**利用协同过滤算法进行推荐**

本系统为工业从业者提供了技术问答，专业知识文献等方便而实用的功能，但是在繁多的信息中，找到自己想要看的资料可能需要花费大量的时间搜索，筛选，分辨……这时，一个稳定适用的推荐系统的出现显得尤为重要。

本系统主要采用基于物品的协同过滤算法，在用户的实际使用过程中，系统会记录下用户的各种行为，每一种行为在系统中都赋予了对应的权重。大量用户在一段时间的使用后，每份资料，包括技术问答，文章等等，都会形成一个用户评分矩阵，该矩阵为每个用户对这份资料的综合评分，而评分则是根据用户平时的行为进行加权计算得出。

最后，计算各份资料之间的余弦相似度，即可得到资料之间的相关度，接着便根据相关度进行推荐。

在这一过程中，如何计算权重决定了推荐系统是否实用。下面是本系统将用户的行为进行分类和加权的部分信息：

行为 权重

浏览问题 2

关注问题 3

回答问题 3

对问题的评论点赞 3

举报问题 2

浏览文章 2

收藏文章 4

点赞文章 3

点踩文章 2

评论文章 3

举报文章 2

以文章的行为举例，浏览文章作为最基本的用户行为，基于较低的权重2，当用户未对某篇文章有任何行为的时候，默认权重为1。而收藏文章则说明该文章对该用户的作用较大，因此基于较高的权重4。而点赞和评论文章说明用户一定完全并且较为认真地阅读过文章，这也能反应用户地兴趣点，因此给予权重3.而点踩和举报文章虽然能反映用户地认真阅读程度，但是这些负面行为同时也表明了这些文章无法对用户提供帮助，因此和基本浏览行为同权2。

**基于词频-逆文本频率指数的模糊搜索匹配系统**

当推荐的内容不能满足用户时，搜索是用户的不二选择。然而，搜索结果量大而杂，结果的前几项能高度符合用户的需求是必须的。由于本系统的特征是问答和文章交流，因此搜索结果也多为文献资料，采用词频-逆文本频率指数(tf-idf)能高效地表示两段文字地相关程度。

当用户在搜索框输入搜索内容时，若搜索内容为短语或较长的语句，则系统便会与数据库内积累的热门搜索项进行相似度计算，同时综合各个搜索项的热度给出最为相近的搜索词，以此来预测用户想要的输入。当然，tf-idf算法对短语之间的效果并不明显。当用户完成输入，正式进入搜索时，系统首先使用平常的模糊匹配关键字获取相关资料，然后计算资料内关键字的词频和逆文本频率指数，计算出tfidf值。最后按照tfidf值降序排列即为最终呈现给用户的搜索结果。

**采用卷积神经网络的智能分类**

本系统专注于特定几个方向的技术交流：减震降噪 传感器 新材料应用 绿色能源 射频技术 人工智能 流体机械 水处理技术 食品技术 制冷系统设计 燃烧技术 空气净化技术 机械结构设计 加热技术。当用户发布文章或问答的时候，智能地进行分类可以为用户的使用提供很大的便利，同时也减轻了管理员筛选分类资料的压力。因此本系统

使用CNN进行训练。网络结构为256核的卷积层，卷积核尺寸为5\*5，后跟最大池化层，最后为全连接层，并使用0.5的dropout率。在训练之前，需要对训练集进行预处理，第一步将所有训练集进行分词，并建立词汇表。词汇表中去除了常用并且不对文章意义产生影响的停用词

**OCR图像处理识别身份证信息**

为了减轻用户在加入本站时所需要填写的各种资料和信息,本系统提供了身份证图片上传自动捕获关键信息的功能。虽然市面上有阿里，腾讯等公司提供现有的ocr身份证识别，但本系统采用tesseract-ocr作为主要工具，并进行了深度地校准和训练。本系统在接受新用户时需要获得出生日期，姓名，身份证号码等关键信息，而未经过训练的ocr识别率十分不理想

|  |  |  |  |
| --- | --- | --- | --- |
| 次数/识别率/目标 | 姓名 | 出生日期 | 身份证 |
| 10 | 4/10 | 2/10 | 6/10 |
| 100 | 23/100 | 11/100 | 67/100 |
| 1000 | 309/1000 | 88/1000 | 747/1000 |

由于大量身份证的来源较为困难，我们通过背景图片+文字合成的方式生成包含不同信息的身份证图片。依据相关资料将字体和汉字所在位置与真实身份证进行匹配，并且添加了水印和部分噪点。同时，身份证信息中除了名字是以百家姓+名字常用字库中随机抽取组合，其他的信息均做了真实化处理。例如生日和身份证号码的匹配，性别和身份证号码的匹配，以及居住地区采用真实存在的省份，城市，街道，并与身份证的校检码匹配。一切都为了尽量贴合真实身份证。

在初始测试后，低下的识别率并不能完成本系统的需求，因此我们进行大量操作以提高识别率

**训练专用字库**

识别率不高的最大因素就是无法正确识别汉字。由于tesseract不自带中文字库，而网上提供的字库也不符合本系统的使用情景，因此训练专用字库成了首要优化方案。

在优化过程中，我们采用jTessEditor工具的辅助，为大量身份证数据集创建汉字识别定位信息，由两位后端负责矫正，合并，拆分等操作，将所有错误识别信息矫正，并补充完整漏识别，删除额外识别内容后，新的字库便通过漫长的训练过程，正式能够投入使用了。

在新的字库应用后，识别率如下

|  |  |  |  |
| --- | --- | --- | --- |
| 次数/识别率/目标 | 姓名 | 出生日期 | 身份证 |
| 10 | 4/10 | 5/10 | 7/10 |
| 100 | 36/100 | 27/100 | 85/100 |
| 1000 | 411/1000 | 223/1000 | 792/1000 |

三项信息的识别率有了些许提高，但还是未能达到使用标准，于是设计人员进行下一步优化

**获取边界裁剪身份证区域**

由于真实的图片来源复杂，背景干扰大，仅仅截取身份证的内容并识别可能可以提高准确率。在这方面，设计人员采用opencv提供的库进行处理，处理过程中发现，若光线过亮或过暗，身份证边缘有异物阻挡，都会使边界识别失败。克服这些问题后，获取边界的准确度就大大提升了

**放大图片**

由于图片的清晰度不固定，模糊的图片可能无法正常识别，因此放大合适的比例可以一定程度上增加识别率

**灰度化图片**

原始图片为RGB三色通道的彩色图片，这在识别文字信息时是冗余的，并且不利于后续的处理，将彩色图片转化为灰度图片后，识别率可以上升‘

**二值化图片**

图片灰度化之后，身份证背景还存在干扰线条，头像等内容，严重影响识别率，在柔和的日光/白灯光/黄灯光下，经过多次测试，得到一个阈值，根据阈值将图片二值化，最终能产生仅存文字信息的黑白图片，使识别率进一步提高

**反色相加去除背景和噪点**

二值化后的图片存在很多噪点，这是由于阈值固定，而光线的变化范围大，固定阈值不能完美地去除干扰信息，反而会产生干扰，并且使字体发生一些变化。设计人员研究以后，采用反色相加，具体操作为，将原图的黑色部分涂白，然后整幅图反色处理，得到新的图片，并和原图的颜色通道相加，所有彩色部分都变成白色，黑色的字能够保留下来，这能提高最终的识别率

在多项处理之后，ocr 的识别率如下

|  |  |  |  |
| --- | --- | --- | --- |
| 次数/识别率/目标 | 姓名 | 出生日期 | 身份证 |
| 10 | 8/10 | 7/10 | 9/10 |
| 100 | 81/100 | 85/100 | 93/100 |
| 1000 | 843/1000 | 897/1000 | 916/1000 |

有了显著的提升