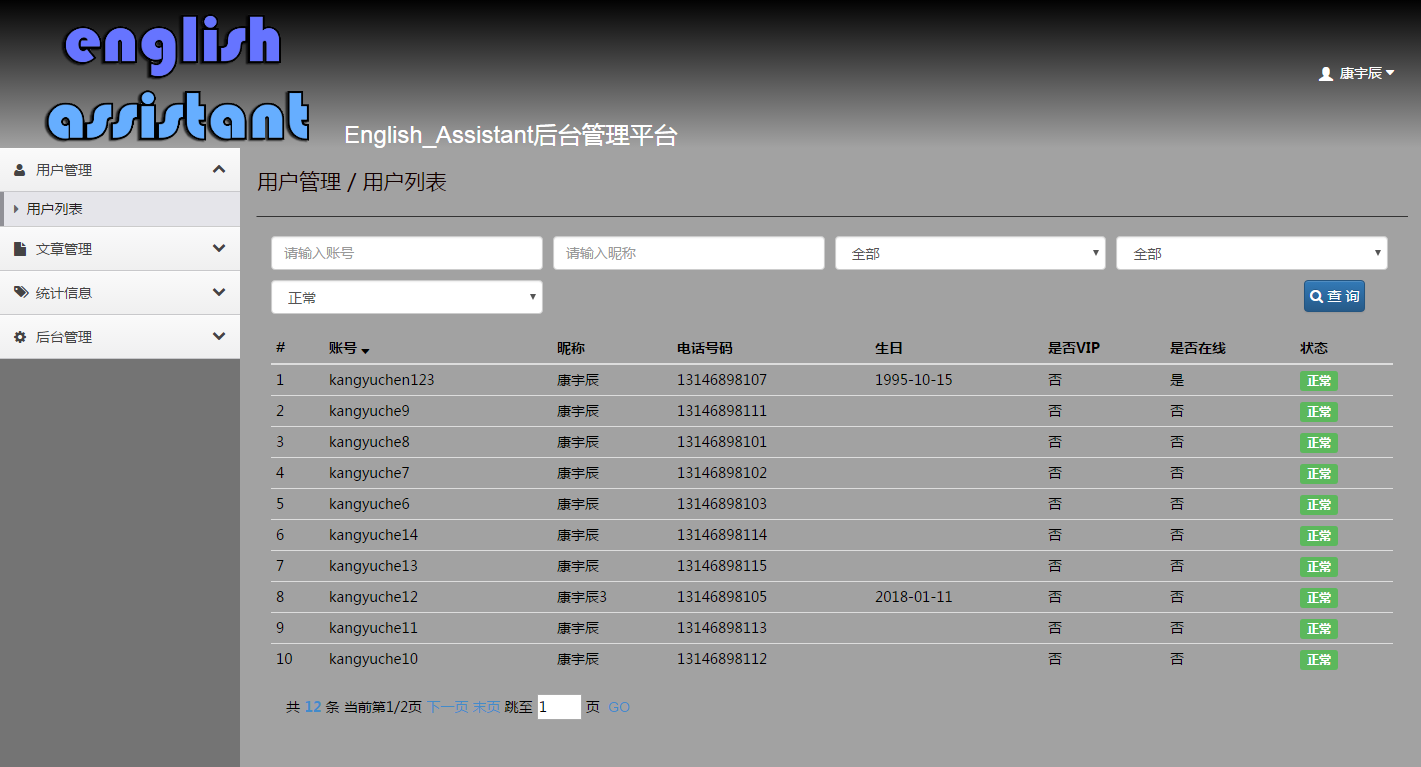


图22 用户列表页面

### **文章列表**

点击左侧菜单进入文章列表页面，如图23，列表显示当前系统所有用户文章的标题、作者、地址、点击量、收藏量、分享量、编辑时间、创建时间等信息，输入搜索条件点击查询按钮可以进行搜索，点击列表表头可以按照对应信息进行正反排序，点击状态列的按钮可以对文章进行删除或恢复操作、设为公开或私密状态，点击内容列的按钮可以查看文章正文内容。



图23 文章列表页面

点击状态列的内容按钮可以查看文章正文内容，如图24。

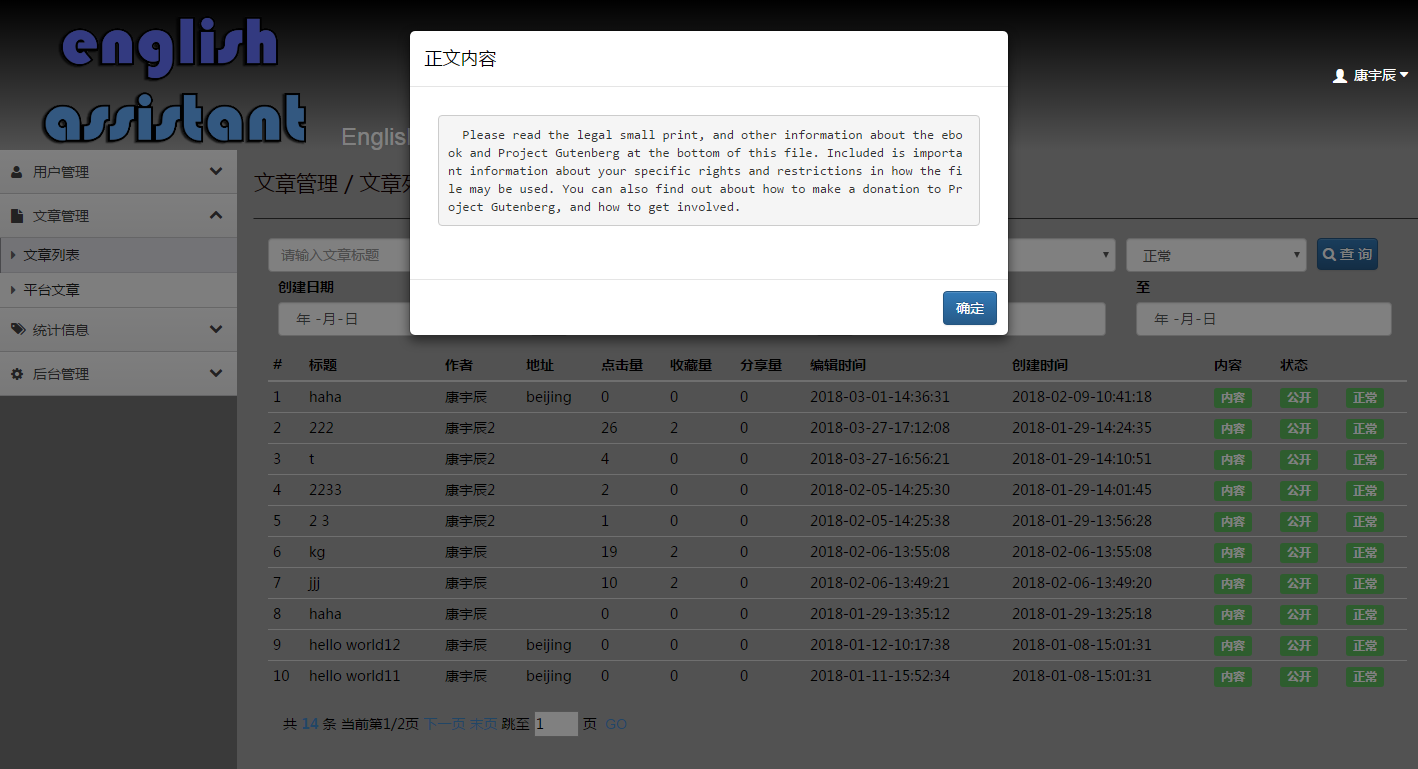


图24 文章正文内容

### **平台文章**

点击左侧菜单进入平台文章页面，如图25，列表显示当前系统所有平台推荐文章的标题、点击量、收藏量、分享量、创建时间等信息，输入搜索条件点击查询按钮可以进行搜索，点击列表表头可以按照对应信息进行正反排序，点击状态列的按钮可以对文章进行删除或恢复操作，点击内容列的按钮可以查看文章正文内容。点击发布新文章按钮可以发布新的平台推荐文章。



图25 平台文章页面

点击发布新文章按钮可以发布新的平台推荐文章，如图26，当前仅支持本地上传doc、docx和txt格式的文件，选择文件后会自动用文件名填充文章标题，也可以手动修改文章标题，点击确定上传文章。

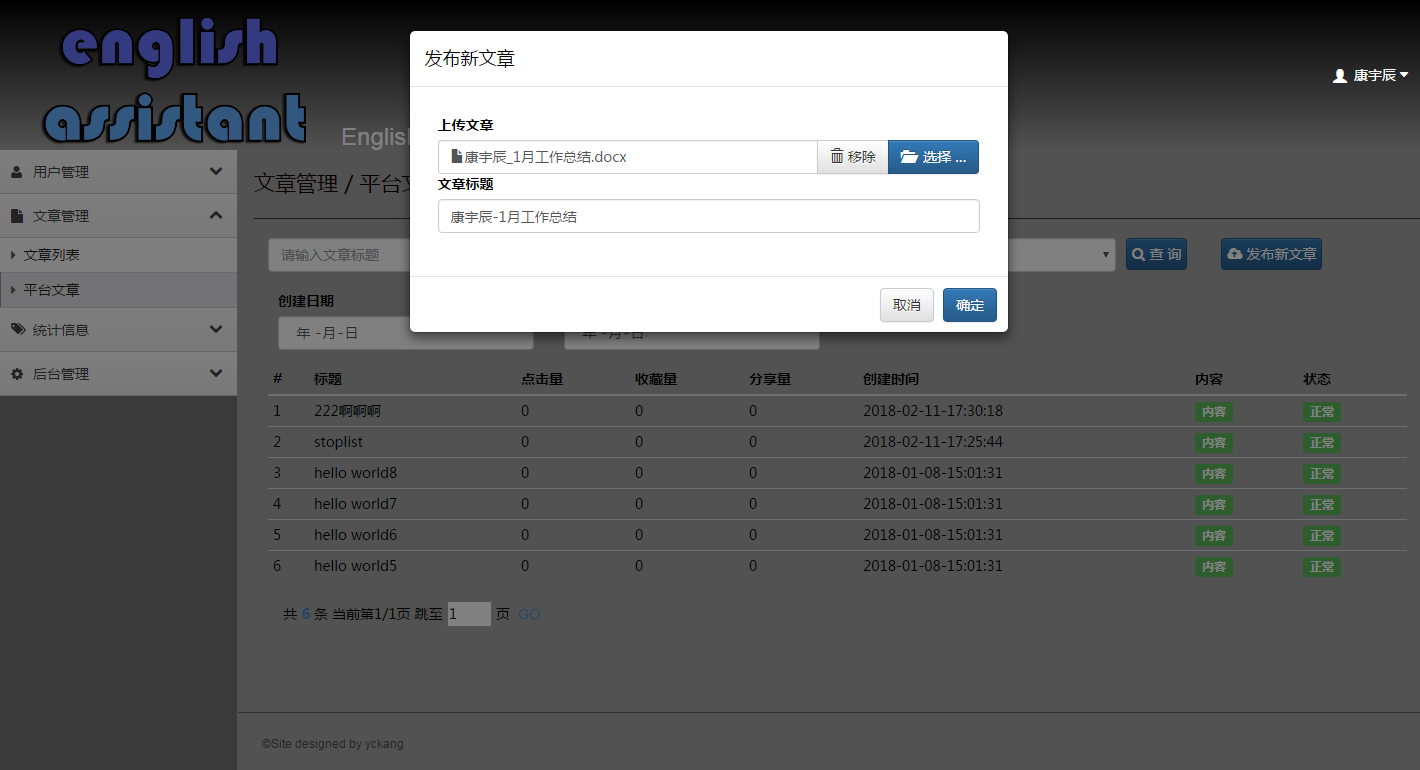


图26 发布新文章

### **在线统计**

点击左侧菜单进入在线统计页面，如图27，图表默认显示最近7天的在线统计数据，可以选择起止日期并点击查询按钮来查看更多数据。统计数据来源为每天0点启动定时任务，统计日在线人数，并存入数据库。

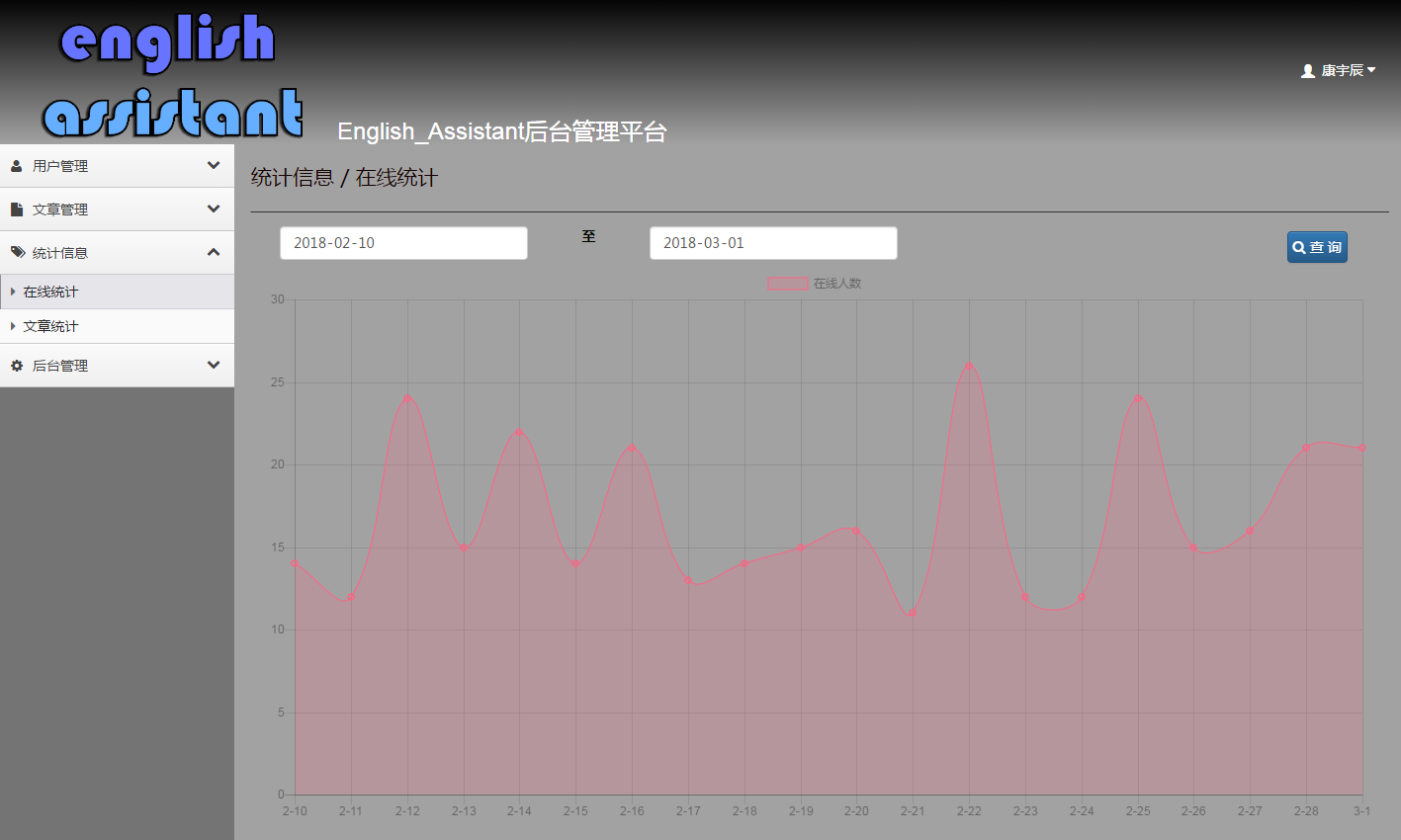


图27 在线统计页面

### **文章统计**

点击左侧菜单进入文章统计页面，如图28，图表默认显示最近7天的文章统计数据，可以选择起止日期并点击查询按钮来查看更多数据。统计数据包括首句提示使用率、免费提示使用率和vip提示使用率，来源为每天0点启动定时任务，统计累积的提示使用率、免费提示使用率和vip提示使用率，并存入数据库。

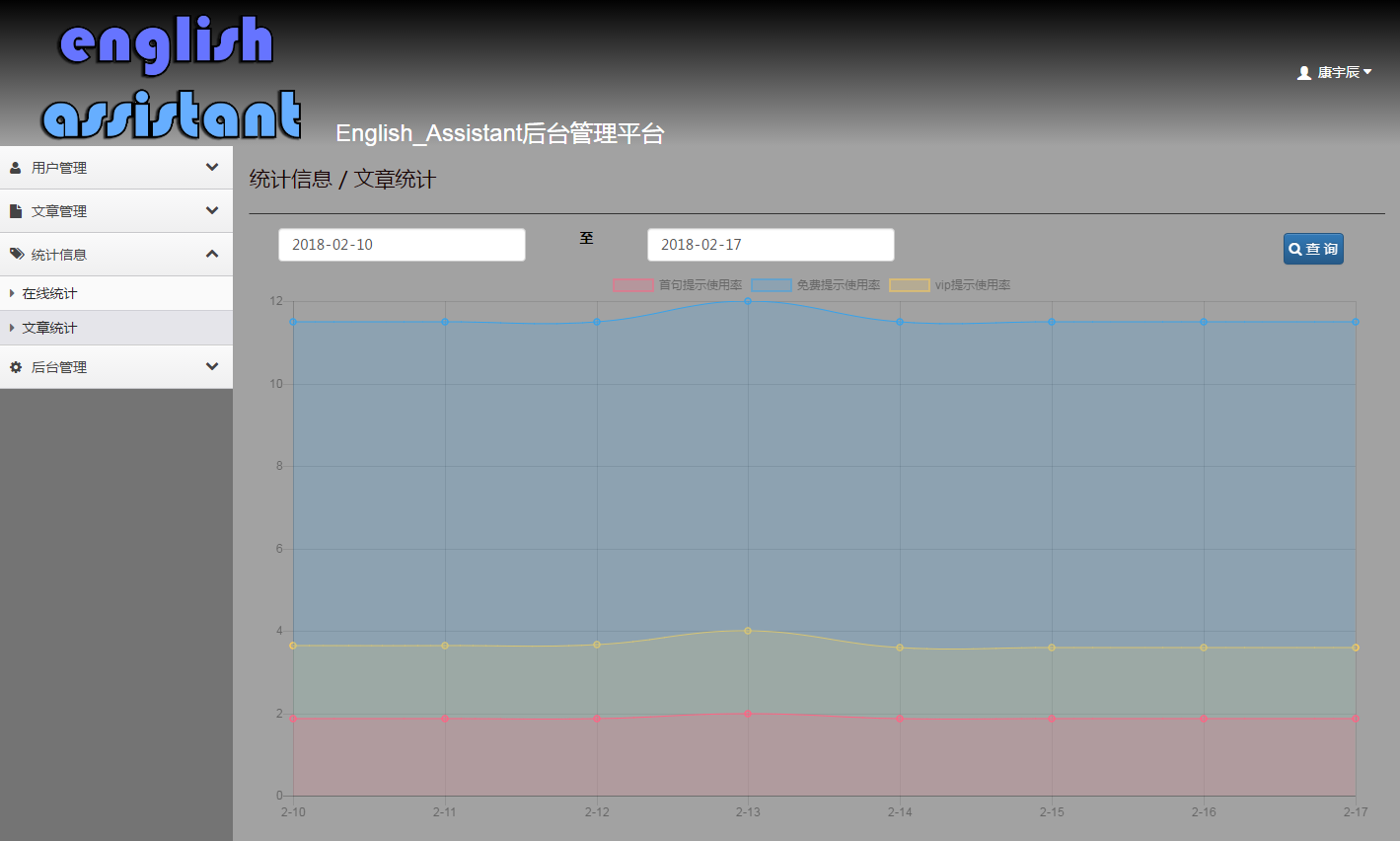


图28 文章统计页面

定时任务的实现是通过一个继承自ServletContextListener的TimerListener监听捕捉时间事件并启动统计任务。

### **日志管理**

点击左侧菜单进入日志管理页面，如图29，列表显示当前系统所有后台管理平台日志的用户名、真实姓名、IP地址、操作、结果、添加时间等信息，输入搜索条件点击查询按钮可以进行搜索，点击列表表头可以按照对应信息进行正反排序。

日志插入通过服务器启动时实例化的LogUtils调用logService完成，所有需要插入日志的操作完成时都需调用LogUtils进行日志插入。



图29 日志管理页面

### **管理员列表**

点击左侧菜单进入管理员列表页面，如图30，列表显示当前系统所有管理员的用户名、真实姓名、电话号码、权限等信息，输入搜索条件点击查询按钮可以进行搜索，点击列表表头可以按照对应信息进行正反排序，点击状态列的按钮可以对管理员进行删除或恢复操作。



图30 管理员列表页面

### **注册审核**

点击左侧菜单进入注册审核页面，如图31，列表显示当前系统所有待审核管理员的用户名、真实姓名、电话号码等信息，输入搜索条件点击查询按钮可以进行搜索，点击列表表头可以按照对应信息进行正反排序，点击操作列的按钮可以对待审核管理员进行通过或驳回操作。



图31 注册审核页面

点击操作列的按钮可以对待审核管理员进行通过操作，如图32，选择一个权限后，点击确定按钮确认通过审核。暂时只有总管理员和综合管理员两个权限，总管理员拥有全部权限，综合管理员拥有除后台管理模块之外的全部权限。

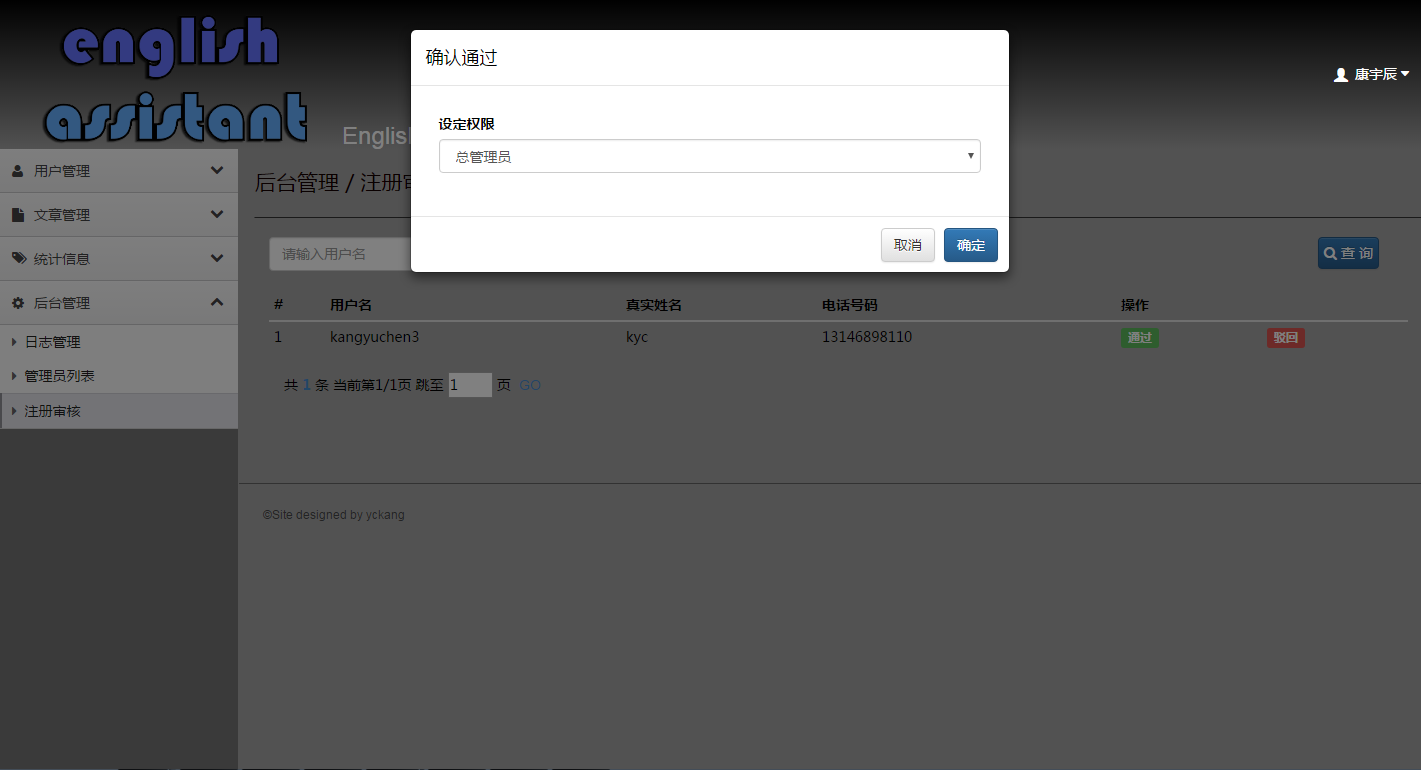


图32 确认通过审核

### **修改密码**

点击右上角下拉菜单的修改密码可以进入修改密码页面，如图33，正确输入原登录密码和正确格式的新登录密码并重复后点击保存以完成修改密码操作。

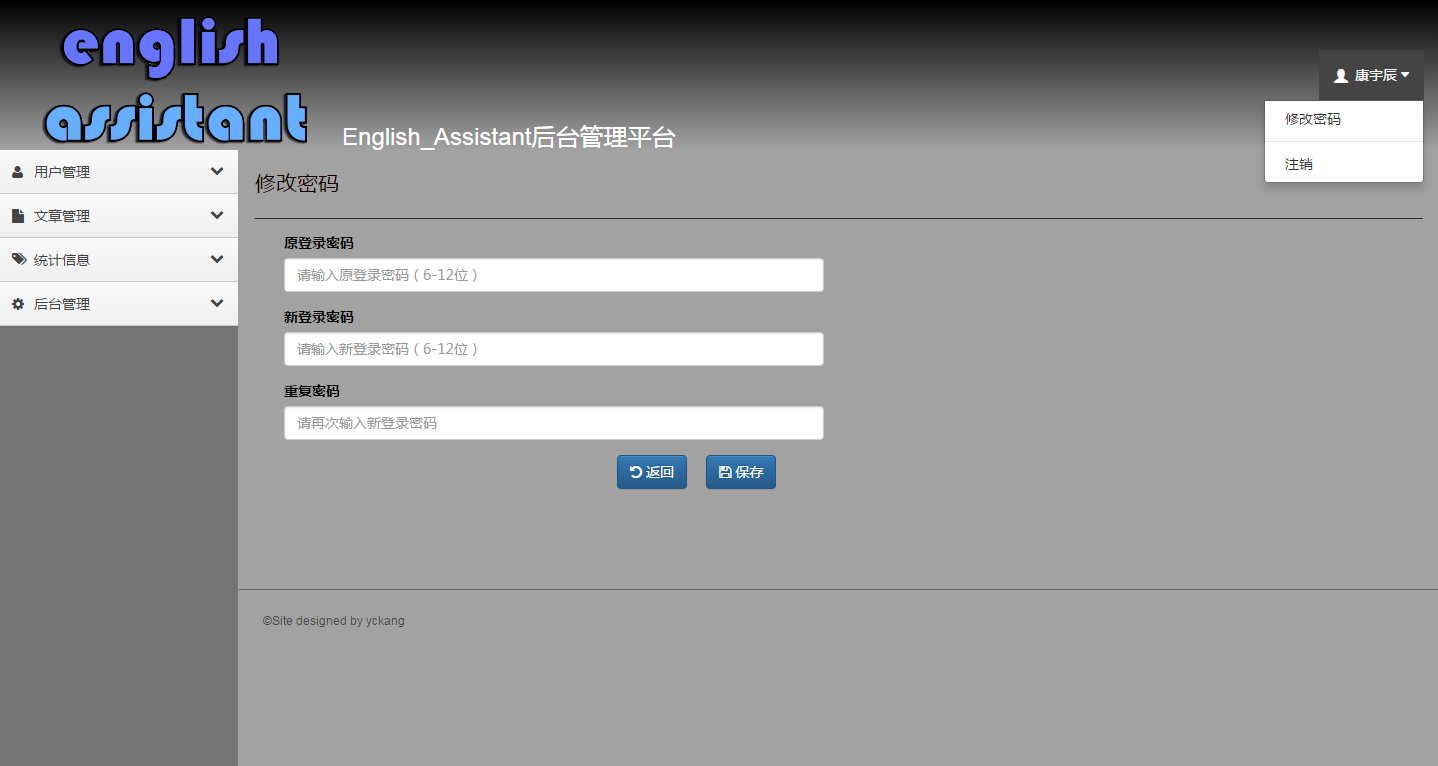


图33 修改密码页面

### **登出**

点击右上角下拉菜单的注销可以注销当前登录并回到登录页面，如果登录时选择了10天内或30天内免登录在注销后也将失效。

* 1. **本章小结**

本章我们详细介绍了本项目的Android端以及Web端设计实现过程以及其中的具体算法、实现方式、约束、展示方式等。该系统项目工程部分至此已经基本完成，接下来我们需要对系统进行测试。

**系统测试**

* 1. **测试环境搭建**

### **JDK环境配置**

1. 下载并且安装jdk-8u144-windows-x64.exe文件。

第二，右击“计算机”->“属性”，然后选择“高级系统设置”的"高级"标签页，之后选择“环境变量”，在“环境变量”选择“新建(W)……”，“变量名”文本框中输入“JAVA\_HOME”，将“变量值”输入自己电脑JDK的安装路径C:\Program Files\Java\jdk1.8.0\_144后单击“确定”保存。找到“path”的系统变量名，选中之后点击“编辑”按钮，在弹出的窗口中的“变量值”后输入“;”和自己电脑JDK中bin文件夹的路径C:\Program Files\Java\jdk1.8.0\_144\bin，单击“确定”保存。

### **Tomcat安装**

下载apache-tomcat-7.0.63-windows-x64.zip压缩包并解压即可。

### **Android模拟器Genmotion安装**

打开Genymotion官网，<https://www.genymotion.com/>,注册账号并登陆，下载with VirtualBox的版本并安装，进入软件后登陆账号（注意验证账号邮箱），点击Add按钮添加一个新的虚拟设备：Custom Phone-7.1.0-API25-768\*1280，等待添加完成即可。

### **Mysql数据库安装**

下载并执行mysql-5.5.28-winx64.msi文件，安装完成后执行english\_assistant.sql脚本导入本项目数据库，该数据库中包含一些开发过程中的测试文件。

* 1. **项目部署和准备**

### **服务器端项目部署**

修改Web项目的config.properties文件，链接至mysql数据库。打包Web项目为war包，解压至文件夹english\_assistant内，将english\_assistant文件夹拷贝至Tomcat安装目录wtpwebapps文件夹D:\apache-tomcat-7.0.63-windows-x64\apache-tomcat-7.0.63\wtpwebapps中，使用Tomcat安装目录bin目录下的shutdown.bat和startup.bat重启Tomcat。

### **Android项目测试准备**

将Android项目contact配置中的IP链接地址修改至服务器IP地址，打包发布App并安装至虚拟设备Custom Phone-7.1.0-API25-768\*1280，打开应用程序，通过所有要求权限。

### **Web项目测试准备**

打开浏览器访问<http://localhost:8080/english_assistant>，进入后台管理平台。

* 1. **系统测试**

### **Android端测试**

打开应用程序，进入登录页面，点击“没有账号，点击注册”链接处成功跳转至注册页面，正确输入各栏数据后点击注册，提示注册成功并返回登录页面。输入注册的账号和密码，点击登录按钮，进入了文章页面。

在文章页面可以看到其他用户的文章，上拉可以加载更多5篇文章，下拉可以成功刷新列表。在搜索栏中输入“hello”点击右边的搜索图标，可以看到列表只显示文章标题或内容中包含“hello”的文章。选择右上角下拉菜单中的推荐信，可以看到列表发生了变化，显示“什么都找不到！”。点击左上角图标成功进入我的主页页面，再点击左上角返回图标返回了文章页面。点击文章列表项，进入了文章详情页面。

在文章详情页面，可以看到文章全部正文内容，信息栏显示作者头像、作者昵称、创建时间、浏览量、收藏量、分享量，点击信息栏可以看到创建时间、浏览量、收藏量、分享量的位置变成了最后编辑时间、心情、灵感、语感。点击作者头像，成功进入作者我的主页页面，再点击左上角返回图标返回了文章详情页面。点击右上角下拉菜单中收藏图标，提示收藏成功。点击左上角返回图标返回了文章页面。

点击下边栏的发现图标，切换到了发现页面，同时发现图标变成了蓝色填充。在发现页面可以看到其他用户的文章和平台推荐文章，上拉可以加载更多5篇文章，下拉可以成功刷新列表并且与之前的文章不同。

点击下边栏的写作图标，切换到了写作页面，同时写作图标变成了蓝色填充。在写作页面直接弹出了请选择类别的对话框，选择陈述，可以选择心灵、灵感、语感，可以输入所在地址和文章正文。点击首句提示按钮，可以看到5条不同的首句提示信息，选择并点击一条“Today, ###, which have brought a lot of harms in our daily life. First, ###; Second, ###. What makes things worse is that ###.”可以看到内容粘贴至正文中。进行一些修改补充后，点击免费提示按钮，可以看到5条不同的免费提示信息，选择并点击一条“### has become a hot topic among people, especially among the young and heated debates are right on their way.”可以看到内容粘贴至正文中。进行一些修改补充后，点击刷新文本按钮，可以看到手动输入的单词“Haelth”变成红色，有明显拼写错误，点击该单词，下方展开下拉菜单，点击正确单词纠正为“Health”。再次点击刷新文本按钮，可以看到“Health”又变成红色，点击该单词，下方展开下拉菜单，点击正确单词后发现该单词与前一个句子之间添加了一个空格。再次点击刷新文本按钮，没有出现更多错误。点击VIP提示按钮，可以看到8个提示单词，由于前文中我写了很多“bad”，所以可以看到提示了“worse”、“unfortunately”等词。点击右上角下拉菜单的保存图标，展示保存窗口，输入标题，未勾选仅我可见，点击保存，提示保存成功，并自动新建文章弹出了请选择类别的对话框。输入新的测试内容，点击右上角下拉菜单的删除图标，弹出了删除确认提示，点击删除，提示删除成功，并自动新建文章弹出了请选择类别的对话框。点击下边栏的文章图标，切换至文章页面，可以看到我刚刚保存的文章。

点击下边栏的排行榜图标，切换到了排行榜页面，同时排行榜图标变成了蓝色填充。可以看到我的排名是7提分为0，因为排行榜只有前6个用户有提分，所以后面的其他用户并列第7。列表显示提分前十的用户及其提分，下拉可以成功刷新列表。点击用户头像，成功进入用户我的主页页面，再点击左上角返回图标返回了排行榜页面。

点击下边栏的我的图标，切换到了我的页面，同时我的图标变成了蓝色填充。可以看到昵称栏、我的主页、我的分享、我的收藏、浏览记录等按钮。点击昵称栏进入个人信息页面，显示用户头像、昵称、性别、出生日期、联系电话等信息。点击用户头像，弹出选择附件对话框，选择现在拍照，打开摄像头拍摄一张照片，确认后显示修改成功，可以看到头像已经变成了刚刚拍摄的照片。分别点击昵称、性别、出生日期、联系电话均可以弹出窗口并修改对应内容。点击左上角返回图标返回了我的页面，分别点击我的主页、我的分享、我的收藏、浏览记录均可以打开对应页面，展示对应内容。点击下方退出登录按钮弹出提示确认窗口，确认后成功退出当前登录并返回登录页面。

整个测试过程运行正常，完成了Android移动端既定的功能。

### **Web端测试**

进入后台管理平台, 默认进入登录页面，输入已有的总管理员账号和密码，勾选记住我选项，选择10天内免登录选项，点击登录（阻止浏览器记住密码），成功登录并跳转至首页。关闭并重启浏览器，再次访问后台管理平台，直接进入了首页，表明免登录功能有效。点击右上角姓名处展开下拉菜单，点击注销，成功注销并回到登录页面，发现用户名和密码被自动填充，取消勾选记住我选项，再次登录并注销，发现用户名和密码不再被自动填充，表明记住我功能有效。

再次登录，点击左侧菜单的用户管理下的用户列表，进入用户列表页面，列表显示当前系统所有用户的账号、昵称、电话号码、生日、是否VIP、是否在线等信息。在“请输入账号”处输入“123”点击查询按钮进行搜索，可以看到列表只显示账号包含“123”的用户。点击列表表头“账号”，可以看到列表按照账号升序进行排序，再次点击，可以看到列表按照账号降序进行排序，再次点击恢复至默认排序。点击一个用户状态列的绿色“正常”按钮，可以看到按钮变为红色“已删除”表示已删除该用户，再次点击，按钮恢复绿色“正常”表示该用户恢复正常。

点击左侧菜单的文章管理下的文章列表，进入文章列表页面，列表显示当前系统所有用户文章的标题、作者、地址、点击量、收藏量、分享量、编辑时间、创建时间等信息。在“请输入文章标题”处输入“hello”点击查询按钮进行搜索，可以看到列表只显示文章标题包含“hello”的文章。点击列表表头“标题”，可以看到列表按照文章标题升序进行排序，再次点击，可以看到列表按照文章标题降序进行排序，再次点击恢复至默认排序。点击一个文章状态列的绿色“正常”按钮，可以看到按钮变为红色“已删除”表示已删除该文章，再次点击，按钮恢复绿色“正常”表示该文章恢复正常，点击状态列的绿色“公开”按钮，可以看到按钮变为黄色“私有”表示该文章变成仅作者可见的私有状态，再次点击，按钮恢复绿色“公开”表示该文章恢复所有人可见的公开状态。点击一个文章内容列的绿色“内容”，可以看到弹出了显示正文内容的窗口，点击确定关闭窗口。

点击左侧菜单的文章管理下的平台文章，进入平台文章页面，列表显示当前系统所有平台推荐文章的标题、点击量、收藏量、分享量、创建时间等信息。在“请输入文章标题”处输入“hello”点击查询按钮进行搜索，可以看到列表只显示文章标题包含“hello”的平台推荐文章。点击列表表头“标题”，可以看到列表按照文章标题升序进行排序，再次点击，可以看到列表按照文章标题降序进行排序，再次点击恢复至默认排序。点击一个文章状态列的绿色“正常”按钮，可以看到按钮变为红色“已删除”表示已删除该文章，再次点击，按钮恢复绿色“正常”表示该文章恢复正常。点击一个文章内容列的绿色“内容”，可以看到弹出了显示正文内容的窗口，点击确定关闭窗口。点击发布新文章按钮，弹出了发布新文章窗口，点击选择按钮，选择本地文件“康宇辰-1月工作总结.docx”，下边的文章标题栏被自动填充了“康宇辰-1月工作总结”，点击确定按钮，可以看到窗口内绿色进度条至100%并自动关闭了窗口，列表中已经可以看到新上传的文章“康宇辰-1月工作总结”。

点击左侧菜单的统计信息下的在线统计，进入在线统计页面，可以看到起止日期默认为8天前到昨天，修改起止日期为2-10至3-1，点击查询，可以看到页面上显示了这段时间内的在线人数曲线图。

点击左侧菜单的统计信息下的文章统计，进入文章统计页面，可以看到起止日期默认为8天前到昨天，修改起止日期为2-10至2-17，点击查询，可以看到页面上显示了这段时间内的首句提示使用率、免费提示使用率、VIP提示使用率曲线图。点击上方图例“首句提示使用率”，可以看到图例处字样被划横线，对应曲线图不再显示，再次点击，图例处字样恢复，曲线图恢复显示。

点击左侧菜单的后台管理下的日志管理，进入日志管理页面，列表显示当前系统所有后台管理平台日志的用户名、真实姓名、IP地址、操作、结果、添加时间等信息。在“请输入用户名”处输入“kangyuchen123”点击查询按钮进行搜索，可以看到列表只显示用户名包含“kangyuchen123”的日志记录。点击列表表头“用户名”，可以看到列表按照用户名升序进行排序，再次点击，可以看到列表按照用户名降序进行排序，再次点击恢复至默认排序。

点击左侧菜单的后台管理下的管理员列表，进入管理员列表页面，列表显示当前系统所有管理员的用户名、真实姓名、电话号码、权限等信息。在“请输入用户名”处输入“kangyuchen123”点击查询按钮进行搜索，可以看到列表只显示用户名包含“kangyuchen123”的管理员。点击列表表头“用户名”，可以看到列表按照用户名升序进行排序，再次点击，可以看到列表按照用户名降序进行排序，再次点击恢复至默认排序。

注销当前登录，在登录页面点击注册按钮，进入注册页面，在用户名处输入“kangyuchen”，切换至其他输入栏，发现用户名处显示红色“该用户名已存在”，修改为“kangyuchen3”，切换至其他输入栏，红色提示消失，继续完成其他输入后点击注册，返回登录页面，并弹出提示窗口显示“您仍需等待审核通过才可以登录”。以同样方法再次注册“kangyuchen4”。再次登录已有的总管理员账号。

点击左侧菜单的后台管理下的注册审核，进入注册审核页面，列表显示当前系统所有待审核管理员的用户名、真实姓名、电话号码等信息。可以看到刚刚注册的两个账号。在“请输入用户名”处输入“kangyuchen”点击查询按钮进行搜索，可以看到列表只显示用户名包含“kangyuchen”的待审核管理员。点击列表表头“用户名”，可以看到列表按照用户名升序进行排序，再次点击，可以看到列表按照用户名降序进行排序，再次点击恢复至默认排序。点击一个待审核管理员“kangyuchen3”操作列的绿色“通过”按钮，弹出确认通过窗口，选择“综合管理员”权限，点击确定，显示操作成功，列表中不再显示待审核管理员“kangyuchen3”。点击另一个待审核管理员“kangyuchen4”操作列的红色“驳回”按钮，弹出确认通过窗口，点击确定，显示操作成功，列表中不再显示待审核管理员“kangyuchen4”。再次注销，尝试登陆“kangyuchen4”提示“用户名不存在”。尝试登陆“kangyuchen3”，登录成功，并且左侧菜单没有后台管理菜单，尝试在浏览器地址栏输入后台管理下注册审核的地址，跳转至首页并显示“抱歉，您没有此权限”。

点击右上角姓名处展开下拉菜单，点击修改密码，打开修改密码页面。依次输入原密码，新密码和重复密码后点击保存按钮，提示保存成功。注销登录，使用旧密码登录，提示“密码错误!”,使用新密码登录，登录成功。

整个测试过程运行正常，完成了Web后台管理端既定的功能。

* 1. **本章小结**

本章我们对英语写作智能辅助系统的Android移动端和Web后台管理端进行了功能测试，测试结果为系统运行正常，完成了系统既定的功能。

**结论**

英语写作智能辅助系统以软件工程基本原理为指导，依托SSM框架等框架，利用Stanford CoreNLP等集成工具实现了提供英语写作时的查错、纠正、智能提示等辅助功能，帮助用户更好地进行英语写作的目标。虽然仍存在许多不足，但是从已实现的功能来看已经符合基本需求，并且管理功能较为全面，界面简洁美观，操作友好，可拓展性强，具有实际使用价值。

本论文较为系统地介绍了包括Android移动端和Web后台管理端在内的英语写作智能辅助系统的设计方法和实现过程，从需求分析、设计工具和技术的说明，到总体功能设计、数据库搭建、在到最后的功能实现都进行了较为详细的论述，并且图文并茂，对系统运行情况有较为清晰地展示，完成了“定制整体解决方案，完成整体平台设计与开发以及核心业务层对自然语言（英语）的语法分析，分词、词性标注、专名识别等功能，并通过依存句法分析、词向量、DNN语言模型、词义相似度、短文相似度、评论客观抽取、情感倾向分析等方法，完成英语写作中的实时智能提示功能”的总体需求。

**参考文献**

1. 王天笑. 自然语言处理的现状研究与未来发展初探[J]. 中国科技纵横, 2017(2).
2. 杨扬, 侯红, 郝克刚. 基于容器的IoC控制反转模式的研究[J]. 计算机应用与软件, 2009, 26(4):17-19.
3. 张书贵, 李雪, 陈丽. 在Spring面向切面编程及其应用研究[J]. 电脑编程技巧与维护, 2013(6):8-9.
4. 薛峰, 梁锋, 徐书勋,等. 基于Spring MVC框架的Web研究与应用[J]. 合肥工业大学学报(自然科学版), 2012, 35(3):337-340.
5. Manning C D, Surdeanu M, Bauer J, et al. The Stanford CoreNLP Natural Language Processing Toolkit[C]. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations. 2014: 55-60.
6. Pennington J, Socher R, Manning C. Glove: Global Vectors for Word Representation[C]. Conference on Empirical Methods in Natural Language Processing, 2014:1532-1543.
7. 郁可人, 傅云斌, 董启文. 基于神经网络语言模型的分布式词向量研究进展[J]. 华东师范大学学报(自然科学版), 2017(5):52-65.

**致谢**

感谢指导老师于学军老师的关心和指导，于老师对待学生认真严格的作风使学生受益良多。

感谢责任老师姚范老师在毕业设计工作中给予的帮助。

特别感谢刘轩老师在项目开发及论文写作工作上的大量帮助指导，在项目初期起到了领路人的作用。

感谢这段时间里同事、同学和家人们的关心和帮助。