文件编号：[TEAMNAME]-SWC2018-[TEAMNUMBER]

受控状态：■受控 □非受控

保密级别：□公司级 □部门级 ■项目级 □普通级

采纳标准：CMMI DEV V1.2



[项目LOGO]

[项目名称]

**[Project Name]**

项目开发文档

**Version [Number]**

[YYYY.MM.DD]

**Written by [Team Name]**

[Team LOGO]

**All Rights Reserved**

目录

[1 引言 1](#_Toc526945116)

[1.1 编写目的 1](#_Toc526945117)

[1.2 项目概述 1](#_Toc526945118)

[1.3 项目背景 1](#_Toc526945119)

[1.4 术语和缩略语 1](#_Toc526945120)

[1.5 参考资料 1](#_Toc526945121)

[1.6 项目定位 2](#_Toc526945122)

[1.6.1 应用场景 2](#_Toc526945123)

[1.6.2 目标人群 2](#_Toc526945124)

[1.7 项目目标 2](#_Toc526945125)

[1.8 项目价值 2](#_Toc526945126)

[1.9 创新点 2](#_Toc526945127)

[1.9.1 功能性创新 2](#_Toc526945128)

[1.9.2 非功能性创新 2](#_Toc526945129)

[1.10 解决思路 2](#_Toc526945130)

[1.11 系统亮点 2](#_Toc526945131)

[1.11.1 功能性亮点 2](#_Toc526945132)

[1.11.2 非功能性亮点 2](#_Toc526945133)

[2 开发计划 2](#_Toc526945134)

[2.1 最终呈现形式 2](#_Toc526945135)

[2.2 主要功能描述 2](#_Toc526945136)

[2.3 运行环境 2](#_Toc526945137)

[2.4 验收标准 2](#_Toc526945138)

[2.5 关键问题 3](#_Toc526945139)

[2.6 进度安排 3](#_Toc526945140)

[2.7 开发预算 3](#_Toc526945141)

[3 可行性分析 3](#_Toc526945142)

[3.1 市场可行性分析 3](#_Toc526945143)

[3.1.1 与现行软件对比 3](#_Toc526945144)

[3.2 技术可行性分析 3](#_Toc526945145)

[3.2.1 功能简述 3](#_Toc526945146)

[3.2.2 技术要素 3](#_Toc526945147)

[3.3 资源可行性分析 3](#_Toc526945148)

[4 需求分析 3](#_Toc526945149)

[4.1 数据描述 3](#_Toc526945150)

[4.1.1 静态数据 3](#_Toc526945151)

[4.1.2 动态数据 3](#_Toc526945152)

[4.1.3 数据词典 3](#_Toc526945153)

[4.1.4 数据采集 3](#_Toc526945154)

[4.2 功能需求 3](#_Toc526945155)

[4.2.1 功能模块结构图 4](#_Toc526945156)

[4.2.2 核心功能模块描述 4](#_Toc526945157)

[4.3 非功能性需求 4](#_Toc526945158)

[4.3.1 扩展性和可维护性 4](#_Toc526945159)

[4.3.2 易用性 4](#_Toc526945160)

[4.4 性能需求 4](#_Toc526945161)

[4.4.1 时间特性 4](#_Toc526945162)

[4.4.2 适应性 4](#_Toc526945163)

[4.5 运行需求 4](#_Toc526945164)

[4.5.1 用户界面 4](#_Toc526945165)

[4.5.2 硬件接口 4](#_Toc526945166)

[4.5.3 软件接口 4](#_Toc526945167)

[4.5.4 故障处理 4](#_Toc526945168)

[4.6 用例图 4](#_Toc526945169)

[4.7 核心模块用例规约 5](#_Toc526945170)

[4.8 其他需求 6](#_Toc526945171)

[5 概要设计 6](#_Toc526945172)

[5.1 处理流程 6](#_Toc526945173)

[5.2 总体结构和模块设计 6](#_Toc526945174)

[5.3 功能分配 6](#_Toc526945175)

[5.4 接口设计 6](#_Toc526945176)

[5.4.1 外部接口 6](#_Toc526945177)

[5.4.2 内部接口 6](#_Toc526945178)

[5.4.3 用户界面设计 6](#_Toc526945179)

[5.5 数据结构设计 6](#_Toc526945180)

[5.5.1 逻辑结构设计 6](#_Toc526945181)

[5.5.2 物理结构设计 6](#_Toc526945182)

[5.5.3 数据结构与程序的关系 6](#_Toc526945183)

[5.6 运行设计 6](#_Toc526945184)

[5.6.1 运行模块关系 6](#_Toc526945185)

[5.6.2 运行控制 6](#_Toc526945186)

[5.6.3 运行时间 6](#_Toc526945187)

[5.7 错误/异常处理设计 6](#_Toc526945188)

[5.7.1 错误/异常输出信息 7](#_Toc526945189)

[5.7.2 错误/异常处理对策 7](#_Toc526945190)

[5.8 系统配置策略 7](#_Toc526945191)

[5.9 系统部署方案 7](#_Toc526945192)

[5.10 代码规范 7](#_Toc526945193)

[5.11 其他相关技术与方案 7](#_Toc526945194)

[6 数据库设计 7](#_Toc526945195)

[7 详细设计 7](#_Toc526945196)

[7.1 \*\*功能模块 7](#_Toc526945197)

[7.1.1 功能描述 7](#_Toc526945198)

[7.1.2 性能描述 7](#_Toc526945199)

[7.1.3 输入 7](#_Toc526945200)

[7.1.4 输出 7](#_Toc526945201)

[7.1.5 算法 7](#_Toc526945202)

[7.1.6 程序逻辑 7](#_Toc526945203)

[7.1.7 接口 7](#_Toc526945204)

[7.1.8 存储分配 7](#_Toc526945205)

[7.1.9 限制条件 7](#_Toc526945206)

记录更改历史

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **更改原因** | **版本** | **作者** | **更改日期** | **备 注** |
| 1 | 创建 | 1.0.0 | 队员D | 2018-11-10 | 创建项目开发文档 |
| 2 | 更新 | 1.0.1 | 队员D | 2018-11-12 | 文档的第一第二部分撰写 |
| 3 | 更新 | 1.0.2 | 队员B | 2018-11-13 | 文档的第四部分撰写 |
| 4 | 更新 | 1.0.3 | 队员D | 2018-11-14 | 文档的第三部分撰写 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# 引言

## 编写目的

此开发文档的编写目的是对Temage图文排版项目的各方面做详细全面的介绍、分析与总结。从我们期望解决的问题出发，到项目的目标与定位，再到项目的构想与搭建方案，包含了详细的市场分析、功能介绍、和设计细节。通过此开发文档，你可以更深入地了解项目的理念和项目。

同时我们也希望此文档能够有助于规范项目的管理，保证项目质量，为开发、使用和维护提供有效的指导、解惑和帮助。

## 项目概述

Temage图文排版项目是一个运用机器学习技术实现自动图文排版的web应用。用户给定文字和图片，即可根据文章的主题，在线智能排版文字与图片，选择合适的格式或装饰组件，生成一篇优美悦目的文章。能够极大地节省文章从写作到发布的时间，提高用户的效率和生产力。

项目针对当下互联网的进一步发展为大众生活、媒体传播等方面带来的新变化，具有很好的应用前景和广大的目标用户群体。

Temage注重用户的体验，旨在为用户提供优美友好的界面和优质流畅的服务。

## 项目背景

当今是一个社交媒体的时代，得益于技术的日新月异，信息传播从未如此迅捷。自Facebook、Twitter始，再到微信、微博、今日头条等国人常用的社交软件，每天在人手中浏览、转发、评论的文章不可计数。根据统计，早在2016年，微信公众号账号数量就已近2000万，而活跃用户数量更是达到了7.62亿，如今这些数字还在增长。在这样的大环境下，自媒体等行业的发展也得到了极大的推动。同时,即便是政府、企业、社会团体、组织等也纷纷建立起自己的账号，通过文章向公众传达信息。

不管是生活中还是工作中，越来越多的人需要撰写、发布文章。而其中一个重要环节就是排版，优秀的排版往往能够吸引大众眼球，是文章锦上添花的部分。对此我们主要针对在公众号、微博等平台发布长文的用户进行分析与调研，发现存在以下几点问题：一是排版需要耗费用户不少的时间和精力，二是没有设计的基础的用户，很难做出优秀的版面设计，三是辛苦做出来的排版可能会与文章主题不搭配。

现在用户常用的排版软件，如秀米等，需要用户手动选取样式和装饰组件，手动插入图片。用户仍需要大量的拖拽与调整才能得到满意的排版。目前市场上也有一些自动排版软件，但都只是针对单一的文字，仅有调整缩进、清楚无效空行、替换无效符号等单一的功能，并不能完全解决用户的问题

针对上述问题，结合当下大热的深度学习技术，我们构想出Temage图文排版项目，期望它能根据输入，自动为用户提供专业且合适的版面设计，生成优质的图文排版，为用户节省90%以上的排版时间。同时降低版面设计的门槛，使没有美工基础的用户也能发布精品文章，为大众服务，还能降低自媒体创业的成本。除此之外，Temage还需要对同时代的版面设计进行学习，紧跟时代设计的风向与潮流。

## 术语和缩略语

1. Temage：Temage是本项目的名称，是一个在线的自动排版web应用，旨在为用户发布文章提供快捷优质服务。
2. Tensorflow： Tensorflow是一个采用数据流图(data flow graphs),用于数值计算的开源软件库。
3. Keras： Keras是一个高层神经网络API，Keras由纯Python编写而成并基于Tensorflow、Theano以及CNTK后端。
4. Tornado: Tornado 是一种 Web 服务器软件的开源版本。Tornado 和现在的主流 Web 服务 器框架（包括大多数 Python 的框架）有着明显的区别：它是非阻塞式服务器，而且速度相 当快。
5. Django: Django 是 Python 编程语言驱动的一个开源模型-视图-控制器（MVC）风格的 Web 应用程序框架。

## 参考资料

[1] TalkingData. 2016年自媒体行业洞察报告[R]

[2] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.

[3] Peters M E, Neumann M, Iyyer M, et al. Deep contextualized word representations[J]. arXiv preprint arXiv:1802.05365, 2018.

[4] Lai S, Xu L, Liu K, et al. Recurrent Convolutional Neural Networks for Text Classification[C]//AAAI. 2015, 333: 2267-2273.

## 项目定位

### 应用场景

无论是在工作中还是在日常生活中，当用户想要缩短图文文章发布进程、节省时间精力成本时，当没有设计基础的用户想要做出精美排版时，当用户不知道如何贴合文章主题设计版面时，Temage都可以为用户提供专业优质、量身定制的排版服务和强大的管理服务。用户仅需要上传图片和文字，90%以上的排版工作，Temage来为用户完成。

### 目标人群

此项目适用于有文章排版需求的人，主要包括：

（1）希望节约成本的自媒体个人或团队

（2）希望快速成长的自媒体起步者

（3）想要快捷地获取时尚精美版式设计的普通大众。Temage将在提供优质排版服务的同时，为用户制定专业的个性化服务，同时打造一个新而全的版式交流社区，满足专业与非专业人员的设计需求

## 项目目标

实现一个功能完整，各功能之间联动、协作的自动排版系统。

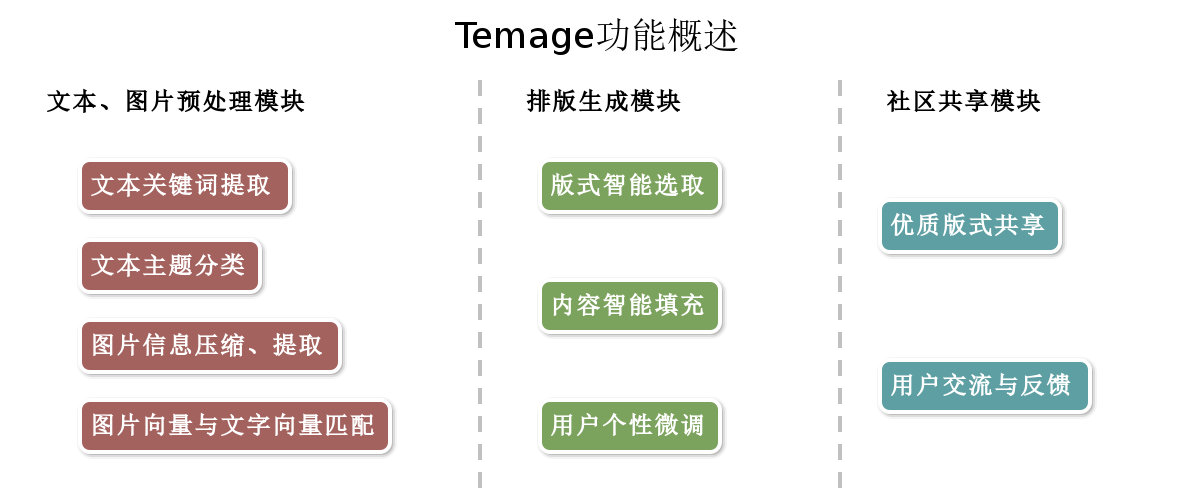


图 1.1 Temage功能概述

首先，能够根据用户输入的文章，确定文章的主题，智能地为用户挑选符合该主题的版式设计。然后，能够对从文本、图片中提取的关键信息进行分析，自动将图片嵌入至文章合适的位置，去掉多余的空行、空格，实现初步的图文排版。之后，结合初排版和挑选的版式，生成优质的最终排版，用户能根据自己的实际需求，对最终排版进行自由修改。同时用户对结果的评分也将对应用进行反馈，使应用的行为能够更加贴合用户。最后能够输出长图、PDF等方便用户发布、分享的文件格式。

项目还需具备社区共享的功能，用户可以既可以将自己调整得到的满意结果分享至社区，也可以将自己的精心设计与其他用户共享。应用还可以根据用户处理的文章类型，为其推荐社区中同类型的优秀案例。项目还将为用户提供强大的管理服务和量身定制服务。

根据我们的调查统计，用户对Temage的要求如下图所示：



图 1.2 用户心目中的核心功能重要程度

Temage在实现全部核心功能的基础上，还需要根据上述调查结果对界面、排版功能进行重点优化。做到界面简洁易用，服务优质迅速。

## 项目价值

在当今这个信息交流迅捷普遍，社交媒体当道的大环境下，此项目具有广大的潜在用户群体。市场上也正好缺少像Temage这样功能完善服务优质的自动图文排版软件，

根据我们发放的调查问卷，在我们的身边大约有一半的人常进行文案创作，其中排版时长占创作总时长比例超过20%的超过了半数。有85%的受调查者表示很期待Temage图文排版项目的诞生。



图 1.3 调查报告结果分析

## 创新点

### 功能性创新

将排版与深度学习技术结合。

首先可以通过分析文章，对其主题进行分类，推荐最符合文章主题的版式以供用户选择。其次，对输入的图片进行信息压缩、提取，与文字相匹配，找到图片在文字最合适的位置，嵌入文章中得到初排版。结合初排版和选取的版式，自动生成排版结果。

排版完成后允许用户根据自身需求对结果微调。用户对结果进行评分，反馈传入系统进行迭代，改善应用。

社区共享模块允许用户相互分享优秀的排版结果，也为用户提供一个交流和问题反馈的环境，帮助用户优化体验。通过分析用户处理的文章类型，还可以为用户推荐社区中同类型的排版案例作为借鉴。

### 非功能性创新

前端使用Tensorflow.js调用模型进行推断，后端使用Tensorflow调用模型进行推断，相互协作，分摊计算压力，减少运行所需时间，保证应用的流畅性和稳定性，提高用户的使用体验。

前后端分离，使用MVVM，提高开发效率，增强代码的可维护性。

## 解决思路

首先要解决的是文本分类和版式推荐。文本分类问题，可以使用较为成熟的模型进行训练、推断。推荐问题我们可通过文本分类的结果进行top-k推荐。

其次要解决的，是初排版的问题，可分为图像识别，文本嵌入和图文匹配三个子问题。图像识别问题可以使用是使深度学习在众多机器学习算法中脱颖而出的 CNN 模型，基于CNN开发的模型种类繁多，可供本项目进行挑选和fine-tuning。文本嵌入问题可以使用RNN-LSTM对文中单词或句子进行encode，得到表示单词或句子的向量。对于图文匹配问题，我们可以基于余弦计算等方法找到最为匹配的图片与文字，再使用基于统计的方法，对文章进行排版。

最后解决的是评分反馈迭代，用户每次使用后都需要根据满意度对结果进行评分，评分数据将存储起来，达到一定条数，筛选后再对模型进行训练。

## 系统亮点

### 功能性亮点

图片能够通过与文字匹配的方式自动嵌入合适的位置。

应用关注用户的实际体验，通过评分反馈的方式进行模型的迭代，以产生更加贴合用户需求的模型。同时社区分享功能也可以为用户创造一个良好的使用环境。

### 非功能性亮点

结合前端的Tensorflow.js和后端的Tensorflow，模型同时分布在前端与后端，能够减少两端的压力，减少运行时间，优化用户体验。

采用前后端分离的MVVM，使得项目逻辑清晰，前后端互不干扰，提高项目的可维护性和可移植性。

# 开发计划

## 最终呈现形式

Temage的最终呈现形式为一个能流畅运行于浏览器的Web应用，运用深度学习技术为用户智能排版。完成文本图片预处理、排版生成和社区共享三大核心模块，能够有效地为用户提供排版服务。保证图片嵌入位置的准确性，保证文章主题分析的准确性，保证排版结果的优质和应用的鲁棒性。同时保证界面简洁、优美、友好，服务流畅、稳定、优质，具有跨平台、跨浏览器的特性。

## 主要功能描述

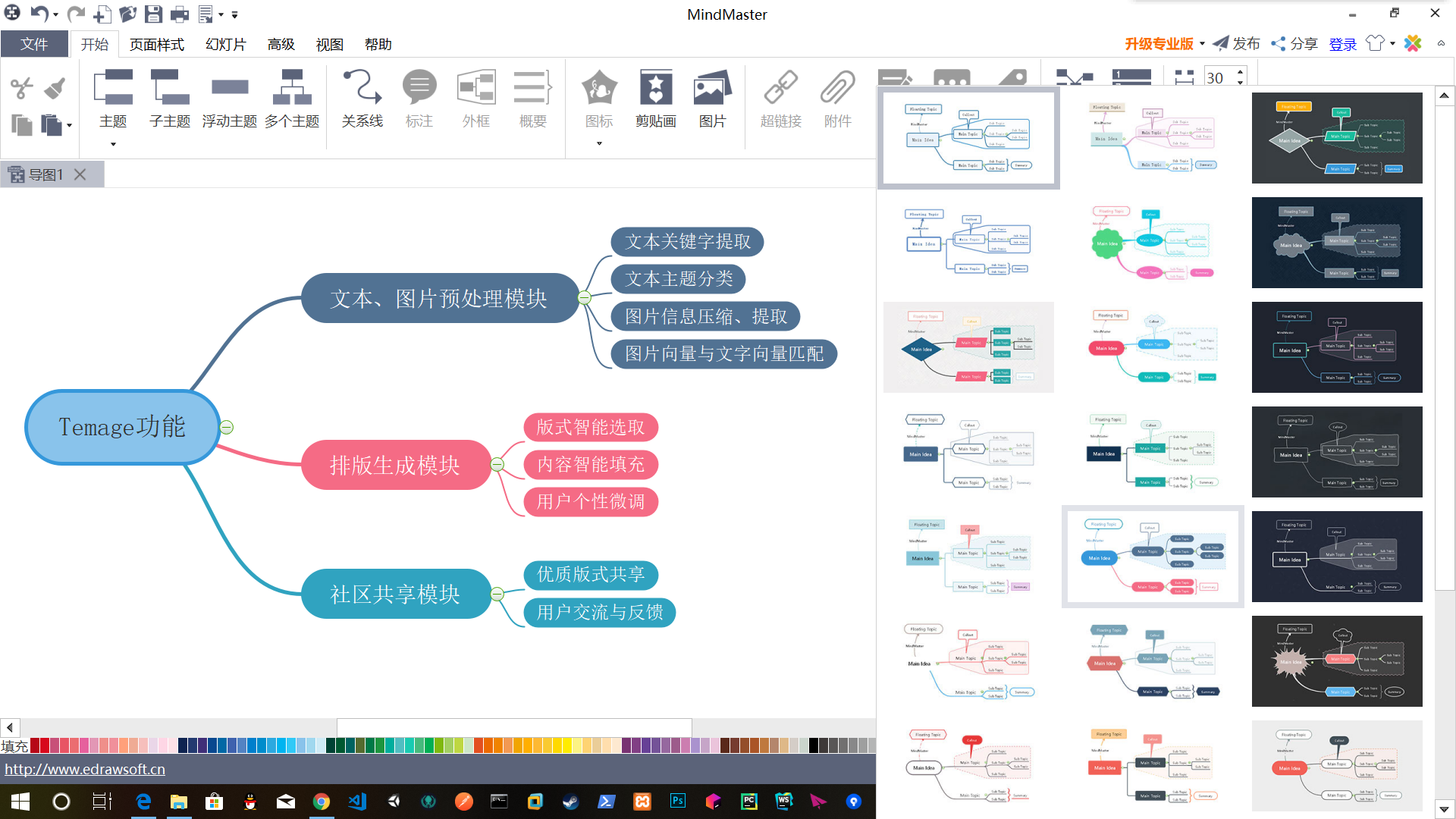


图 2.1 Temage三大功能模块

Temage功能分为三大模块，分别是文本、图片预处理模块，排版生成模块和社区共享模块。在预处理模块下要实现文本关键字提取，文本主题分类，图片信息压缩、提取，图片向量与文字向量匹配的功能。在排版生产模块下，要实现版式智能选取，内容智能填充，用户个性微调的功能。在社区共享模块下，要实现优质版式共享，用户交流和反馈功能，用户还可以浏览社区中与自己文章类型相关的案例，作为借鉴。

除此之外，Temage还有一个重要的功能，即根据用户反馈迭代model，使服务更加贴合用户，并可随时代潮流的变化而变化。

## 运行环境

主流浏览器，如Chrome、Fire Fox、Microsoft Edge等。

## 验收标准

应用实现效果符合预期，能够得到一个大致符合用户预期的排版效果。

应用要具备上述提到的三大功能模块，具有完整性。

应用能够满足目标人群的需求，提供流畅、稳定的优质服务。

界面设计符合用户人群的审美，对用户友好。

## 关键问题

数据集可能对于这个问题来说太小，导致训练出的模型效果不好。我们会尽力增大数据集，或者利用有限的数据集，尝试各种调参方法来改善模型效果。

## 进度安排

项目计划甘特图如下图所示：



图 2.2 Temage图文排版项目计划甘特图

## 开发预算

需要一台服务器，用于部署各应用。

# 可行性分析

## 市场可行性分析

### 与现行软件对比

现有的在线排版软件分两类，一类是在线的非自动排版应用，即给用户提供排版所需的功能，用户需要自己进行操作。比较典型的如用于公众号排版的“秀米”。比较适用于有美工基础的用户，不熟悉使用的用户需要一定的时间学习使用，具有一定的学习成本。

第二类是在线的自动排版应用，但只支持文字的在线排版，功能较为单一，包含修正标点、调整空行缩进、调整字号等功能，而且需要用户预先设置。

Temage则支持自动的图文排版，最大化地方便用户的使用，减少用户所需的操作，从而有效减少用户排版耗费的时间，同时使得没有美工基础的用户也能得到专业的排版设计。

## 技术可行性分析

### 功能简述

上文提到过，Temage包含三大功能模块，文字图片预处理模块、排版生成模块和社区共享模块，其中文字图片预处理模块和排版生成模块需要用到深度学习的技术。

文字图片预处理模块包含四部分：文本关键词提取、文本主题分类、图片信息压缩提取、图片向量与文字向量匹配。

排版生成模块包含三部分：版式智能选取，内容智能填充、用户个性微调。

社区共享模块包含两部分：优质版式共享、用户交流与反馈。

### 技术要素

上述功能所需的技术主要有以下几点：

文本关键词的提取，使用TF-IDE模型，用统计的方法找出文本中关键词，为提高搜索效率，我们可以使用sphinx开源搜索引擎框架对数据库建立索引。图片的识别，使用CNN模型。文本的分类使用ELMo+LSTM+TextCNN模型。根据文本的分类，使用Top-k方法推荐合适的版式。图片文字的匹配，基于余弦计算等方法匹配图片在文字的位置，进行图片与文字的初排版。内容的智能填充，使用RNN-LSTM对文中单词或句子进行encode，得到表示单词或句子的向量，然后进行填充处理。

## 资源可行性分析

人力资源包括队员四人。模型训练数据集：今日头条数据集，共382688条，分布于15个分类中。用于文本分类进行风格推荐的模型训练，采集时间为2018年5月，可用于文本分类和图片识别模型的训练。

# 需求分析

## 数据描述

### 静态数据

静态数据是指在运行过程中主要作为控制或参考用的数据,它们在很长的一段时间内不会变化，一般不随运行而变。

### 动态数据



### 数据词典



### 数据采集

1. 今日头条数据集，共382688条，分布于15个分类中。用于文本分类进行风格推荐的模型训练，采集时间为2018年5月，凭借此我们将得到新闻中的图片和图片上下文，上述数据可用于文本分类的训练和图文匹配的训练。
2. 用户上传数据，用户在使用本产品中将上传相应的文本和图片，在最终产出产品之后该数据可用于继续进行模型的更新。

## 功能需求

### 功能模块结构图

### 核心功能模块描述

表 4.1 核心功能模块描述

|  |  |  |  |
| --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **优先级** |
| **文本预处理模块** | 风格推荐 | 使用TextCNN进行文本分类，结合用户使用习惯，提供风格推荐 |  |
| 文本单句词向量获取 | 使用ELMo+LSTM生成文本单句的向量，作为后序的文本分类和图文匹配的输入 |  |
|  |  |  |
| **图像预处理模块** | 图像向量嵌入 | CNN通过对图像做各种操作之后，将图像编码为向量化结构，作为后序图文匹配的输入 |  |
| **产品生成模块** | 图文匹配 | 根据图像预处理和文本预处理结果进行图文匹配 |  |
| 风格融合 | 将用户选择的风格融合到长图生成中，进行美化 |  |
| **用户反馈模块** | 用户最终产品反馈 | 用户对文本进一步调整后，我们将记录用户的排版顺序，并将该结果用于后序模型的训练之中 |  |
| 用户风格选择反馈 | 记录用户最终选择的风格于用户习惯中，进一步更新用户习惯模型和文本分类模型 |  |
| 用户搜索 | 用户使用关键词搜索提供相应作品的显示，供用户参考 |  |

## 非功能性需求

### 扩展性和可维护性

1. 功能扩展一，可以与各大社交媒体平台进行对接，例如微信，微博，有利于自媒体的进一步发展
2. 功能扩展二，进一步完善社区功能，鼓励用户之间进行交流和借鉴，提升图文编辑的体验感
3. 可维护性：使用MVVM+ Micro Service 架构，可维护性强。
4. 与深度学习相关的大计算量组件（ServerA）与维护社区和日常用户操作组件（ServerB）相互独立，可分别进行扩展与维护。

### 易用性

Temage始终站在用户的角度进行思考

1. 用户在进行操作时具有详尽的操作导航。
2. 页面的editor功能齐全，可以让用户拥有一站式编辑体验。
3. 对于用户编辑结果进行实时保存，防止用户网络不畅通或不小心关闭页面带来的不必要损失。

## 性能需求

### 时间特性

Temage主要耗时模块为文本预处理模块和图像预处理模块，为分担服务器压力，加快运行速度，提升用户体验，我们将部分文本预处理工作交给前端（浏览器）进行处理。我们将文本分类TextCNN模型放在前端进行执行，并在用户进行风格选择的同时，ServerA继续进行图像预处理的相关工作。

### 适应性

1. Temage作为web应用在浏览器中运行，由于需要在前端进行深度学习的推断，用户只需使用有相应能力的浏览器（例如Chrome Safari）均可以使用Temage进行工作。
2. Temage在进行维护升级后，如若对操作方式产生影响，均会在新版本中给予用户显著的提示，确保用户的流畅使用。

## 运行需求

### 用户界面

### 硬件接口

无

### 软件接口

无

### 故障处理

本项目采用Micro Service架构，即便是某服务器出现故障，仍将有部分功能可以继续使用，例如，深度学习的相关的serverA发生故障，不能进行深度学习的推断，serverB正常服务，用户仍旧可以登录进行游廊的游览。server端故障将生成详细的log日志于服务器中，并且会有友好的用户提示。

## 用例图

## 核心模块用例规约

表 4.2 风格推荐用例规约

|  |  |
| --- | --- |
| 用例名称 | 风格推荐 |
| 功能简述 | 讲用户上传的文本分为十六类 |
| 用例编号 | 001 |
| 执行者 | Font-End |
| 前置条件 | 1. 用户上传文本为中文文本 2. 用户上传文本逻辑清晰，行文流畅 3. 用户登录 |
| 后置条件 | 内存储存文本分类和风格推荐的结果 |
| 涉众利益 | 用户：希望得到合适的风格推荐  风格融合：服务器希望得到合适的分类选取合适的美化方式 |
| 基本路径 | 1. 用户上传文字 2. 服务器对文本进行句子嵌入后将结果发送至前端 3. 前端TextCNN模型进行推断得到推荐风格 4. 对用户进行推荐，并让用户进行选择 |
| 扩展路径 | 1. 文本内容为空，将提醒用户 2. 文本内容中文内容过少或不存在，提示用户 |
| 字段列表 | 文本合理，文本类别 |
| 设计规则 | 单独模块 |
| 未解决的问题 | 大量非中文文本将降低分类的准确度 |
| 备注 | 无 |

表 4.3 图像向量嵌入用例规约

|  |  |
| --- | --- |
| 用例名称 | 图像向量嵌入 |
| 功能简述 | 用户上传图像进行向量嵌入 |
| 用例编号 | 002 |
| 执行者 | ServerB |
| 前置条件 | 1. 用户上传图片大小限制 2. 用户上传图片与文本有关联 3. 用户登录 |
| 后置条件 | 内存储存向量矩阵，用于之后的图文匹配 |
| 涉众利益 | 图文匹配：服务器希望得到合适向量，有利于图文匹配 |
| 基本路径 | 1. 用户上传图片和文字 2. ServerA图像进行向量嵌入 3. ServerA向ServerB发送结果 |
| 扩展路径 | 1. 图片内容为空，将提醒用户 2. 图片图像分辨率过大或过小都将警告用户 |
| 字段列表 | 图像编号，图像向量矩阵 |
| 设计规则 | 单独模块 |
| 未解决的问题 | 向量嵌入将消耗较长时间 |
| 备注 | 无 |

表 4.4 智能排版用例规约

|  |  |
| --- | --- |
| 用例名称 | 智能排版 |
| 功能简述 | 完成对用户素材的排版 |
| 用例编号 | 003 |
| 执行者 | ServerA |
| 前置条件 | 1. 用户已上传符合要求的图片与文字 2. 用户已选取作品风格 3. 用户登录 |
| 后置条件 | 内 |
| 涉众利益 | 用户：获取产品。  用户反馈模块：获取反馈实例，进入用户反馈阶段。 |
| 基本路径 | 1. 用户上传图片和文字 2. ServerA将图文匹配结果和风格对应样式融合，发送至前端 |
| 扩展路径 | 无 |
| 字段列表 | Html文本，状态码 |
| 设计规则 | 单独模块 |
| 未解决的问题 | 无 |
| 备注 | 无 |

表 4.5 关键词提取用例规约

|  |  |
| --- | --- |
| 用例名称 | 关键词提取 |
| 功能简述 | 完成对文本关键词提取，用于用户进行搜索 |
| 用例编号 | 004 |
| 执行者 | ServerA |
| 前置条件 | ① 存在已生成的作品 |
| 后置条件 | 无 |
| 涉众利益 | 用户：获取产品。  用户反馈模块：获取反馈实例，进入用户反馈阶段。 |
| 基本路径 | 1. 用户生成最终作品 2. ServerA使用关键词提取算法得到作品文本关键词，存入数据库 |
| 扩展路径 | 无 |
| 字段列表 | 状态码 |
| 设计规则 | 单独模块 |
| 未解决的问题 | 无 |
| 备注 | 为减少用户的响应时间，该操作可能为异步操作 |

表 4.6 关键词搜索用例规约

|  |  |
| --- | --- |
| 用例名称 | 关键词搜索 |
| 功能简述 | 用户进行搜索，寻找相关的已有作品 |
| 用例编号 | 005 |
| 执行者 | ServerA |
| 前置条件 | 1. 存在已生成的作品 |
| 后置条件 | 无 |
| 涉众利益 | 用户：获取相关作品，用于借鉴。 |
| 基本路径 | 1. 用户进行搜索 2. ServerA使用sphinx框架搜索，得到相关数据返回前端 |
| 扩展路径 | 无 |
| 字段列表 | 作品数据 |
| 设计规则 | 单独模块 |
| 未解决的问题 | 无 |
| 备注 | 无 |

表 4.7 产品参考用例规约

|  |  |
| --- | --- |
| 用例名称 | 产品参考 |
| 功能简述 | 完成对文本关键词提取，用于用户进行搜索 |
| 用例编号 | 006 |
| 执行者 | ServerA |
| 前置条件 | ① 存在已生成的作品 |
| 后置条件 | 无 |
| 涉众利益 | 用户：用户可看到已存在的作品。 |
| 基本路径 | 1. 用户通过搜索或浏览游廊看到作品 |
| 扩展路径 | 服务器出错或无权限观看（403） |
| 字段列表 | 作品列表，状态码 |
| 设计规则 | 单独模块 |
| 未解决的问题 | 无 |
| 备注 | 无 |

表2.7 产品参考用例规约

|  |  |
| --- | --- |
| 用例名称 | 产品参考 |
| 功能简述 | 完成对文本关键词提取，用于用户进行搜索 |
| 用例编号 | 007 |
| 执行者 | ServerA |
| 前置条件 | 1. 存在已生成的作品 |
| 后置条件 | 无 |
| 涉众利益 | 用户：可对其他产品进行评价。 |
| 基本路径 | 1. 用户通过搜索或浏览游廊看到作品 2. 用户填写产品评价 3. 用户提交产品评价 |
| 扩展路径 | 服务器出错或无权限观看（403） |
| 字段列表 | 作品ID，用户ID，作品评价 |
| 设计规则 | 单独模块 |
| 未解决的问题 | 无 |
| 备注 | 无 |

## 其他需求

* 1. 为保证用户的文字图片的版权，用户可以选择在生成的长图上使用本平台的水印，以保障自己的权益。

# 概要设计

## 处理流程

## 总体结构和模块设计

## 功能分配

## 接口设计

### 外部接口

### 内部接口

### 用户界面设计

## 数据结构设计

### 逻辑结构设计

### 物理结构设计

### 数据结构与程序的关系

## 运行设计

### 运行模块关系

### 运行控制

### 运行时间

## 错误/异常处理设计

### 错误/异常输出信息

### 错误/异常处理对策

## 系统配置策略

## 系统部署方案

## 代码规范

## 其他相关技术与方案

# 数据库设计

# 详细设计

## \*\*功能模块

### 功能描述

### 性能描述

### 输入

### 输出

### 算法

### 程序逻辑

### 接口

### 存储分配

### 限制条件