中国矿业大学
CHINA UNIVERSITY OF MINING AND TECHNOLOGY

基于Bert-UNILM的标题生成算法

——以2022年软件杯智能创作平台为例

计算机科学与技术学院　计算机科学与技术2019-04班
学号：06192081　06192103
答辩人：胡钧耀　黄凯

目 录

CONTENTS

01 | 问题背景

02 | 项目展示

03 | 算法技术分析

04 | 改进与思考

01 | 问题背景

问题背景
项目展示
算法设计与分析
改进与思考

研究课题背景

计算法学的展示了AI赋能传统文科专业，辅助人类处理文本语言

2021法研杯在最高人民法院和中国中文信息学会的指导下，开始探究AI赋能

计算法学分会

（Computational Law 缩写CCF CL）

计算法学分会拟于2021年底成立，委员包括海内外知名高校的学者，司法系统的实务专家，公司法务部长和知名律师以及科技企业界人士，覆盖90余家单位，是国内首个专门针对计算科学与法学交叉领域进行深度研究的新兴学术组织。

计算法学的研究领域包括：

（1）基于中国大数据优势的预测式侦查和警务以及电子证据，同时开展关于判决预测和法律文书自动生成的实证研究；

（2）立法、司法、执法以及纠纷解决的智能化，在重点场景形成深度应用创新的产学研一体化生态社区，不断开发新型法律科技和服务软件；

（3）计算机语义系统以及规则本位和案例本位的自动法律推理，致力于提升计算法学应用基础研究和科学理论研究的水准；

（4）数字经济发展中涌现的各种法律科技问题和知识产权问题的解决，探索数据信托功能的不同机制设计；

（5）加强关于数据伦理、数据合规、算法公正、算法透明的国际对话，健全人工智能治理体系。

当前，因为物联网、大数据、人工智能的合力效应，社会治理模式和法学研究范式正经历着变革，呈现出法学与科技深度交叉融合的趋势。专委会希望成为沟通法学界、计算机科学界、法律实务界、新兴科技企业界以及各种行业协会的重要桥梁，通过将相关学科及行业领域资源进行汇总、建设计算法学领域活动大平台、建设二级法学学科的科研细分平台以及建立奖励机制等工作，促进本研究领域的学术交流和产学研合作，培养贯通理工科和人文社科知识、兼具解决问题和批判思考能力的复合型人才，并承担计算法学相关教育、知识普及等社会服务。

问题背景
项目展示
算法设计与分析
改进与思考

参考文献

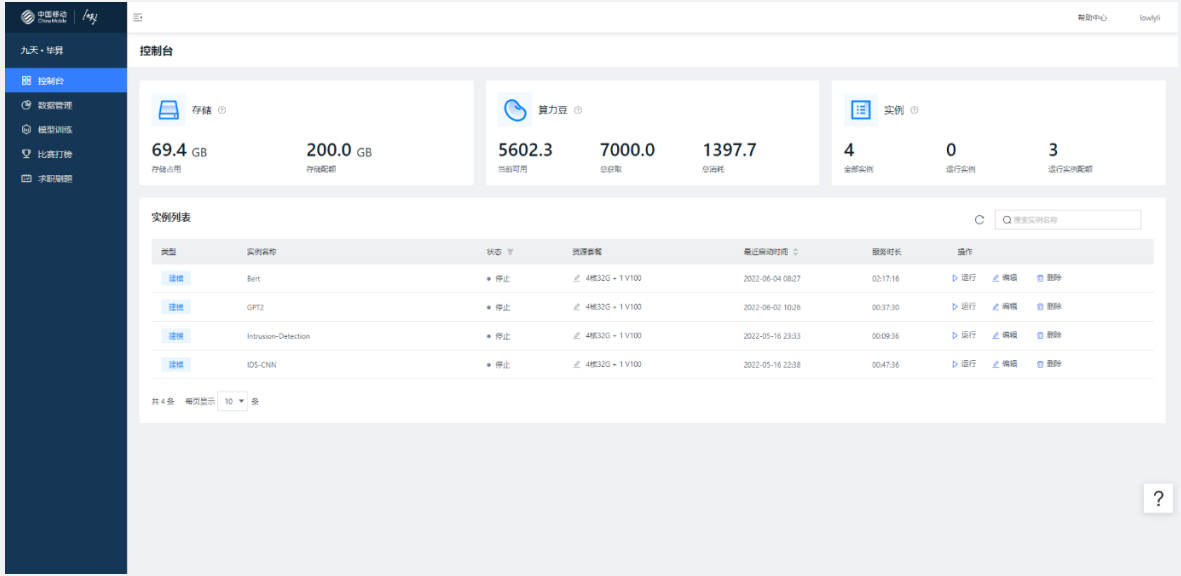
名称	类型	大小
Attention Is All You Need.pdf	Adobe Acrobat 文档	2,151 KB
bert.pdf	Adobe Acrobat 文档	757 KB
GPT2-language_models_are_unsupervised_multitask_learners.pdf	Adobe Acrobat 文档	570 KB
GPT3.pdf	Adobe Acrobat 文档	6,610 KB
GPT-language_understanding_paper.pdf	Adobe Acrobat 文档	529 KB
LSTM.pdf	Adobe Acrobat 文档	554 KB
NIPS-2014-sequence-to-sequence-learning-with-neural-networks-Paper.pdf	Adobe Acrobat 文档	140 KB
TextRank.pdf	Adobe Acrobat 文档	115 KB
UNILM.pdf	Adobe Acrobat 文档	412 KB
ZEN Pre-training Chinese Text Encoder.pdf	Adobe Acrobat 文档	1,023 KB

02 | 项目展示

问题背景
项目展示
算法设计与分析
改进与思考

V100显卡累计运行90+h

尝试GPT2,BERT,以及改进后的bert



问题背景
项目展示
算法设计与分析
改进与思考

最终模型只训练了30epoch（算力跟不上），按照batch_size=5，只遍历了训练集6次，如果扩大效果可以更加优化。

```
Train for 1147 steps
Epoch 1/30
/opt/conda/lib/python3.7/site-packages/tensorflow_core/python/framework/indexed_slices.py:433: UserWarning: Converting sparse IndexedSlices to a den
  "Converting sparse IndexedSlices to a dense Tensor of unknown shape. "
/opt/conda/lib/python3.7/site-packages/tensorflow_core/python/framework/indexed_slices.py:433: UserWarning: Converting sparse IndexedSlices to a den
  "Converting sparse IndexedSlices to a dense Tensor of unknown shape. "
1147/1147 [==============================] - 818s 714ms/step - loss: 3.2821
Epoch 2/30
1147/1147 [==============================] - 786s 685ms/step - loss: 2.2896
Epoch 3/30
1147/1147 [==============================] - 779s 679ms/step - loss: 1.9720
Epoch 4/30
1147/1147 [==============================] - 781s 681ms/step - loss: 1.7298
Epoch 5/30
1147/1147 [==============================] - 787s 686ms/step - loss: 1.5234
Epoch 6/30
1147/1147 [==============================] - 785s 685ms/step - loss: 1.3333
Epoch 7/30
1147/1147 [==============================] - 787s 686ms/step - loss: 1.1704
Epoch 8/30
1147/1147 [==============================] - 786s 685ms/step - loss: 1.0120
Epoch 9/30
1147/1147 [==============================] - 784s 683ms/step - loss: 0.8697
Epoch 10/30
1147/1147 [==============================] - 783s 683ms/step - loss: 0.7423
Epoch 11/30
1147/1147 [==============================] - 786s 685ms/step - loss: 0.6243
Epoch 12/30
1147/1147 [==============================] - 789s 688ms/step - loss: 0.5208
Epoch 13/30
1147/1147 [==============================] - 784s 684ms/step - loss: 0.4257
Epoch 14/30
1147/1147 [==============================] - 783s 682ms/step - loss: 0.3501
Epoch 15/30
```

```
metrics = evaluator.evaluate(test_data)  # 评测模型
print('valid_data:', metrics)
```

✓ 201m 17.8s

```
100%|██████████| 1679/1679 [3:21:17<00:00,  7.19s/it]

valid_data: {'rouge-1': 0.3041298785856503, 'rouge-2': 0.19703214401602984, 'rouge-l': 0.2872118283372123, 'bleu':
0.13044759471052714}
```

```
1147/1147 [==============================] - 836s 728ms/step - loss: 0.0633
Epoch 27/30
1147/1147 [==============================] - 842s 734ms/step - loss: 0.0598
Epoch 28/30
1147/1147 [==============================] - 821s 716ms/step - loss: 0.0551
Epoch 29/30
1147/1147 [==============================] - 801s 699ms/step - loss: 0.0509
Epoch 30/30
1147/1147 [==============================] - 799s 697ms/step - loss: 0.0484
2022-06-02 16:51:22.036229: W tensorflow/core/kernels/data/generator_dataset_op.cc:103] Error occurred when finalizing GeneratorDataset iterator: Ca
[24]: <tensorflow.python.keras.callbacks.History at 0x7f171422a710>
```

问题背景
项目展示
算法设计与分析
改进与思考

输入文字，自动生成关键词，摘要和标题(主要优化为标题)

文章新建

文章属性

作者名 lichunyang

标题 返乡创业青年谱写乡村振兴新篇章

摘要 同时，这些返乡创业年轻人善学习、肯钻研，在实现自我成长的同时，也让更多人看到乡村发展的机遇，带动更多人投身乡村振兴这片热土。让愿意留在乡村、建设家乡的人留得安心，让愿意扎根田野、回报乡村的人更有信心，在这场乡村与返乡创业青年的"双向奔赴"中，青春之花必将在田间地头绚丽绽放，共同绘就产业兴旺、生态宜居、乡风文明、治理有效、生活富裕的美好图景。可以说，年轻人返乡创业，拓展了乡村产业，让资金、技术、人才等要素加速向农村汇聚。近年来，越来越多返乡创业人员用坚持、专业和热爱在农村挥洒汗水，为乡村振兴注入了强劲动能。返乡创业青年为乡村振兴添砖加瓦，广袤乡村回馈他们以更多人生出彩的机会。90后返乡创业青年袁小梅从城市回到农村老家，发展养蜂产业，帮助乡亲致富。

类型 党建

关键词 返乡创业,年轻人,返乡创业,乡村产业,返乡创业青年,乡村振兴

返回列表 AI生成 立即提交

文章新建

从大山走出来又回到大山"的彝族小伙立克拢拢，帮助所在村庄的115户612名贫困群众脱贫，推动村集体资产从零增至231.7万元；90后返乡创业青年袁小梅从城市回到农村老家，发展养蜂产业，帮助乡亲致富；湖南省花垣县十八洞村的苗家女孩施林娇大学毕业后主动回到村里创业，通过拍摄短视频、直播等方式，让家乡的风景、美食、民俗、建筑为更多人所知.....如今，越来越多的年轻人选择返乡创业，在实现自身梦想的同时，带动乡亲们就业致富。

习近平总书记强调："推动乡村全面振兴，关键靠人。"广袤的田野，蕴藏着无限的机遇与希望，为许多人提供了人生出彩的舞台。农业农村部数据显示，截至今年4月，全国各类返乡入乡创业人员超过1100万。《"十四五"农业农村人才队伍建设发展规划》明确提出，到2025年，培育100万名农村创业带头人，返乡入乡创业人员1500万人。从发展种植养殖业、开办农家乐，到成为非遗传承人、发展农村电商，再到开发乡村旅游、创办小微企业.....近年来，越来越多返乡创业人员用坚持、专业和热爱在农村挥洒汗水，为乡村振兴注入了强劲动能。

青年是整个社会力量中最积极、最有生气的力量，在返乡创业、振兴乡村中也扮演着重要角色。不少返乡创业的年轻人接受过良好的教育、有过在市场打拼的经历，具有开阔的视野、活跃的思维，了解市场需求，善于培育高效优质农产品，延长产业链、价值链。同时，这些返乡创业年轻人善学习、肯钻研，在实现自我成长的同时，也让更多人看到乡村发展的机遇，带动更多人投身乡村振兴这片热土。可以说，年轻人返乡创业，拓展了乡村产业，让资金、技术、人才等要素加速向农村汇聚。

如何吸引更多有抱负、有能力的年轻人返乡创业，考验着各个地方落实引才政策的力度和温度。栽好梧桐树，自有凤凰来。从大力发展乡村产业，成立创业园、搭建创业平台，到回应返乡创业人员具体诉求，出台各类支持政策；从在项目落地、资金筹措、精细服务、技术护航等方面提供保障，到完善人才服务乡村激励机制，让农村的机会吸引人、让农村的环境留住人.....事实证明，凝聚政策合力，形成良好的干事创业环境，才能让返乡创业青年回得来、留得住、干得好，最大限度激发人才内在活力，让他们在农村广阔天地大施所能、大展才华、大显身手。

也需要看到，在泥土里扎下根基，从无到有、从有到优发展产业，蹚出一条新路来，可能遇到这样或那样的困难和挑战。从城市来到乡村打拼，同样面临角色的转换，需要适应从高楼大厦、灯火通明到田垄阡陌、鸟语虫鸣的环境变化；不少农业及相关的创业项目前期投入大、回报周期长，怎样打开局面、突破瓶颈、打造品牌等，都需要进行全面细致的考量。就此而言，返乡创业绝非轻而易举就能成功。走好这条路，离不开相关政策的支持，更需要返乡创业青年敢闯敢拼、不懈奋斗，攻坚克难、勇毅前行。

乡村振兴，是一个呼唤人才同时造就人才的舞台。返乡创业青年为乡村振兴添砖加瓦，广袤乡村回馈他们以更多人生出彩的机会。让愿意留在乡村、建设家乡的人留得安心，让愿意扎根田野、回报乡村的人更有信心，在这场乡村与返乡创业青年的"双向奔赴"中，青春之花必将在田间地

03 | 算法技术分析

问题背景
项目展示
算法设计与分析
改进与思考

训练集为长文本内容生成摘要和标题，相对短文本更难

train.json - Visual Studio Code [管理员]

train.json


```
[
    {
        "title": "望海楼美国打“台湾牌”是危险的赌博",
        "content": "近期，美国国会众院通过法案，重申美国对台湾的承诺。对此，中国外交部发言人表示，有关法案严重违反一个中国原则和中美三个联合公报规定，粗暴干涉中国内政，中方对此坚决反对并已向美方提出严正交涉。\n事实上，中国高度关注美国国内打“台湾牌”、挑战一中原则的危险动向。近年来，作为“亲台”势力大本营的美国国会动作不断，先后通过“与台湾交往法”“亚洲再保证倡议法”等一系列“挺台”法案，“2019财年国防授权法案”也多处触及台湾问题。今年3月，美参院亲台议员再抛“台湾保证法”草案。众院议员继而在4月提出众院版的草案并在近期通过。上述法案的核心目标是强化美台关系，并将台作为美“印太战略”的重要伙伴。同时，“亲台”议员还有意制造事端。今年2月，5名共和党参议员致信众议院议长，促其邀请台湾地区领导人在国会上发表讲话。这一动议显然有悖于美国与台湾的非官方关系，其用心是实质性改变美台关系定位。\n上述动向出现并非偶然。在中美建交40周年之际，两国关系摩擦加剧，所谓“中国威胁论”再次沉渣泛起。美国对华认知出现严重偏差，对华政策中负面因素上升，保守人士甚至成立了“当前中国威胁委员会”。在此背景下，美国将台海关系作为战略抓手，通过打“台湾牌”在双边关系中增加筹码。特朗普就任后，国会对总统外交政策的约束力和塑造力加强。其实国会推动通过涉台法案对行政部门不具约束力，美政府在2018年并未提升美台官员互访级别，美军舰也没有“访问”台湾港口，保持着某种克制。但从美总统签署国会通过的法案可以看出，国会对外交产生了影响。立法也为政府对台政策提供更大空间。\n然而，美国需要认真衡量打“台湾牌”成本。首先是美国应对危机的代价。美方官员和学者已明确发出警告，美国卷入台湾问题得不偿失。美国学者曾在媒体发文指出，如果台海爆发危机，美国可能需要“援助”台湾，进而导致新的冷战乃至与中国大陆的冲突。但如果美国让台湾自己面对，则有损美国的信誉，影响美盟友对同盟关系的支持。其次是对中美关系的危害。历史证明，中美合则两利、斗则两伤。中美关系是当今世界最重要的双边关系之一，保持中美关系的稳定发展，不仅符合两国和两国人民的根本利益，也是国际社会的普遍期待。美国蓄意挑战台湾问题的底线，加剧中美关系的复杂性和不确定性，损害两国在重要领域合作，损人又害己。\n美国打“台湾牌”是一场危险的赌博。台湾问题是中国核心利益，中国政府和人民决不会对此坐视不理。中国敦促美方恪守一个中国原则和中美三个联合公报规定，阻止美国会审议推进有关法案，妥善处理涉台问题。美国悬崖勒马，才是明智之举。\n（作者系中国国际问题研究院国际战略研究所副所长）"
    },
    {
        "title": "大力推进高校治理能力建设",
        "content": "在推进“双一流”高校建设进程中，我们要紧紧围绕为党育人、为国育才，找准问题、破解难题，以一流意识和担当精神，大力推进高校的治理能力建设。\n增强政治引领力。坚持党对高校工作的全面领导，始终把政治建设摆在首位，增强校党委的政治领导力，全面推进党的建设各项工作。落实立德树人根本任务，把培养社会主义建设者和接班人放在中心位置。紧紧抓住思想政治工作这条生命线，全面加强师生思想政治工作，推进“三全育人”综合改革，将思想政治工作贯穿学校教育管理服务全过程，努力让学生成为德才兼备、全面发展的人才。\n提升人才聚集力。人才是创新的核心要素，创新驱动本质上是人才驱动。要坚持引育并举，建立绿色通道，探索知名专家举荐制，完善“一事一议”支持机制。在大力支持自然科学人才队伍建设的同时，实施哲学社会科学人才工程。立足实际，在条件成熟的学院探索“一院一策”改革。创新科研组织形式，为人才成长创设空间，建设更加崇尚学术、更加追求卓越、更加关爱学生、更加担当有为的学术共同体。\n培养学生竞争力。遵循学生成长成才的规律培育人才，着力培养具有国际竞争力的拔尖创新人才和各类专门人才，使优势学科、优秀教师、优质资源、优良环境围绕立德树人的根本任务配置。淘汰“水课”，打造“金课”，全力打造世界一流本科教育。深入推进研究生教育综合改革，加强事关国家重大战略的高精尖急缺人才培养，建设具有国际竞争力的研究生教育。\n激发科技创新力。在国家急需发展的领域挑大梁，就要更加聚焦科技前沿和国家需求，狠抓平台建设，包括加快牵头“武汉光源”建设步伐，积极参与国家实验室建设，建立校级
```

摘要，标题生成算法基本方法为**抽取和生成**
抽取中为聚类算法，textrank算法，分类算法，这里先进行降处理维采用textrank算法，基于概率图和马尔科夫链。
生成算法这里采用seq2seq架构，可以用CNN（卷积效果差），RNN+LSTM（曾经的主流），Transformer（目前主流）
利用谷歌和open AI基于Transformer改进的Bert和GPT进行生成。

GPT利用了Transformer解码器进行构建，在尝试过程中因为算力和训练集太小，故转向Bert，2021榜单现在为GPT3（海量数据预训练）

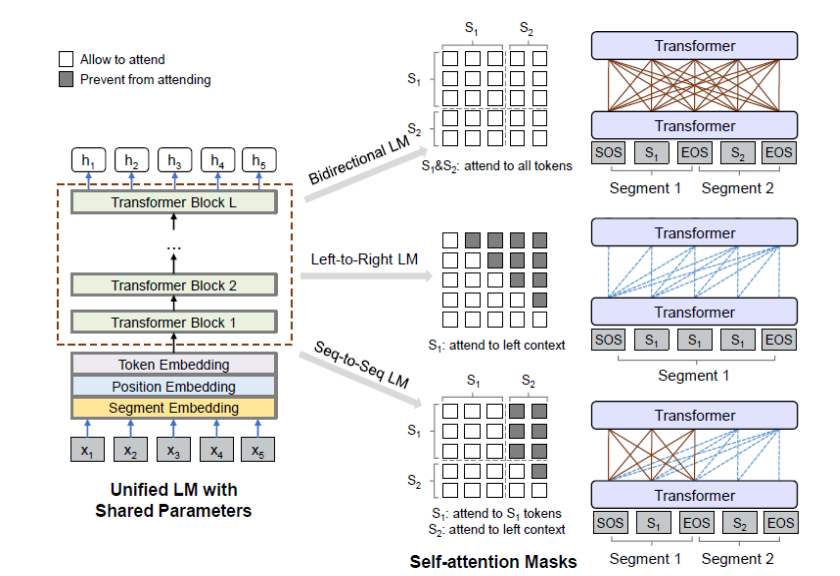
Bert采用Transformer的编码器，引入Mask机制和对Attention进行魔改，极大的提高了模型的准确率也更好应用于小样本微调。Bert的预处理采用了哈工大的中文训练（29天）。

问题背景
项目展示
算法设计与分析
改进与思考

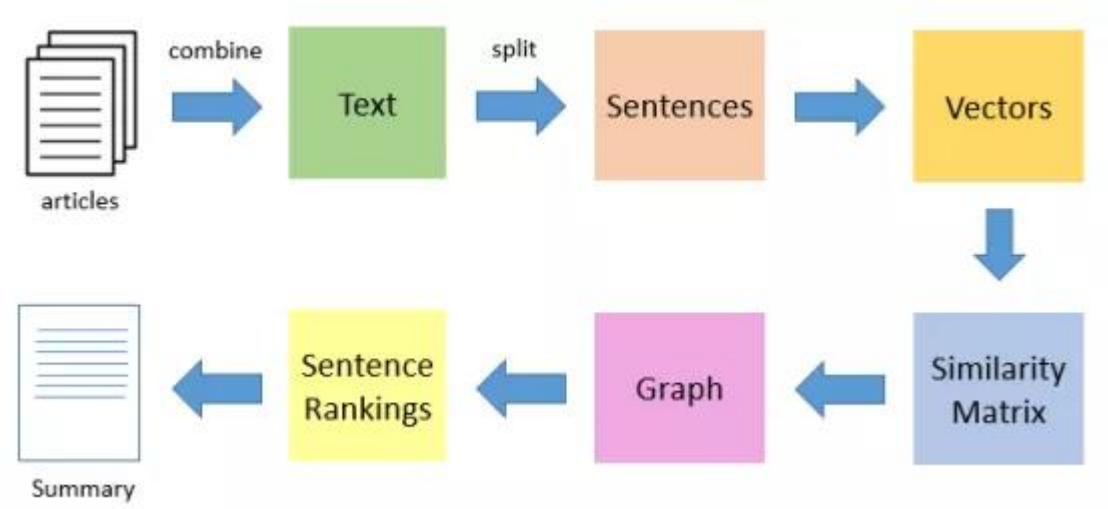
但在最终模型中，进一步更换为UniLM

UniLM是微软研究院在Bert的基础上，最新产出的预训练语言模型，被称为统一预训练语言模型。它可以完成单向、序列到序列和双向预测任务，可以说是结合了AR和AE两种语言模型的优点，Unilm在**抽象摘要、生成式问题回答和语言生成数据集的抽样领域**取得了最优秀的成绩。

2019年提出，2020年和2021多次冲上NLP领域榜单

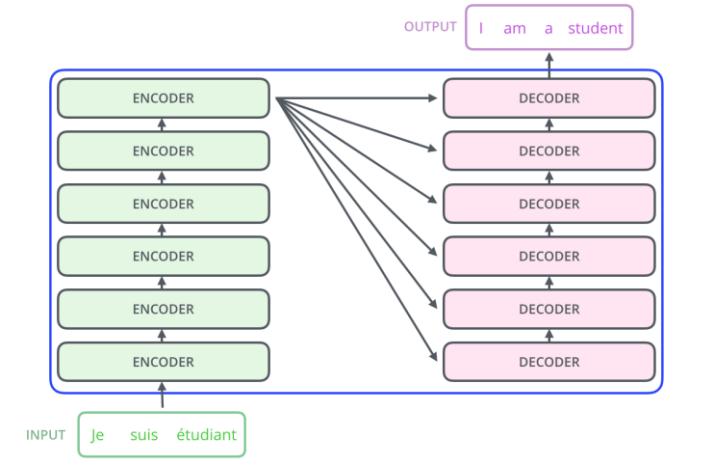


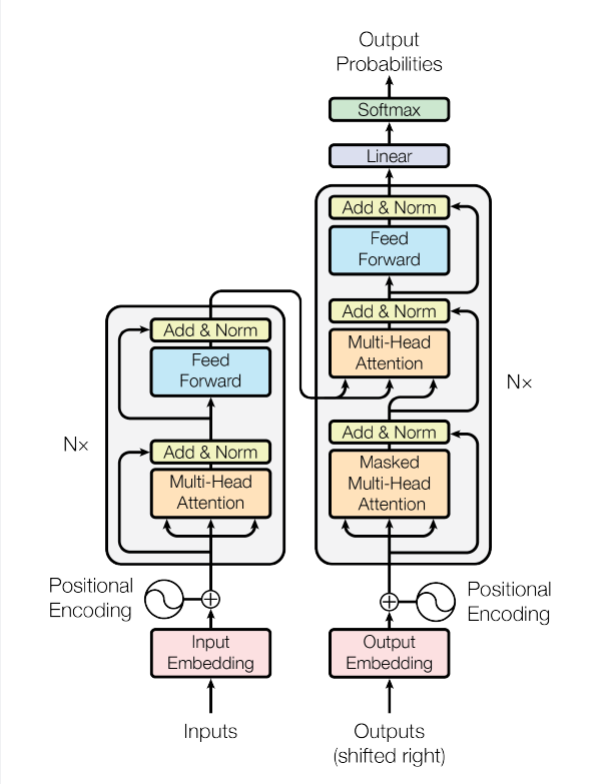
1.第一步是把所有文章整合成文本数据
2. 接下来把文本分割成单个句子
3. 然后，我们将为每个句子找到向量表示（词向量）。
4. 计算句子向量间的相似性并存放在矩阵中
5. 然后将相似矩阵转换为以句子为节点、相似性得分为边的图结构，用于句子TextRank计算。
6. 最后，一定数量的排名最高的句子构成最后的摘要。



问题背景
项目展示
算法设计与分析
改进与思考

Transformer框架，这里不细讲了，只需要知道整体框架就行，后面都是这个框架改动罢了，这里注意Attention和自回归训练



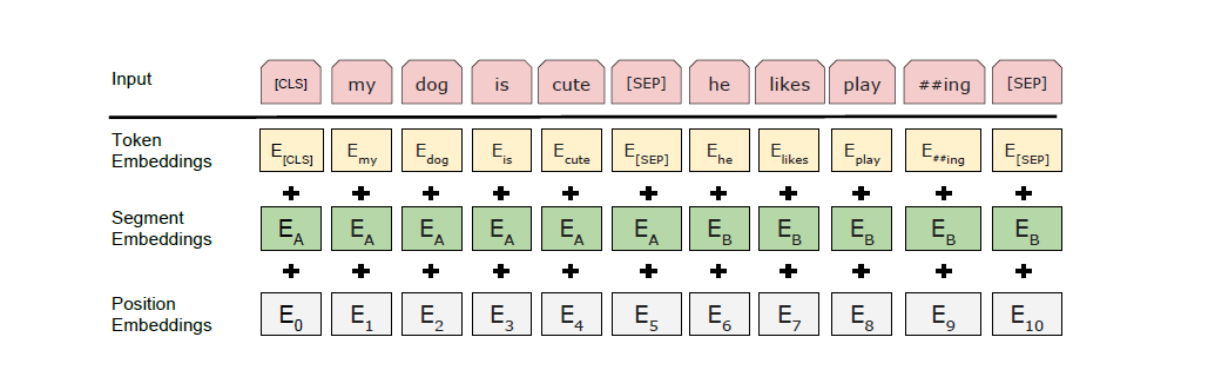


问题背景
项目展示
算法设计与分析
改进与思考

BERT将为NLP带来里程碑式的改变，也是NLP领域近期最重要的进展。

对比OpenAI GPT(Generative pre-trained transformer)，BERT是双向的Transformer block连接；就像单向RNN和双向RNN的区别，直觉上来讲效果会好一些。

这里的Embedding由三种Embedding求和而成：
就是扩展了vector，可以前后学习，并实现后续改进



BERT将为NLP带来里程碑式的改变，也是NLP领域近期最重要的进展。

任务1: Masked LM
采用前后文本学习，预训练类似做完形填空，猜中间的词，更好的理解文本。
任务2：下一句预测
传统的预测语句之间的关联性，进而拥有出色的语言处理能力

数据生成器将执行以下操作，而不是始终用[MASK]替换所选单词：

80%的时间：用[MASK]标记替换单词，例如，my dog is hairy → my dog is [MASK]
10%的时间：用一个随机的单词替换该单词，例如，my dog is hairy → my dog is apple
10%的时间：保持单词不变，例如，my dog is hairy → my dog is hairy. 这样做的目的是将表示偏向于实际观察到的单词。

问题背景
项目展示
算法设计与分析
改进与思考

通过不同任务，去优化同一分模型参数，在不同任务中，模型参数是共享的。

在预训练阶段，UniLM模型通过三种不同目标函数的语言模型（包括：双向语言模型，单向语言模型和序列到序列语言模型），去共同优化同一个Transformer网络；为了控制对将要预测的token可见到的上下文，作者使用了不同的self-attention mask来实现。换句话说，就是通过不同的掩码来控制预测单词的可见上下文词语数量，实现不同的模型表征。

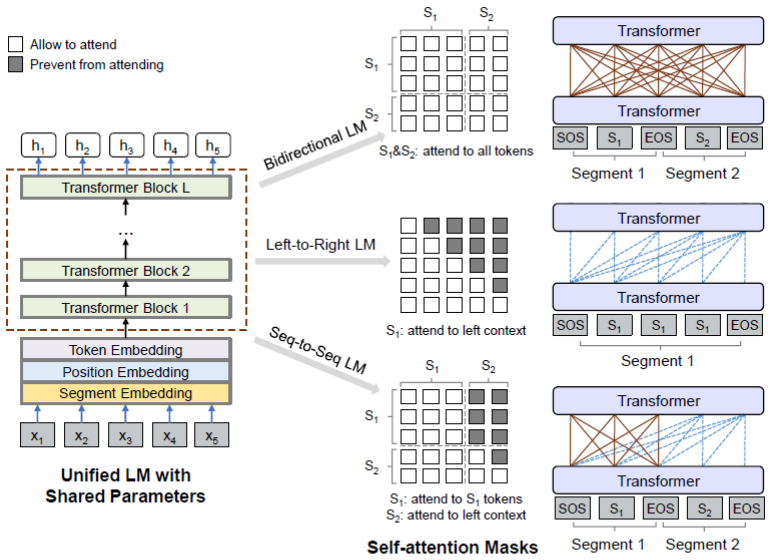


Figure 1: Overview of unified LM pre-training. The model parameters are shared across the LM objectives (i.e., bidirectional LM, unidirectional LM, and sequence-to-sequence LM). We use different self-attention masks to control the access to context for each word token. The right-to-left LM is similar to the left-to-right one, which is omitted in the figure for brevity.

问题背景
项目展示
算法设计与分析
改进与思考

自我改进：

改进交叉熵，利用稀疏**Softmax**构建Sparse Softmax

Sparse Softmax的思想源于《From Softmax to Sparsemax: A Sparse Model of Attention and Multi-Label Classification》、《Sparse Sequence-to-Sequence Models》等文章，大佬的改进这里借用。

```python
class CrossEntropy(Loss):
    """交叉熵作为loss，并mask掉输入部分
    """
    def compute_loss(self, inputs, mask=None):
        y_true, y_mask, y_pred = inputs
        y_true = y_true[:, 1:]  # 目标token_ids
        y_mask = y_mask[:, 1:]  # segment_ids，刚好指示了要预测的部分
        y_pred = y_pred[:, :-1]  # 预测序列，错开一位
        loss = K.sparse_categorical_crossentropy(y_true, y_pred)
        loss = K.sum(loss * y_mask) / K.sum(y_mask)
        return loss
```

问题背景
项目展示
算法设计与分析
改进与思考

模型调整：

重写编码解码，与任务匹配，实现标题的生成（主流都是摘要，标题准确率太低）

```python
class AutoTitle(AutoRegressiveDecoder):
    """seq2seq解码器
    """
    @AutoRegressiveDecoder.wraps(default_rtype='probas')
    def predict(self, inputs, output_ids, states):
        token_ids, segment_ids = inputs
        token_ids = np.concatenate([token_ids, output_ids], 1)
                                                    s, np.ones_like(output_ids)], 1)
                                                    _ids])[:, -1]
```

```python
class data_generator(DataGenerator):
    """数据生成器
    """
    def __iter__(self, random=False):
        batch_token_ids, batch_segment_ids = [], []
        for is_end, (title, content) in self.sample(random):
            token_ids, segment_ids = tokenizer.encode(
                content, title, maxlen=maxlen
            )
            batch_token_ids.append(token_ids)
            batch_segment_ids.append(segment_ids)
            if len(batch_token_ids) == self.batch_size or is_end:
                batch_token_ids = sequence_padding(batch_token_ids)
                batch_segment_ids = sequence_padding(batch_segment_ids)
                yield [batch_token_ids, batch_segment_ids], None
                batch_token_ids, batch_segment_ids = [], []
```

```python
e(text, maxlen=max_c_len)
, segment_ids],
于beam search
```

最终模型架构和结果：
累计训练接近1亿个参数
（NLP模型普遍偏大）
获得多个训练模型

Filter files by name

/ bert /

Name	Last Modified
bert4keras	7 days ago
best_model2_fin_2	13 days ago
chinese_wobert_L-12_H-768_A-12	2 years ago
csl	14 days ago
data	14 days ago
model	11 days ago
my_model	14 days ago
my_model_fin	14 days ago
my_model2_fin	13 days ago
sort_data	14 days ago
预测后处理.py	2 years ago
预处理.py	2 years ago
bert_data.zip	14 days ago
bert.zip	15 days ago
best_model_fin.weights.data-000...	14 days ago
best_model_fin.weights.data-000...	14 days ago
best_model_fin.weights.index	14 days ago
best_model.weights.data-00000-...	14 days ago
best_model.weights.data-00001-...	14 days ago
best_model.weights.index	14 days ago
bk-unilm.py	15 days ago

Launcher | train.ipynb

Code

```
Transformer-10-MultiHeadSelfAtt (None, None, 768)    0          Transformer-9-FeedForward-Norm[0]
                                                                Transformer-10-MultiHeadSelfAtten
Transformer-10-MultiHeadSelfAtt (None, None, 768)    1536       Transformer-10-MultiHeadSelfAtten
Transformer-10-FeedForward (Fee (None, None, 768)    4722432    Transformer-10-MultiHeadSelfAtten
Transformer-10-FeedForward-Drop (None, None, 768)    0          Transformer-10-FeedForward[0][0]
Transformer-10-FeedForward-Add  (None, None, 768)    0          Transformer-10-MultiHeadSelfAtten
                                                                Transformer-10-FeedForward-Dropou
Transformer-10-FeedForward-Norm (None, None, 768)    1536       Transformer-10-FeedForward-Add[0]
Transformer-11-MultiHeadSelfAtt (None, None, 768)    2362368    Transformer-10-FeedForward-Norm[0
                                                                Transformer-10-FeedForward-Norm[0
                                                                Transformer-10-FeedForward-Norm[0
                                                                Attention-UniLM-Mask[0][0]
Transformer-11-MultiHeadSelfAtt (None, None, 768)    0          Transformer-11-MultiHeadSelfAtten
Transformer-11-MultiHeadSelfAtt (None, None, 768)    0          Transformer-10-FeedForward-Norm[0
                                                                Transformer-11-MultiHeadSelfAtten
Transformer-11-MultiHeadSelfAtt (None, None, 768)    1536       Transformer-11-MultiHeadSelfAtten
Transformer-11-FeedForward (Fee (None, None, 768)    4722432    Transformer-11-MultiHeadSelfAtten
Transformer-11-FeedForward-Drop (None, None, 768)    0          Transformer-11-FeedForward[0][0]
Transformer-11-FeedForward-Add  (None, None, 768)    0          Transformer-11-MultiHeadSelfAtten
                                                                Transformer-11-FeedForward-Dropou
Transformer-11-FeedForward-Norm (None, None, 768)    1536       Transformer-11-FeedForward-Add[0]
MLM-Dense (Dense)               (None, None, 768)    590592     Transformer-11-FeedForward-Norm[0
MLM-Norm (LayerNormalization)   (None, None, 768)    1536       MLM-Dense[0][0]
MLM-Bias (ScaleOffset)          (None, None, 14068)  14068      Embedding-Token[1][0]
MLM-Activation (Activation)     (None, None, 14068)  0          MLM-Bias[0][0]
cross_entropy (CrossEntropy)    (None, None, 14068)  0          Input-Token[0][0]
                                                                Input-Segment[0][0]
                                                                MLM-Activation[0][0]
==================================================================================================
Total params: 96,861,172
Trainable params: 96,861,172
Non-trainable params: 0
```

04 | 建议与总结

ROUGH评估

相关文章：ROUGE: A Package for Automatic Evaluation of Summaries (Chin-Yew Lin, 2004)

摘要：文章提出了一种面向召回率的评价方法 ROUGE（Recall-Oriented Understudy for Gisting Evaluation）。该方法可细分为 ROUGE-N, ROUGE-L, ROUGE-W 以及 ROUGE-S 四种评价指标。

ROUGE-N 指标计算生成的摘要与相应的参考摘要的 n-gram 召回率，具体的公式为：

$$ROUGE-N=\frac{\sum_{S\in ReferenceSummaries}\sum_{gram_n\in S}Count_{match}(gram_n)}{\sum_{S\in ReferenceSummaries}\sum_{gram_n\in S}Count_{(}gram_n)}$$

其中分母部分计算参考摘要中 n-gram 的个数，分子部分计算参考摘要和自动摘要共有的 n-gram 的个数。

问题背景
项目展示
算法设计与分析
改进与思考

BLEU

相关文章：BLEU: a Method for Automatic Evaluation of Machine Translation (Kishore Papineni, 2002)

摘要：文章提出了一种评估机器翻译结果质量的方法 BLEU（Bilingual Evaluation Understudy）。

BLEU 指标先计算生成翻译与相应的参考翻译的 n-gram 精确率，具体的公式为：

$$p_n = \frac{\sum_{C \in Candidate} \sum_{gram_n \in C} Count_{clip}(gram_n)}{\sum_{C' \in Candidate} \sum_{gram_n \in C'} Count_{(}gram_n)}$$

其中分母部分计算生成翻译中 n-gram 的个数，分子部分计算参考摘要和自动摘要共有的 n-gram 的个数。

然后再计算具体的 BLEU 值：

$$BLEU = BP * exp(\sum_{n=1}^{N} w_n log p_n)$$

问题背景
项目展示
算法设计与分析
改进与思考

最终目前的评估得分是：

Rouge-1： 0.3

Rouge-2： 0.19

Rouge-l： 0.28

Bleu： 0.13

```
metrics = evaluator.evaluate(test_data)  # 评测模型
print('valid_data:', metrics)
```

✓ 201m 17.8s

```
100%|██████████| 1679/1679 [3:21:17<00:00,  7.19s/it]

valid_data: {'rouge-1': 0.3041298785856503, 'rouge-2': 0.19703214401602984, 'rouge-l': 0.2872118283372123, 'bleu':
0.13044759471052714}
```

模型不足

1. 运行时间太慢，长文本先做降维处理，导致过程繁琐，后续模型bert虽然轻量化可时间复杂度高，以本电脑轻薄本为例子，无GPU状态供电，预测一个需要8-20s，而在不供电（轻薄本自动降频），需要1m20s（难以忍受）。
2. 模型目前rouge-1为0.3，虽然效果良好，但未达到最优，仍然有优化空间，扩大算力和训练次数，增加回调函数可以使模型更优。
3. 可以引入Copy等机制提高模型的rouge得分。

```python
def evaluate(self, data, topk=1):
    total = 0
    res = []
    rouge_1, rouge_2, rouge_l, bleu = 0, 0, 0, 0
    for title, content in tqdm(data):
        total += 1
        title = ' '.join(title).lower()
        pred_title = ' '.join(autotitle.generate(content, topk)).lower()
        res.append(pred_title)
        if pred_title.strip():
            scores = self.rouge.get_scores(hyps=pred_title, refs=title)
            rouge_1 += scores[0]['rouge-1']['f']
            rouge_2 += scores[0]['rouge-2']['f']
            rouge_l += scores[0]['rouge-l']['f']
            bleu += sentence_bleu(
                references=[title.split(' ')],
                hypothesis=pred_title.split(' '),
                smoothing_function=self.smooth
            )
    pd.DataFrame(res).to_csv('res3.csv', index=None)
    rouge_1 /= total
    rouge_2 /= total
    rouge_l /= total
    bleu /= total
    return {
        'rouge-1': rouge_1,
        'rouge-2': rouge_2,
        'rouge-l': rouge_l,
        'bleu': bleu,
    }
```

中国矿业大学
CHINA UNIVERSITY OF MINING AND TECHNOLOGY

基于Layui+Django的前后端设计

——以2022年软件杯智能创作平台为例

计算机科学与技术学院　计算机科学与技术2019-04班
学号：06192081　06192103
答辩人：胡钧耀　黄凯

目 录

CONTENTS

01 / 项目概述

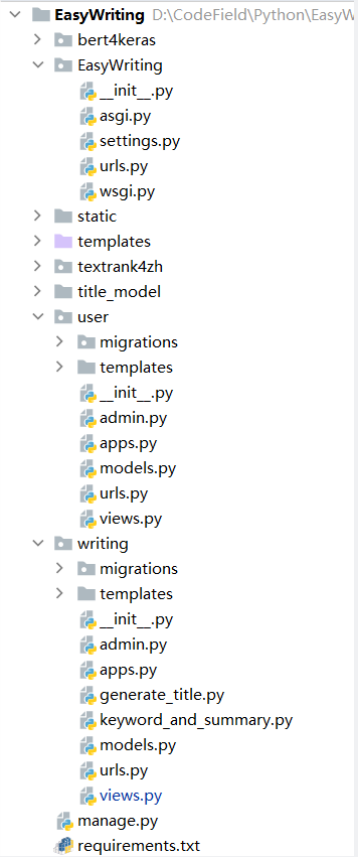
02 / 用户应用

03 / 写作台应用

04 / 总结与展望

01 | 项目概述

项目概述
用户应用
写作台应用
总结与展望



Django框架

1. 功能完善、要素齐全：自带大量常用工具和框架
2. 强大的数据库访问组件：Django的Model层自带数据库ORM (Object Relational Mapping) 组件
3. Django先进的App设计理念: App是可插拔的，是不可多得的思想。
4. 自带后台管理系统 admin：只需要通过简单的几行配置和代码就可以实现一个完整的后台数据管理控制平台。

项目概述
用户应用
写作台应用
总结与展望

开源模块化前端 UI 框架——layui

一套开源的 Web UI 解决方案，采用自身经典的模块化规范，并遵循原生 HTML/CSS/JS 的开发方式，极易上手，拿来即用。

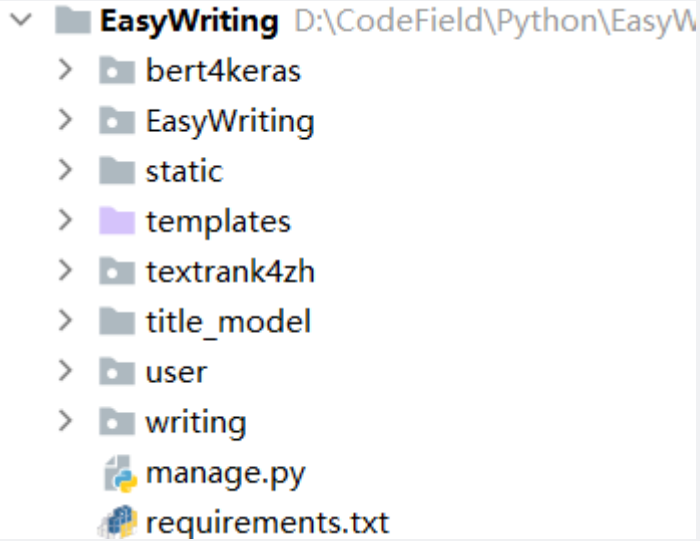
项目概述
用户应用
写作台应用
总结与展望

项目整体结构

bert4keras：*bert*训练模型
textrank4zh：*textrank*训练模型
title_model：预训练模型参数

EasyWriting：项目整体配置
static：静态资源
templates：模板

user：用户应用
writing：写作应用



注：在*Django*中，将**不同的功能**的**前端页面与后端交互**放在同一个包中，将之称为应用。例如*user*包中包含了用户的登录注册页面以及登陆注册的后端交互功能。这一点与*JavaWeb*不同，*JavaWeb*中将所有的前端页面放在一个位置，后端的*servlet*放在同一个位置。

项目概述
用户应用
写作台应用
总结与展望

项目整体配置 setting.py

*EasyWriting*文件夹保存了项目的整体配置
其中最重要的是*settings.py*和*urls.py*

*settings.py*文件中规定了根目录，调试模式，可允许访问的*IP*地址，项目安装的应用以及与*Mysql*数据库的连接等。

```
DEBUG = True
ALLOWED_HOSTS = ['*']

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'writing.apps.WritingConfig',
    'user.apps.UserConfig'
]
```

项目概述
用户应用
写作台应用
总结与展望

项目整体配置 setting.py

*EasyWriting*文件夹保存了项目的整体配置
其中最重要的是*settings.py*和*urls.py*

*settings.py*文件中规定了根目录，调试模式，可允许访问的*IP*地址，项目安装的应用以及与*Mysql*数据库的连接等。

```
MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    # 'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    # 'django.middleware.clickjacking.XFrameOptionsMiddleware',
]
```

项目概述
用户应用
写作台应用
总结与展望

项目整体配置 setting.py

*EasyWriting*文件夹保存了项目的整体配置
其中最重要的是*settings.py*和*urls.py*

*settings.py*文件中规定了根目录，调试模式，可允许访问的*IP*地址，项目安装的应用以及与*Mysql*数据库的连接等。

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
      'NAME': 'writing',
      'USER': 'root',
      'PASSWORD':  ‘**************',
      'HOST': '121.37.68.41',
      'PORT': '3306',
   }
}
```

项目概述
用户应用
写作台应用
总结与展望

项目整体配置 urls.py

*EasyWriting*文件夹保存了项目的整体配置
其中最重要的是*settings.py*和*urls.py*

相当于根目录*urls.py*，与其他应用的*urls.py*形成一个树，进行路径搜索

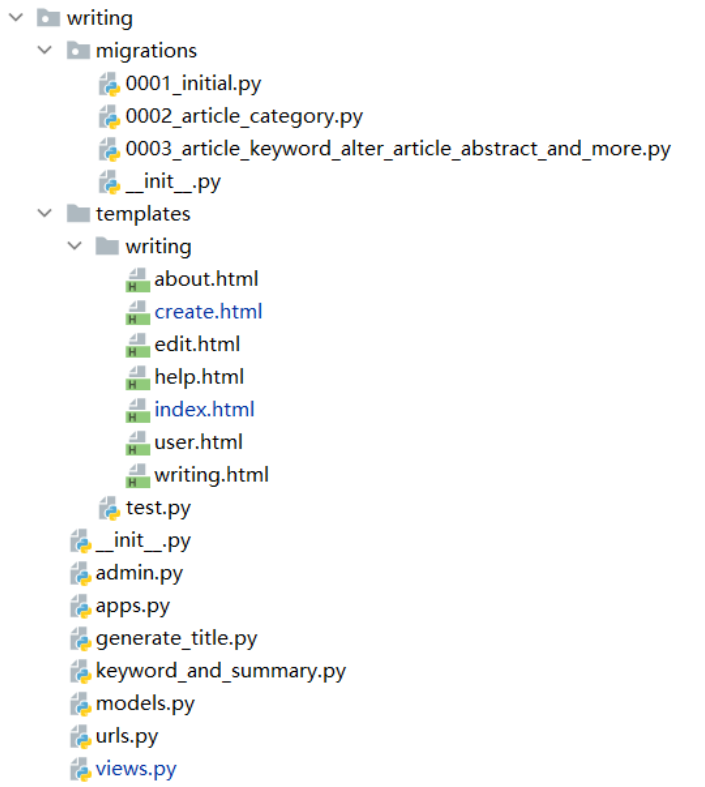
```
from django.contrib import admin
from django.urls import include, path


# 注意要引入自己的 views

urlpatterns = [
    path('', include('user.urls')),
    path('admin/', admin.site.urls),
    path('writing/', include('writing.urls')),
]
```

项目概述
用户应用
写作台应用
总结与展望

Django应用的模板

migrations：增量式数据迁移，（首次执行）根据*models*生成相应的迁移文件。（再次执行）根据*models*和已有迁移文件差别，生成新的迁移文件。

templates：应用的*html*页面
apps.py：应用的配置
models.py：应用需要创建的数据库
urls.py：每个页面中对应的*views.py*中的方法
views.py：逻辑处理，相当与*JavaWeb*中的*Servlet*



02 | 用户应用

项目整体结构
用户应用
写作台应用
总结与展望

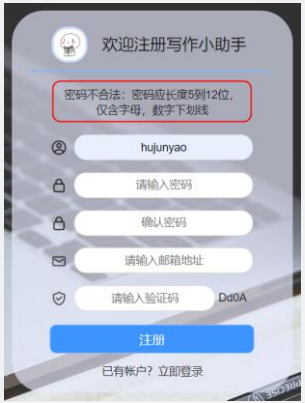
登录页面

登陆界面如上图所示，在用户点击登陆之后会通过js脚本检查用户名和密码是否符合规范，若符合，则查找数据库中的用户数据。

如果用户数据不匹配，则会将错误msg保存到scope域（locals()）中，并在页面中显示。

项目整体结构
用户应用
写作台应用
总结与展望

注册页面



如果用户数据不匹配，则会将错误msg保存到scope域（locals()）中，并在页面中显示。

项目整体结构
用户应用
写作台应用
总结与展望

登陆后端交互

```
def loginView(request):
    if request.user.is_authenticated:
        return redirect("/writing")
    else:
        if request.method == "POST":
            username = request.POST.get("username")
            password = request.POST.get("password")
            if User.objects.filter(username=username):
                user = authenticate(username=username, password=password)
                if user:
                    login(request, user)
                    request.session['username'] = username
                    msg = "登录成功"
                    # print(request.session.get('username'))
                    request.session.set_expiry(86400)
                    # 保存一天的session
                    return redirect("/writing")
                else:
                    msg = "用户名密码错误，或被限制登录"
            else:
                msg = "用户名不存在"
    return render(request, "user/login.html", locals())
```

项目整体结构
用户应用
写作台应用
总结与展望

注册后端交互

```
def registerView(request):
    if request.method == "POST":
        username = request.POST.get("username")
        password = request.POST.get("password")
        email = request.POST.get("email")
        if User.objects.filter(username=username):
            msg = "用户名已存在"
        else:
            user = User.objects.create_user(username=username,
password=password, email=email)
            msg = "注册成功"
            return redirect("/login/")
    return render(request, "user/register.html", locals())
```

注销后端交互

```
def logoutView(request):
    logout(request)
    return redirect('user:login')
```

03 | 写作台应用

数据库和服务器均已部署，*http://121.36.220.171:8001/*

项目整体结构
用户应用
写作台应用
总结与展望

Article模型

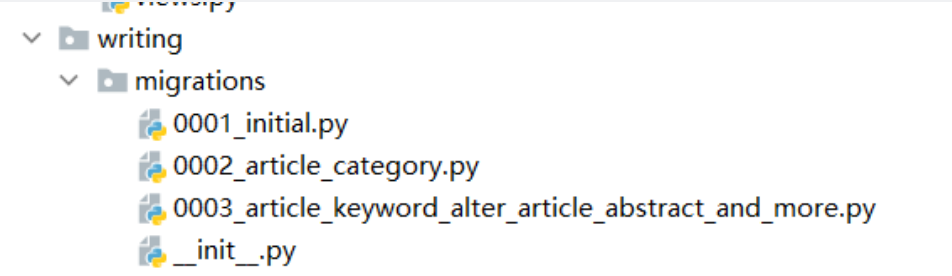
	id	t...	a...	body	au...	cre...	updat...		keyword
38	56	谈出待就业毕	并且政府的	<p>据报道，当地时间5月2	hujunyao	2022-06-20…	2022-06-20 2…	国际	待就，美国
39	57	加强农业人才	基层党组织要	<p>在3月6日下午的政协联	hujunyao	2022-06-20…	2022-06-20 2…	经济	人才工作,人才队伍建设,农业人才,农业
40	58	帮助高校毕业	近日，“高校	<p>就业稳，人心就稳，社	lichunyang	2022-06-20…	2022-06-20 2…	党建	高校毕业生就业,高校毕业生,毕业生就业
41	59	让党徽铭刻于	为使毕业生党	<p style="text-ali…	hujunyao	2022-06-21…	2022-06-21 0…	党建	毕业生党员
42	60	在这场美好的	座谈会上，同	<p style="text-ali…	hujunyao	2022-06-21…	2022-06-21 0…	党建	毕业生
43	61	践行师德师风	报告会上，龚	<p style="text-ali…	hujunyao	2022-06-21…	2022-06-21 0…	党建	师德师风
44	62	巴蜀文化厚植	日前，四川广	<p>日前，四川广汉三星堆	hujunyao	2022-06-21…	2022-06-21 0…	党建	考古发现,蜀文化,巴蜀地区,考古发掘,蜀
45	63	智能填报志愿	全国本专科院	<p>近些天，伴随着高考填	hujunyao	2022-06-21…	2022-06-21 0…	文化	志愿填报,填报志愿,高考志愿填报,大数
46	64	以材料作文应	与往年的高考	<p>2022年高考今日开考。	hujunyao	2022-06-21…	2022-06-21 0…	党建	结合材料,语文作文,高考作文题,材料作
47	65	这张智慧志愿	全国本专科院	<p>近些天，伴随着高考填	hujunyao	2022-06-21…	2022-06-21 0…	科技	高考志愿填报,大数据,填报志愿,志愿填
48	66	博物馆教育应	从发挥教育功	<p>为持续推进我国博物馆	hujunyao	2022-06-21…	2022-06-21 0…	党建	博物馆发展,实现博物馆,突出博物馆,博
49	67	让花茂村脱贫	2021年2月，	<p>党的十八大以来，习近	hujunyao	2022-06-21…	2022-06-21 1…	经济	推进乡村,习近平总书记,创新发展
50	68	把握好acm大	在学校领导的	<p><span style="te…	hujunyao	2022-06-21…	2022-06-21 1…	文化	acm
51	69	计算机科学与	学院大力发展	<p>计算机科学与技	hujunyao	2022-06-21…	2022-06-21 1…	文化	技术专业,网络实验室
52	70	把寒假时光用	她首先了解同	<p><span style="te…	hujunyao	2022-06-21…	2022-06-21 1…	文化	留校学生
53	71	用行政优势助	计算机学院院	<p><span style="te…	hujunyao	2022-06-21…	2022-06-21 1…	文化	深信服,计算机学院,战略合作
54	72	用行政优势助	计算机学院院	<p>北方持续高温，局地气	hujunyao	2022-06-21…	2022-06-21 1…	文化	深信服,计算机学院,战略合作
55	73	高温预警要特	应急管理部防	<p>北方持续高温，局地气	hujunyao	2022-06-21…	2022-06-21 1…	经济	出现龙卷,西江干流,高温天气,出现高温
56	74	让乡村建设过	他强调，要深	<p>全国乡村建设工作会议2	hujunyao	2022-06-21…	2022-06-21 1…	经济	推进乡村,乡村建设,农村基本,推进乡村

项目整体结构
用户应用
写作台应用
总结与展望

Article模型

```
python manage.py makemigrations
python manage.py migrate
```

```
class Article(models.Model):
    title = models.CharField(max_length=50)
    abstract = models.TextField(max_length=300)  # 摘要
    body = models.TextField()  # 正文（因为文章的正文字数使用TextField）
    keyword = models.CharField(max_length=40)
    category = models.CharField(max_length=5)  # 类型
    author = models.CharField(max_length=10)  # 作者
    create_time = models.DateTimeField(auto_now_add=True)  # 创建时间
    update_time = models.DateTimeField(auto_now=True)  # 更新时间
```



项目整体结构
用户应用
写作台应用
总结与展望

主页

框架：*django*，*layui*，可视化：*woldcloud*，*echarts*，爬虫：*requests*，*BeautifulSoup*

项目整体结构
用户应用
写作台应用
总结与展望

Echart可视化展示后端

以一周新建文章为例

```
@login_required(login_url='/login/')
def index(request):
    username = request.session.get('username')

    # 近七天的个人新建和全站新建文章数量统计
    data_user = []
    data_all = []
    for i in range(-6, 1):
        start = datetime.date.today() + datetime.timedelta(days=i)
        end = datetime.date.today() + datetime.timedelta(days=i + 1)
        data_user.append(Article.objects.filter(create_time__range=(start, end),
author=username).count())
        data_all.append(Article.objects.filter(create_time__range=(start, end)).count())
    list_recent_a_week = [(datetime.date.today() +
datetime.timedelta(days=i)).strftime('%m-%d') for i in range(-6, 1)]

    # 所有时间全站人一天中分时段活跃情况（创建文章数量）
    data_hour_all = []
    for i in range(0, 24, 2):
        data_hour_all.append(Article.objects.filter(create_time__hour=i).count())
……
```

项目整体结构
用户应用
写作台应用
总结与展望

Echart可视化展示前端

以一周新建文章为例

```
<div id="main" style="width: auto; height:300px;"></div>
<script type="text/javascript">
var myChart = echarts.init(document.getElementById('main'));
        var option = {title: {text: '近一周新建文章数量',},
        legend: {data: ['用户', '全站'],top: '2%',right: '15%'},
        xAxis: {data: {{ list_recent_a_week | safe}}},
        yAxis: [
            {type: 'value', name: '用户', position: 'left', alignTicks: true,},
            {type: 'value', name: '全站', position: 'right', alignTicks: true,}
        ],
        series: [
            {name: '用户', type: 'bar', data: {{ data_user | safe }}},
            {name: '全站', type: 'line', yAxisIndex: 1, data: {{ data_all | safe }}}
        ]
    }
   myChart.setOption(option);
</script>
```

项目整体结构
用户应用
写作台应用
总结与展望

Wordcloud云图后端

```
# 近七天所有人文章内容关键词统计云图，一天只绘制一个，减少计算量
today = datetime.date.today().strftime("%y%m%d")
static_root = "static_new"
filename = static_root + "/media/web/wordcloud/wordcloud_" + today + ".png"
if not os.path.exists(filename):
    start = datetime.date.today() - datetime.timedelta(days=7)
    end = datetime.date.today()
    all_text = list(Article.objects.filter(create_time__range=(start,
end)).values_list("keyword", flat=True))
    text = ','.join(all_text)

    wordcloud = WordCloud(
        background_color="white",
        max_words=50,
        width=1200,
        height=800,
        # mask=background,
        font_path= static_root + '/media/font/font.TTF'
    )
    wordcloud.generate(text)
    wordcloud.to_file(static_root + "/media/web/wordcloud/wordcloud_" +
datetime.date.today().strftime("%y%m%d") + ".png")
```

项目整体结构
用户应用
写作台应用
总结与展望

爬虫后端

以百度榜单为例

```
# 爬虫配置
kv = { ‘user-agent’ :  ‘Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) ’
                     ‘Chrome/102.0.5005.63 Safari/537.36 Edg/102.0.1245.30’ ,
       ‘cookie’ :  ‘*****'}
SEARCH_NUM = 20

# 百度
url = "https://top.baidu.com/board?tab=realtime"
selector = 'div.content_1YWBm'
response = requests.get(url, headers=kv).text
baidu = BeautifulSoup(response, features="lxml").select(selector + " > a")[:SEARCH_NUM]
baidu1 = BeautifulSoup(response, features="lxml").select(selector + " div.c-single-text-
ellipsis")[:SEARCH_NUM]
baidu_link_and_title = []
for i in range(SEARCH_NUM):
    baidu_link_and_title.append([baidu[i].get("href"), baidu1[i].get_text().strip()])
```

项目整体结构
用户应用
写作台应用
总结与展望

爬虫前端

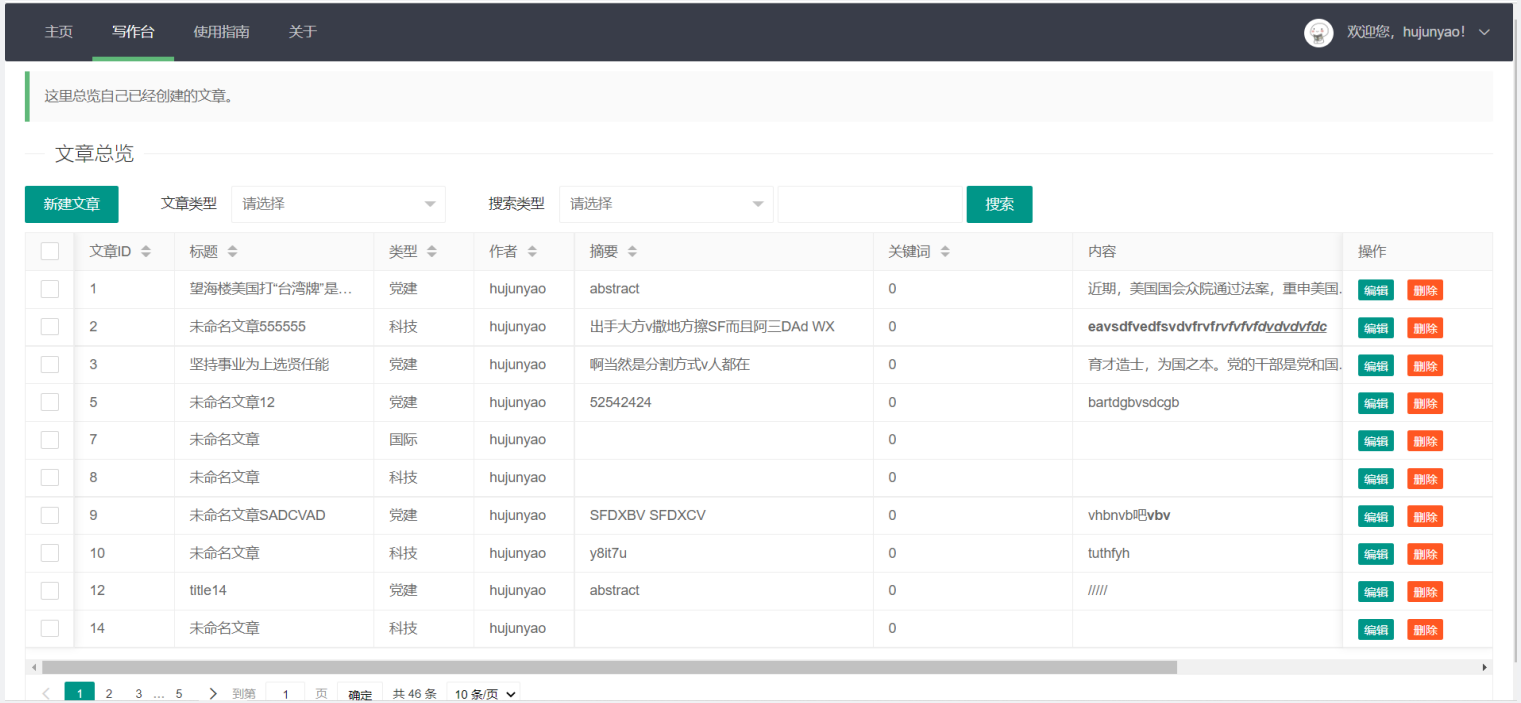
以百度榜单为例

```
<div class="layui-tab-content">
    <div class="layui-tab-item layui-show" style="height: 180px;overflow:auto;">
        <table class="layui-table" lay-even="" lay-skin="nob">
            <colgroup><col width="20"><col width="200"><col></colgroup>
            <thead><tr><th>排名</th><th>百度话题</th></tr></thead>
            <tbody>
            {% for baidu in baidu_link_and_title %}
                <tr>
                    <td>{{ forloop.counter }}</td>
                    <td>
                        <a href="{{ baidu.0 }} " target="_blank">{{ baidu.1 }}</a>
                    </td>
                </tr>
            {% endfor %}
            </tbody>
        </table>
    </div>
```

项目整体结构
用户应用
写作台应用
总结与展望

写作台总览

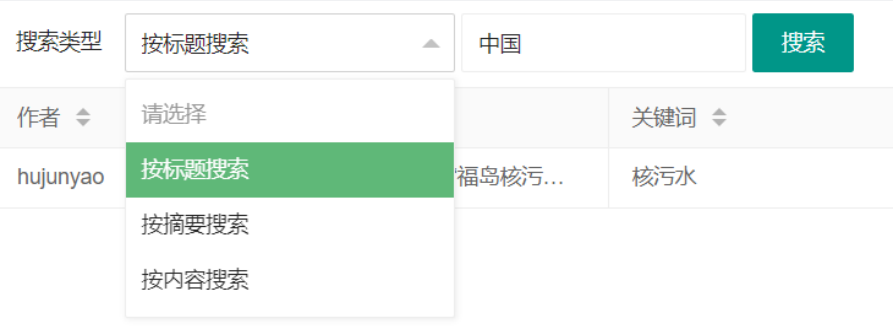
列表页面显示出了数据库中所有的文章，可以新建、查找、修改、删除，及对应数据库的增删改查，并可以分页显示。



项目整体结构
用户应用
写作台应用
总结与展望

写作台总览

文章查找
可以按照文章类型查找、搜索类型+搜索条件查找



项目整体结构
用户应用
写作台应用
总结与展望

写作台总览

文章查找
可以按照文章类型查找、搜索类型+搜索条件查找

```
@login_required(login_url='/login/')
def query(request):
    pageNo = int(request.POST.get("pageNo"))
    pageSize = int(request.POST.get("pageSize"))
    left = (pageNo - 1) * pageSize
    right = pageNo * pageSize
    type = request.POST.get("search_type")
    search = request.POST.get("search")
    category = request.POST.get("search_category")
    if not category:
        category = ""
    if type == 'title':
        article = Article.objects.filter(author=request.session.get('username'), category__contains=category,
                                         title__contains=search)
    elif type == 'abstract':
        article = Article.objects.filter(author=request.session.get('username'), category__contains=category,
                                         abstract__contains=search)
    elif type == 'body':
        article = Article.objects.filter(author=request.session.get('username'), category__contains=category,
                                         body__contains=search)
    else:
        article = Article.objects.filter(author=request.session.get('username'), category__contains=category)
    context = {"code": 0,"msg": "","count": article.count(),"data": list(article[left:right].values())}
    return JsonResponse(context)
```

项目整体结构
用户应用
写作台应用
总结与展望

写作台总览

文章查找

可以按照文章类型查找、搜索类型+搜索条件查找

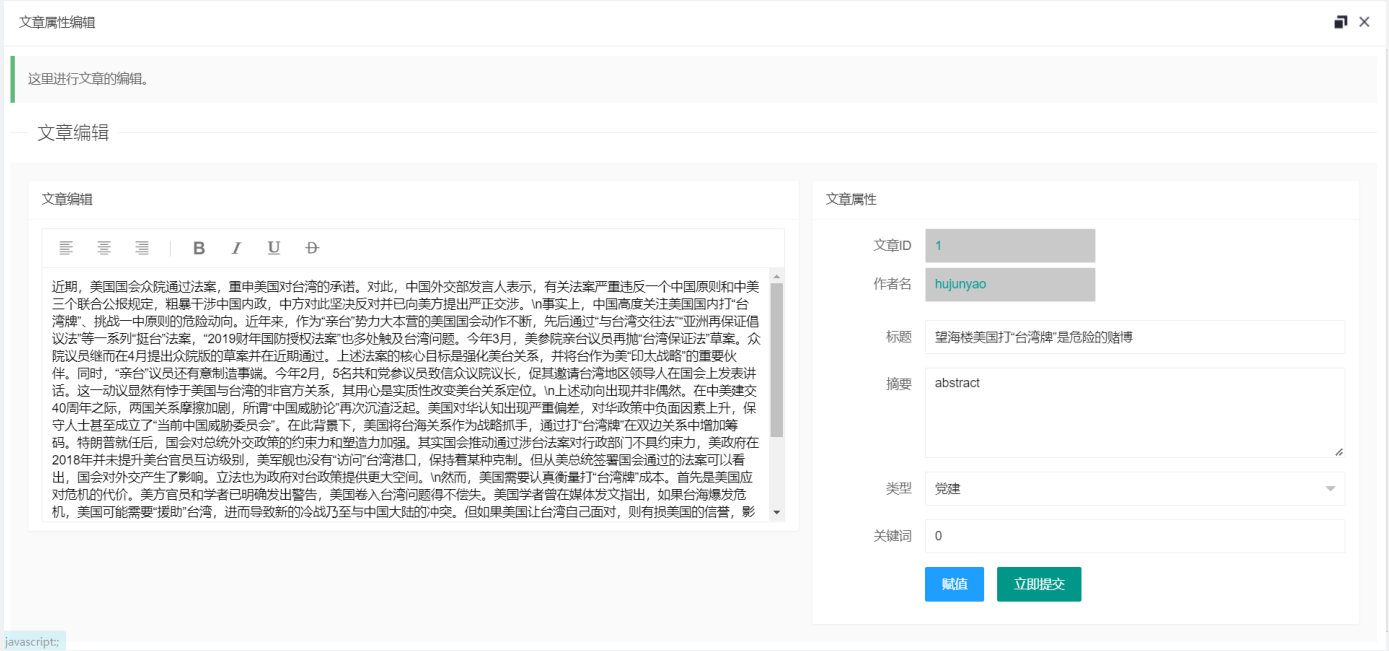
```html
<form action="" class="layui-form">
    {% csrf_token %}
    <div class="layui-inline">
        <label class="layui-form-label">文章类型</label>
        <div class="layui-input-block">
            <select name="search_category">
                <option value="" selected=""></option>
                <option value="党建">党建</option>
            </select>
        </div>
    </div>
    <div class="layui-inline">
        <label class="layui-form-label">搜索类型</label>
        <div class="layui-input-block">
            <select name="search_type">
                <option value="" selected=""></option>
                <option value="title">按标题搜索</option>
            </select>
        </div>
    </div>
    <div class="layui-inline">
        <div class="layui-input-inline">
            <input name="search" autocomplete="off" id="input_search" class="layui-input">
        </div>
        <button type="submit" class="layui-btn" id="search" lay-submit="" lay-filter="search">搜索
      </button>
    </div>
</form>
```

```python
form.on('submit(search)', function
(data) {
    table.reload('article', {
        url: '{% url
'writing:query'%}',
        method: "post",
        page: {curr: 1},
        where: data.field,
        request: {
            pageName: 'pageNo',  //
页码的参数名称，默认：page
            limitName: 'pageSize'
//每页数据量的参数名，默认：limit
        },
    })
    return false;
});
```

项目整体结构
用户应用
写作台应用
总结与展望

新建/编辑

文章新建、编辑
在数据库中以**富文本的形式**存储文章，
调用layui框架的文本编辑工具。



项目整体结构
用户应用
写作台应用
总结与展望

新建/编辑

```
//监听行工具事件
table.on('tool(article)', function (obj) {
    var data = obj.data, layer = layui.layer;
   if (obj.event === 'del') {} else if (obj.event === 'edit') {
        layer.open({
            type: 2, title: '文章属性编辑', maxmin: true,
            content: "../" + data.id, area: ['1000px', '620px']
            , cancel: function (index, layero) {table.reload('article');}
        });}});
```

```
@login_required(login_url= '/login/' )
def update(request):
    try:
        selected_article = Article.objects.get(pk=request.POST[ 'id' ])
        selected_article.abstract = request.POST[ 'abstract' ]
        selected_article.title = request.POST[ 'title' ]
        …
        selected_article.save()
        return HttpResponse("success")
    except:
        return HttpResponse("fail")
```

项目整体结构
用户应用
写作台应用
总结与展望

新建/编辑

```
form.on('submit(btn_edit)', function (data) {
    $.ajax({
        url: "{% url 'writing:add'%}",type: 'post',
        data: data.field,dataType: "text",
        success: function (result) {
            if (result == 'success') {layer.msg('提交成功', {icon: 1});}
            else {layer.msg('提交失败', {icon: 2});}}
    });
    return false;//阻止表单跳转
});
```

```
@login_required(login_url='/login/')
def add(request):
    try:
        Article.objects.create(title=request.POST.get('title ‘),…)
        return HttpResponse("success")
    except:
        return HttpResponse("fail")
```

项目整体结构
用户应用
写作台应用
总结与展望

删除

```
//监听行工具事件
table.on('tool(article)', function (obj) {
    var data = obj.data, layer = layui.layer;
    if (obj.event === 'del' ) {
    layer.confirm("确定删除这行数据吗？", function () {
    $.ajax({
        url: "{% url 'writing:delete' %}",method: "post",
        data: {'article_id': data.id},
        success: function (res) {
            if (res === "success") {layer.msg('删除成功', {icon: 1});obj.del();}
            else {layer.msg('删除失败', {icon: 2});}},
        async: false
    })layer.close();});} else if (obj.event ===  'edit' ) {…;}});
```

```
@login_required(login_url='/login/')
def delete(request):
    if request.GET.get('article_id'):
        Article.objects.filter(pk=request.GET.get('article_id')).delete()
        return HttpResponse("success")
    else: return HttpResponse("fail")
```

04 | 项目总结

问题背景
项目展示
改进与思考
项目总结

项目总结：

本次项目我们使用了textrank算法与NLP中的Bert_UniLM模型生成了文章的摘要、关键词以及标题，前端使用了Layui进行界面的搭建，同时利用ecahrt进行可视化的数据展示，后端使用Django框架进行逻辑处理。

整体上做到了标题、摘要、关键词的生成较为准确，前端的UI设计合理、美观，数据可视化程度高，系统运行流畅。对于题目的要求而言，完成度较高。主要是在时间方面效果不佳，没有算力，计算速度慢。

感谢各位老师同学们的观看

指导老师：张艳梅　　答辩人：胡钧耀　黄凯