

自然语言论文检索项目

(中期检查表见本文最后一页)

项目组成人员：诸欣扬、赖杨丽多、张晗、管心妍

中期项目的结构导图：



一、 概述：

我们的检索项目主要分为两个部分。首先是前后端的数据连接，我们完成了网页制作，网页前后端数据已经可以成功连接并实现模糊搜索功能。我们制作了主界面、搜索结果界面、论文信息界面三级网页，使用爬虫爬取了东南大学硕博论文并进行了初步数据分类，并使用爬虫所得结果构建了具体的搜索数据库，成功实现了前后端数据连接。然后是人工智能部分，我们进行了机器学习主流算法的原理学习和理解，初步了解了自然语言处理的概念，对 Seq2Seq 进行了学习和数据集的导入

二、 各部分工作详细汇报：

人工智能部分：

1.人工智能部分介绍

项目的人工智能部分主要为**基于语义解析的知识库问答**和**基于 BERT 模型的细粒度搜索**。

知识库中储存着各个实体及其关系，是**知识库问答**的数据基础。知识库问答的主要方法可以分为基于信息提取的方法、基于语义解析的方法和基于向量空间建模的方法三类。我们采取的方法为**基于语义解析**的方法。语义解析的目的在于将自然语言问句解析成为计算机能够理解的逻辑形式，再将这些逻辑形式转换成计算机能够理解的检索语句以进行检索**错误!未找到引用源。**。若直接使用模型将自然语言问句解析成为检索语句，得到的检索语句可能存在错误，不能直接进入搜索引擎进行检索，故，我们将自然语言问句首先解析成 lambda-表达式表示的逻辑形式，然后再将逻辑形式翻译成对应的检索语句，这样可以保证检索语句不会出现语法错误而导致查询失败。在进行语义解析时，现阶段使用较多的方法是基于深度学习和基于预训练模型的方法。我们将自然语言问句解析成逻辑形式，是从序列到序列的任务，类似于翻译[2]，故使用在处理序列问题上有很好表现的 Seq2Seq 模型来完成此任务。

细粒度搜索是指，搜索引擎可以对文章的内容进行理解并寻求搜索的结果。就本文的论文检索系统而言，细粒度搜索即

检索的结果不仅仅局限于知识库的各实体之间的关系或各实体的属性，还可以来自于其属性的内容中，如：摘要的内容。我们采用 BERT 模型来搭建可以对文本内容进行阅读理解模型，从而实现对论文的内容进行细粒度搜索。

我们的问答系统支持**自然语言问句解析**，即用户可以用自然语言问句进行提问，这是论文检索系统的一大创新；此外，我们还利用**KBQA-BERT 进行知识库问答**，对检索信息进行提取，使得用户在获取相关论文的同时可以一眼看到关键学术信息，从而有针对性地寻找论文。这避免了传统的关键词检索配对带来的局限性和低效率。

2. 调研成果：主要自然语言处理模型原理即项目中功能介绍

2.1 Seq2Seq 模型

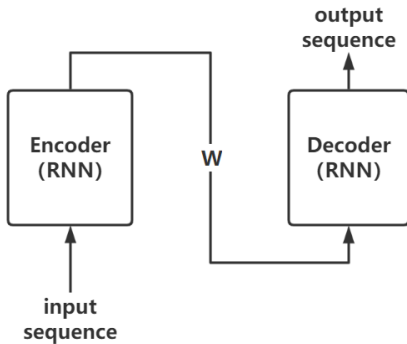


图 2-1 Seq2Seq 模型结构示意图

Seq2Seq 模型[3]是谷歌在 2014 年提出的一种 Encoder-Decoder 结构的模型，它可以将一个序列映射成另一个序列，该模型在处理机器翻译、问答系统、语音识别等问题上有很好的表现。Seq2Seq 模型的结构如图 2-1 所示，其编码器将输入的序列编码成固定长度的向量，该向量是富含序列的语义信息的表示，解码器则将编码器得到的语义表示解码成一个长度不固定的序列，其编码器和解码器都通过循环神经网络（RNN）模型来实现，可以使用 LSTM 和 GRU 模型。具体而言，在编码器中，一次将序列输入模型中，上一个字符的隐藏状态和下一个字符一起作为下一次的输入，并得到一个固定长度的向量 W。对于解码器来说，编码器的最后一个隐藏状态和向量 W 作为解码器的输入解码后得到对应的输出字符，当解码到停止符 EOS 时，解码结束。

在只使用 Seq2Seq 模型时，效果并不能达到理想状态，为了使其有更好的效果便引入了 Attention 机制。Attention 机制能够将注意力放在关键元素上，比如在将导师指导学生的关系解析成逻辑形式时，则注意句子中有关导师和学生的部分。引入 Attention 机制后的 Seq2Seq 模型，其主要变化在解码器部分。将 Attention 机制引入到 Seq2Seq 模型中时，编码器的每一个隐藏状态都保存下来，在处理每一个字符时，使用该字符及该字符之前的序列对应的隐藏状态来生成一个新的语义表示，而不是向前文所述编码器只生成一个语义表示。在解码时，Attention 则会对每一个自然语言的字符赋予权值，权值越大则说明在此时该字符越重要。

在进行自然语言问句解析之前，首先要对问句进行命名实体识别，也就是要对问句内容分析得到与知识库有关的信息——该问句涉及哪些实体。命名实体的方法大致可以分为基于规则和词典的方法、基于统计机器学习的方法和基于深度学习的方法**错误!未找到引用源。**。我们的各实体关系较为简单，提问方式也有明显的规律可循，所以此处采用的命名实体识别方式是基于规则和 LAC 工具来实现的。LAC（Lexical Analysis of Chinese）**错误!未找到引用源。**是百度自然语言处理部研发的一款联合的词法分析工具，可以实现中文分词、词性标注、专名识别等功能。在进行命名实体识别时，除了 LAC 能够识别的人名、时间、地点、机构外，还可以添加词典使其能够识别出本知识库中的实体，通过添加词典，可以使 LAC 识别出人名、年份、学校、学科、关键词、论文。而在判断人名究竟是老师还是作者时，则可以根据规则匹配来进行识别。在对某导师提问时，问句中的人名前后通常会有指导、学生、导师、老师等字样。其规则可以设定如下，其中 per 表示识别出来的人名字符串：

```

r"[w|\W]*[[导师|老师]%s"%(per)
```

$r "[w|W]^* [s[w|W]]_{(0,6)} [\text{指导}|\text{参加指导}|\text{参加工作}][w|W]^* [\text{学生}|\text{论文}]^* \% (\text{per})$

$r "[w|W]^* [w|W]^* [s[w|W]]_{(0,6)} [\text{学生}|\text{论文}]^* \% (\text{per})$

$r "[w|W]^* [[w|W]^* [s[w|W]]_{(0,6)} [\text{指导}|\text{教导}][w|W]]_{(0,2)} [\text{学生}|\text{论文}]^* \% (\text{per})$

若没有识别出实体,则在检索时直接将原始问题输入到检索语句中,对摘要内容进行匹配检索。在识别出命名实体之后,则可以对问句进行解析。具体的解析步骤如下所示:

step1. 命名实体识别,将识别出的命名实体替换成对应的实体名称得到标准问句,并将提取出来命名实体进行保存,如对于问题“自 2014 年到 2018 年,张三导师指导过哪些文章?”,识别后得到问句模板“自[年份 0]到[年份 1],[导师 0]指导过哪些文章?”,以及对应的实体列表[2014 年, 2018 年, 张三]。

step2. 将标准问句输入到解析模型进行解析,解析后得到对应的逻辑形式,上述问题解析为:

$\lambda a. \exists x(\text{论文}(a) \wedge \text{导师}([\text{导师}0]) \wedge \text{年份}(x) \wedge \text{年份}([\text{年份}0]) \wedge \text{年份}([\text{年份}1])$
 $\wedge \text{导师_}(a, [\text{导师}0]) \wedge \text{年份_}(a, x) \wedge \leq(x, [\text{年份}1]) \wedge \geq(x, [\text{年份}0]))$

step3. 将逻辑形式翻译成 ES 检索语句。

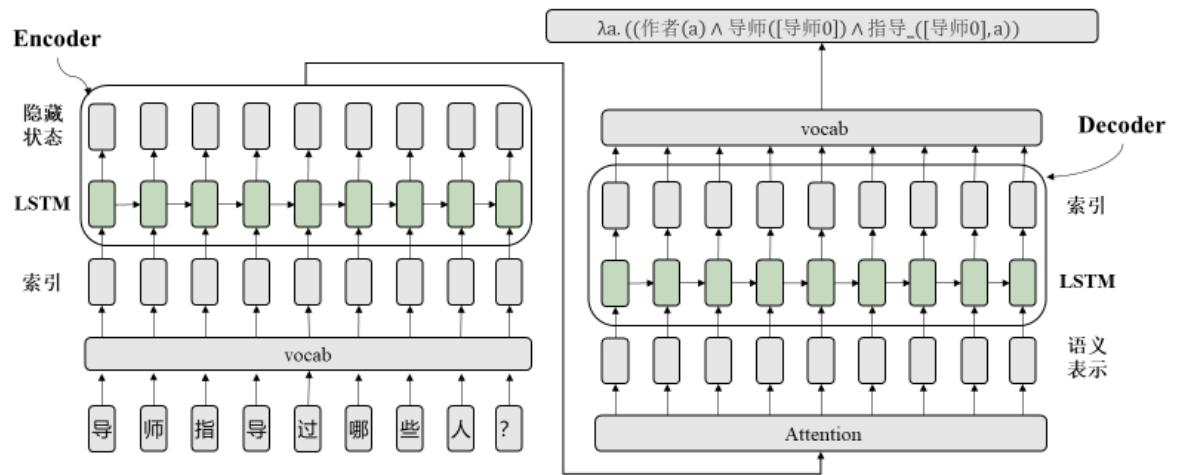


图 4-1 解析模型结构示意图

我们的解析模型如图 4-1 所示,是采用加入了 Attention 机制的、基于 LSTM 的 Seq2seq 模型来构建的。最初尝试将 BERT 模型作为编码器来实现该模型,但是本文的逻辑形式-问句模板对太少,而 BERT 模型的参数比较多,并不能够达到理想的效果。在此使用专门处理序列到序列问题的 Seq2Seq 模型的结果会更好,从逻辑形式到问句本身也是一个序列到序列的问题,类似于翻译[2],在机器翻译上有着出色表现的 Seq2Seq 模型来说当然可以适用于该解析任务。解析模型的 Seq2Seq 模型的编码器和解码器都是基于 LSTM 模型实现的。在实现解析时,首先要有一个词汇和索引相对应的词汇表,通过该词汇表,可以将字符转换成独一无二的索引,从而可以实现字符与索引间的一一对应。在将问句输入到模型时,需要将问句的字符转化成索引,然后 Seq2Seq 的编码器对输入进行编码,解码后 Attention 机制对每一个隐藏状态及其以前的隐藏状态通过加权形成语义表示来作为解码器的输入,每一个输入都是不同的语义表示,解码器会将其解码成对应逻辑形式的索引,然后再次通过词汇表即可得到对应的逻辑形式。

2.3 BERT 模型

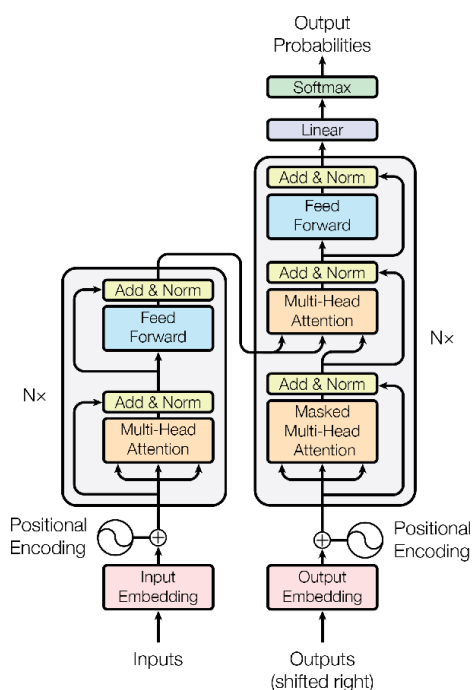


图 2-2 Transformer 模型结构示意图错误!未找到引用源。

BERT 模型(Bidirectional Encoder Representations from Transformer)错误!未找到引用源。是由谷歌提出的预训练语言模型，是基于 Transformer 模型错误!未找到引用源。的双向的编码器，它可以将语句编码成包含丰富的语义信息的表示。将文本的语义表示在特定自然语言处理的任务中作微调，可以将 BERT 模型应用于该项特定任务中，BERT 模型在阅读理解、文本分类等诸多 NLP 任务上都表现非常出色。BERT 模型是基于 Transformer 模型得到的，Transformer 模型是谷歌公司在之前提出的 NLP 模型，其结构如图 2-2 所示，它也是 Encoder-Decoder 结构型的模型，但与前文提到的 Seq2Seq 模型不同的是，Transformer 模型用 Attention 代替了 RNN 模型，加入了 position embedding 来考虑序列中各字符的位置关系。BERT 模型利用了 Transformer 的 encoder 部分，在训练时，引入了两个无监督的任务，分别为遮挡语言模型（Masked Language Model）和下一句预测任务（Next Sentence Prediction），来使预训练过的 BERT 模型具有语义表示的功能。

BERT 模型是在大规模的、与特定 NLP 任务无关的文本语料上进行训练后得到的语言模型，将语句输入到 BERT 模型中可以得到该语句富含语义信息的表示。在完成阅读理解这个任务时，需要对预训练的 BERT 模型进行微调，使其可以基于语义表示继续得到阅读理解任务的答案。

在进行阅读理解时，我们采用的是抽取式机器阅读理解，即从阅读文本中寻找该问题的答案对应的片段，显然，该问题的答案在阅读文本中的连续位置上，那么阅读理解模型想要找到问题答案只需从阅读文本中寻找答案所在的位置即可。故，阅读理解模型应该根据输入的问题和阅读文本，判断出答案在阅读文本中的起止位置。其本质上是一个分类问题，模型需要判断每一个字符所在的位置是不是答案的起止位置。我们的阅读理解模型如图 4-3 所示。将问题和阅读文本作为模型的输入，在处理中文文本时，以一个字符为最小单位。通过 BERT 模型对语句进行编码得到输入的每个字符的语义信息，对 BERT 模型的最后一层进行分类，判断该位置是否为答案的起止位置，最后根据预测的起止位置即可得到阅读理解的答案。

在对自然语言进行处理时，不能直接将每一个字符作为模型的输入，模型也不能直接得到对应的文本，需要将字符映射成计算机能够读取的形式，这里是通过 BERT 的词汇表得到对应字符的索引，词汇表长度为 21128，包括中文字符、英语字母、特殊字符等，每个字符与其索引一一对应。在模型预测结束后，根据预测出的答案在文本中的起始位置，得到输入模型的索引形式的语句的对应段落，便得到了问题的答案片段。

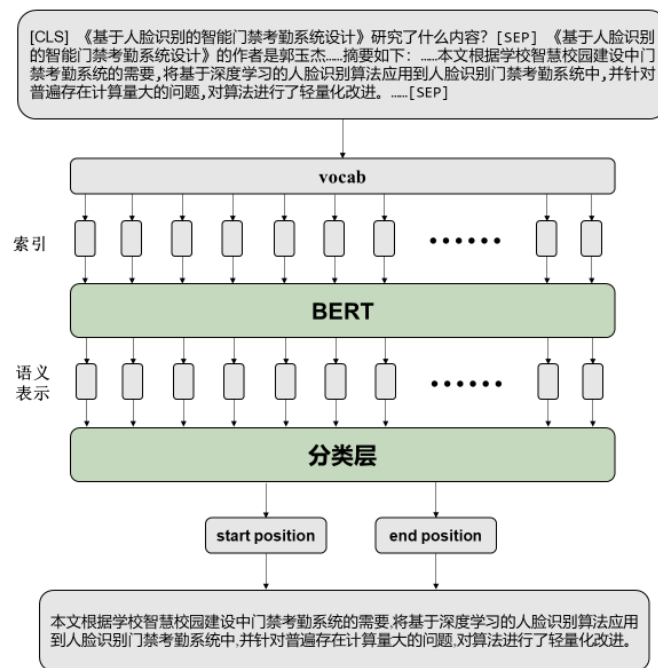
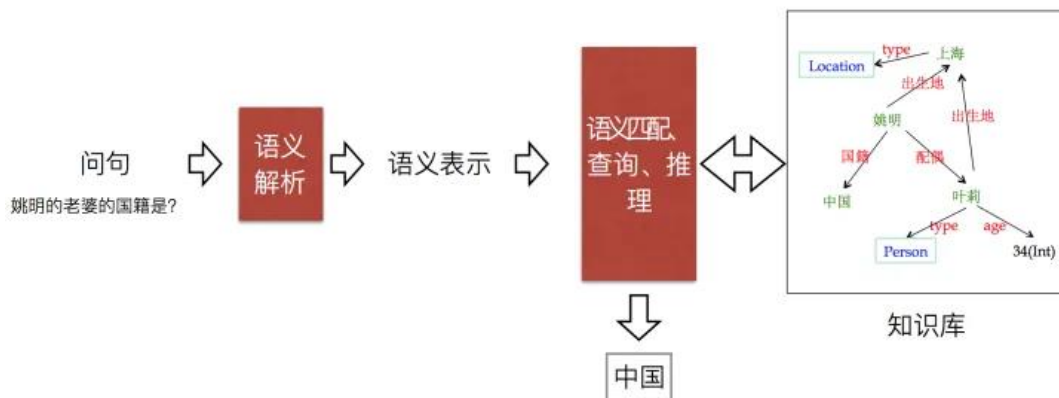


图 4-3 阅读理解模型结构示意图

在面对具体的文本时，有可能会出现阅读文本过长的情况，导致其长度超过了 BERT 模型指定的最大文本长度。此时，最粗略的方法是，按照最大的长度对文本进行分割，对分割后得到的每一段分别进行阅读理解，判断答案所在的位置。但是，当分割点在答案的起止位置之间时，会打断答案的完整性，这样的做法明显不可取。我们采取的方式是设置一个固定长度的窗口对过长的文本进行滑动，滑动时的步长应该是小于窗口长度的，这样可以保证窗口滑动时不遗漏所有的文本内容，并且能够将上一个窗口内容的一定长度的尾部内容作为该窗口内容的开头。窗口按照设定好的步长滑动一次便得到一个子文本，窗口滑动完成后便得到了过长阅读文本的所有子文本。将这些子文本分别和问题相结合便得到了一个子样本。在训练集中，一个子样本就是一个训练样本；在测试集中，在得到一个问题的所有子样本的预测结果后，需要综合考虑后得到一个最佳答案。

知识库问答（knowledge base question answering,KB-QA）即给定自然语言问题，通过对问题进行语义理解和解析，进而利用知识库进行查询、推理得出答案。如下图所示：



与对话系统、对话机器人的交互式对话不同，KB-QA 具有以下特点：

答案：回答的答案是知识库中的实体或实体关系，或者 no-answer（即该问题在 KB 中找不到答案），当然这里答案不一定唯一，比如 某导师指导的论文有哪些 。而对话系统则回复的是自然语言句子，有时甚至需要考虑上下文语境。

评价标准：回召率 (Recall)，精确率 (Precision)，F1-Score。而对话系统的评价标准以人工评价为主，以及 BLEU 和 Perplexity。

知识库问答的主流方法可以分为三类：

语义解析 (Semantic Parsing)：该方法是一种偏 linguistic 的方法，主体思想是将自然语言转化为一系列形式化的逻辑形式 (logic form)，通过对逻辑形式进行自底向上的解析，得到一种可以表达整个问题语义的逻辑形式，通过相应的查询语句（类似 lambda-Calculus）在知识库中进行查询，从而得出答案。

信息抽取 (Information Extraction)：该类方法通过提取问题中的实体，通过在知识库中查询该实体可以得到以该实体节点为中心的知识库子图，子图中的每一个节点或边都可以作为候选答案，通过观察问题依据某些规则或模板进行信息抽取，得到问题特征向量，建立分类器通过输入问题特征向量对候选答案进行筛选，从而得出最终答案。

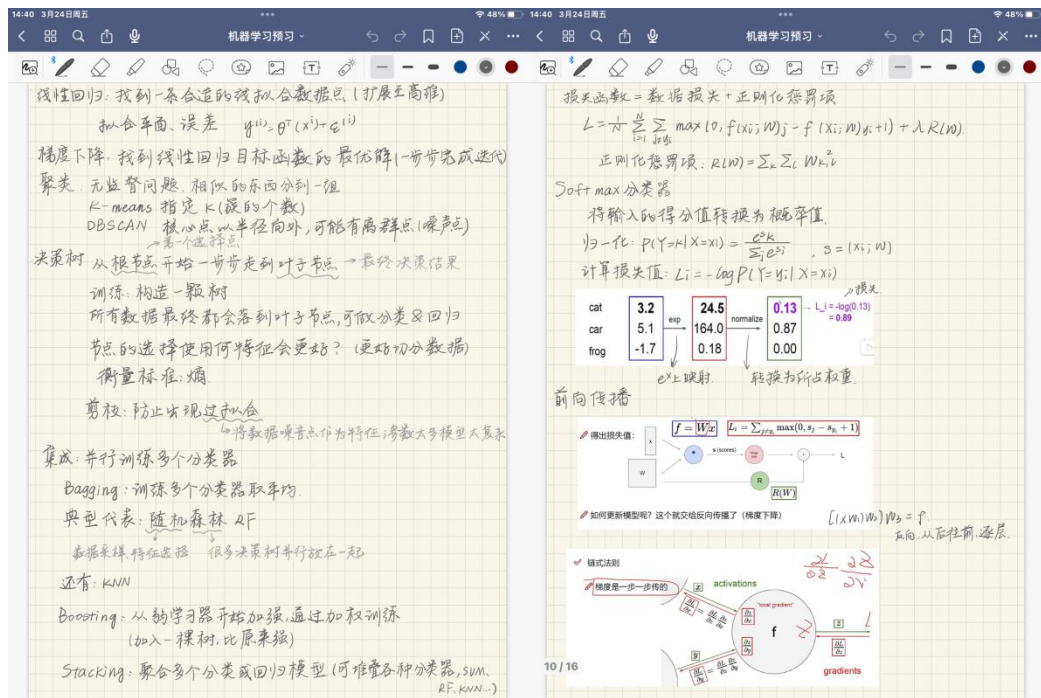
向量建模 (Vector Modeling)：该方法思想和信息抽取的思想比较接近，根据问题得出候选答案，把问题和候选答案都映射为分布式表达 (Distributed Embedding)，通过训练数据对该分布式表达进行训练，使得问题和正确答案的向量表达的得分（通常以点乘为形式）尽量高，如下图所示。模型训练完成后则可根据候选答案的向量表达和问题表达的得分进行筛选，得出最终答案。

我们项目采用的是语义解析的形式，在使用 Seq2Seq 对自然语言问句进行解析之后，将提取的关键信息返回 BERT 模型阅读理解生成的数据库中进行检索，对内容进行提取之后返回结果，呈现在用户界面。

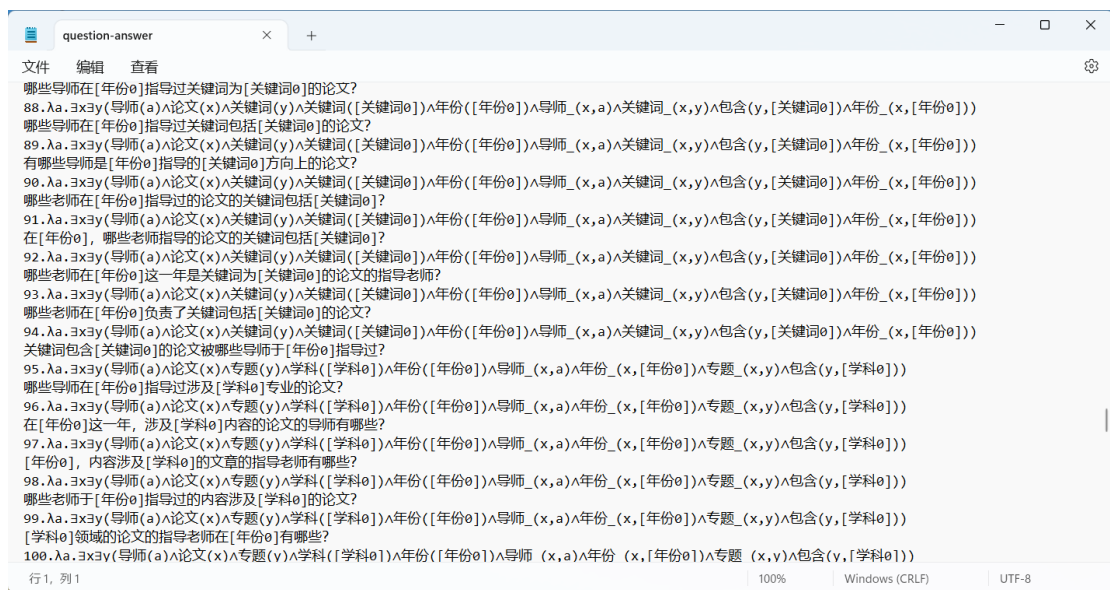
3.实践成果：主要自然语言处理模型学习及效果展示

我们组中的管心妍同学利用寒假对机器学习的主流算法如线性回归、支持向量机、决策树、神经网络、贝叶斯分类等进行了原理上的学习，并和组内同学介绍了其作用。

下面展示了部分学习笔记。



此后，我们构建了自然语言问句处理问句-答案对的数据集。



对 Seq2Seq 进行了学习和调试。

首先导入数据集并输出所有问句及其解析形式的字符个数。

将问句及解析形式添加进输入列表。

```

def get_dataset(data_path, num_samples):
    input_texts = [] # 存放输入的问句字符串
    target_texts = [] # 对应解析形式字符串

    input_characters = set() # 英文字符
    target_characters = set() # 法文字符
    with open(data_path, 'r', encoding='utf-8') as f:
        lines = f.read().split('\n')
    for line in lines[: min(num_samples, len(lines) - 1)]:
        input_text, target_text, _ = line.split('\t')
        # 用tab作用序列的开始，用\n作为序列的结束
        target_text = '\t' + target_text + '\n'

        input_texts.append(input_text) # 添加进输入列表
        target_texts.append(target_text)

        for char in input_text: # 判断是否在问句字符集合中
            if char not in input_characters:
                input_characters.add(char)
        for char in target_text:
            if char not in target_characters:
                target_characters.add(char)
    return input_texts, target_texts, input_characters, target_characters

```

将输入序列转化为独热编码，并将识别结果作为输入传入到解码网络。


```

encoder_input_data = np.zeros(
    (len(input_texts), max_encoder_seq_length, num_encoder_tokens),
    dtype='float32') #转化为独热编码

decoder_input_data = np.zeros(
    (len(input_texts), max_decoder_seq_length, num_decoder_tokens),
    dtype='float32')

decoder_target_data = np.zeros(
    (len(input_texts), max_decoder_seq_length, num_decoder_tokens),
    dtype='float32')

for i, (input_text, target_text) in enumerate(zip(input_texts, target_texts)):
    # 为末尾加上" "空格
    for t, char in enumerate(input_text):
        encoder_input_data[i, t, input_token_index[char]] = 1.
        encoder_input_data[i, t + 1:, input_token_index[' ']] = 1. #长度不足的位置设为空格

    # 相当于前一个内容的识别结果，作为输入，传入到解码网络中
    for t, char in enumerate(target_text):
        decoder_input_data[i, t, target_token_index[char]] = 1.
        if t > 0:
            # decoder_target_data不包括第一个tab
            decoder_target_data[i, t - 1, target_token_index[char]] = 1.

    decoder_input_data[i, t + 1:, target_token_index[' ']] = 1.
    decoder_target_data[i, t:, target_token_index[' ']] = 1.

```

读取数据集并定义神经网络及训练样本数。

```

# 每一次输入64个batch
batch_size = 64
# 训练一百个世代
epochs = 100
# 256维神经元
latent_dim = 256
# 一共400个样本
num_samples = 400

# 读取数据集
data_path = 'question-answer.txt'

```

开始训练，导入权重文件，将输入传入编码器，由解码器生成输出。训练完毕后保存模型。

```

# 开始训练
model.compile(optimizer='rmsprop', loss='categorical_crossentropy',
              metrics=['accuracy'])
model.fit([encoder_input_data, decoder_input_data], decoder_target_data,
        batch_size=batch_size,
        epochs=5,
        validation_split=0.2)

# 保存模型
model.save('out.h5')
K.clear_session()

encoder_inputs = Input(shape=(None, num_encoder_tokens))

encoder_outputs, state_h, state_c = LSTM(latent_dim, return_state=True)(encoder_inputs)
encoder_states = [state_h, state_c]
encoder_model = Model(encoder_inputs, encoder_states)
encoder_model.load_weights("out.h5", by_name=True)
encoder_model.summary()

decoder_inputs = Input(shape=(None, num_decoder_tokens))
decoder_state_input_h = Input(shape=(latent_dim,))
decoder_state_input_c = Input(shape=(latent_dim,))

decoder_states_inputs = [decoder_state_input_h, decoder_state_input_c]

decoder_outputs, state_h, state_c = LSTM(latent_dim, return_sequences=True,

```

进行运行和调试。

```

stop_condition = False
decoded_sentence = ''
while not stop_condition:
    # 以t为开头，一个一个的预测
    output_tokens, h, c = decoder_model.predict(
        [target_seq] + states_value)

    sampled_token_index = np.argmax(output_tokens[0, -1, :])
    sampled_char = reverse_target_char_index[sampled_token_index]
    decoded_sentence += sampled_char
    # 如果达到结尾
    if (sampled_char == '\n' or len(decoded_sentence) > max_decoder_seq_length):
        stop_condition = True
    states_value = [h, c]

decode_sequence()

```

Run: seq2seq_lstm - 2023-03-24 20:41:08.715165: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations: AVX AVX2. To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.

Epoch	Loss	Accuracy	Val Loss	Val Accuracy
Epoch 1/5	1.2668	0.7182	1.1887	0.6959
Epoch 2/5	0.9733	0.7362	1.0264	0.7105
Epoch 3/5	0.8741	0.7584	0.9409	0.7382
Epoch 4/5	0.7641	0.7853	0.8320	0.7613
Epoch 5/5	0.7343	0.7942	-	-

接下来，我们将进一步学习如何操作和调试 BERT 模型，完成后续部分工作。

- [1] 王守会,覃飙.知识库问答系统研究进展[J].小型微型计算机系统,2021,42(09):1793-1801
- [2] 包树南. 基于深度学习的自然语言到机器语言的翻译解析研究[D].西安电子科技大

学,2020.DOI:10.27389/d.cnki.gxadu.2020.000587.

[3] Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks[J]. Advances in neural information processing systems, 2014, 27.

数据爬取部分：

对东南大学硕博网站的论文信息进行了爬取与保存，主要步骤如下：

- 1. 获取页面 1 中的论文中问题名下面对应的 url，并存入数组为后续使用做准备；
- 2. 读取所有的 url 并对页面 2 需要内容进行爬取，再对爬取的内容进行清洗处理，构建数据库；
- 3. 将数据库写入 Excel 表格中。

这个部分我们爬取的网站地址为：

硕士：

http://223.3.67.16/tpi65/SingleSearch?id=&name=SSLW&displayDBName=%e7%a1%95%e5%a3%ab%e8%ae%ba%e6%96%87%e5%ba%93

博士：

http://223.3.67.16/tpi65/SingleSearch?id=4&name=BS2013&displayDBName=%u535A%u58EB%u8BBA%u6587%u5E93

序号	论文中文题名	姓名	学号	归档日期	下载
1	术后胃瘫与Roux-en-Y重建消化道肠-肠吻合方式的相关性...	常微	155634	2018-06-08	↓
2	生物质热解提质转化为高品质液体燃料的生命周期评价	瞿婷婷	090317	2012-4-23	↓
3	基于多参量退化数据的可靠性建模与维修策略优化	瞿众洲	100259	2013-5-24	↓
4	劳工权利的法律保护——以国际贸易为视角	瞿振华	096030	2012-5-24	↓
5	垃圾车车厢随机振动疲劳分析及结构优化	瞿云飞	060116	2008-12-24	↓
6	基于蛋白质组学方法的金纳米粒子生物相容性机理研究	瞿颖华	061356	2009-5-26	↓
7	应用磁共振组织追踪技术评价左室心肌力学参数的研究	瞿洋洋	142908	2017-06-02	↓
8	复合氧化物负载钨酸钠催化剂的制备及其乙酰丙酸加氢性能研究	瞿叙昊	182883	2021-09-23	↓
9	基于非参数方差结构的半参数回归模型的异方差检验	瞿肖怡	061016	2009-3-26	↓
10	高灵敏度GPS跟踪基带设计	瞿晓	131197	2016-06-06	↓

术后胃瘫与Roux-en-Y重建消化道肠-肠吻合方式的相关性分析：回顾性临床研究	
浏览量: 0 下载量: 0	
全文下载 在线预览	
姓名: 盛薇	
学号: 155634	
论文英文关键词: Roux-en-Y_postoperative gastroparesis_anastomotic fashion_	
学位: 硕士	
第一导师姓名: 汤文浩	
第一导师单位: 东南大学	
论文总页数: 52	
分类号: R619	
归档日期: 2018-06-08	
论文答辩日期: 2018-05-24	
论文保密级别: 公开	
论文英文题名: ANALYSIS OF POSTOPERATIVE GASTROPARESIS ASSOCIATED WITH THE ANASTOMOTIC FASHION OF JEJUNOJEJUNAL ANASTOMOSIS IN ROUX-EN-Y RECONSTRUC	
TION: A RETROSPECTIVE STUDY	
论文中文关键词: Roux-en-Y_Billroth_术后胃瘫_肠侧吻合_侧侧吻合	
<p>中文摘要: 背景: 胃瘫是腹部外科手术后较为常见的并发症,也是住院时间延长和再次入院的常见原因。在许多腹部手术的消化道重建方式中,Roux-en-Y已被证明比Billroth I和 Billroth II有更少的手术并发症。不过,Roux-en-Y术后胃瘫的发病率相对较高,其形成机制迄今尚未完全明了。Roux-en-Y胃肠道重建至少有2个吻合口,一个是头侧的(胃-肠、胆-肠或胰-肠)吻合口,另一个是尾侧的肠-肠吻合口。肠肠吻合可分两种术式:侧侧吻合与端侧吻合。目前肠肠吻合与术后胃瘫发生尚不清楚。目的:通过回顾性临床研究,分析术后胃瘫与Roux-en-Y重建消化道肠-肠吻合方式之间的关系。方法:收集2013年7月至2017年10月在中大医院行Roux-en-Y重建消化道手术但保留了全胃或近端胃的病例进行回顾性研究。总共回顾分析符合纳入标准的180例Roux-en-Y重建手术病例,探讨吻合方式与术后胃瘫的相关性。结果:在180例Roux-en-y吻合术后患者中,胃瘫发病率为23%(n=41)。肠-肠侧侧吻合术后胃瘫的发生率明显高于肠-肠端侧吻合(28%(33/118) vs 13%(8/62),P=0.022)。结论:术后胃瘫的发病率与吻合口的重建方式有明显的关系。Roux-en-Y术中采用肠肠侧侧吻合方式的患者更易出现术后胃瘫。</p> <p>英文摘要: Background: Postoperative gastroparesis (PG) is a common complication and one of the causes for prolonged hospital stay or readmission after abdominal surgeries. Roux-en-Y reconstruction method has been proven to be beneficial over Billroth reconstruction methods in abdominal surgeries, however it commonly presents with postoperative gastroparesis as complication, while its pathogenesis remains poorly understood up to date. Roux-en-Y reconstruction surgeries comprise of a minimum of two anastomoses, one proximal (gastrojejunostomy, choledochojejunostomy or pancreaticojejunostomy) and one distal (jejunojejunostomy). The distal anastomosis can be either a side to side anastomosis or end to side anastomosis. Up till now, the relationship between the anastomotic fashion of jejunojejunal anastomosis and postoperative gastroparesis has not been established.Objective: The aim of this study was to explore the association of postoperative gastroparesis with the anastomotic fashion of jejunojeju</p> <p>参考文献: 1. Wang YR, Fisher RS, Parkman HP. Gastroparesis-related hospitalizations in the United States: trends, characteristics, and outcomes, 1995-2004. Am J Gastroenterol. 2004; 103:313-322.2. Parkman HP, Hasler WL, Fisher RS, American Gastroenterological A. American Gastroenterological Association medical position statement: diagnosis and treatment of gastroparesis. Gastroenterology. 2004; 127:1589-1591.3. Parkman HP, Camilleri M, Farrugia G, et al. Gastroparesis and functional dyspepsia: excerpts from the AGA/ANMS meeting. Neurogastroenterol Motil. 2010; 22:113-133.4. Malleo G, Crippa S, Butturini G, et al. Delayed gastric emptying after pylorus-preserving pancreaticoduodenectomy: validation of International Study Group of Pancreatic Surgery classification and analysis of risk factors. HPB (Oxford). 2010; 12:610-618.5. Xiao XM, Gao C, Yin W, et al. Pylorus-Preserving versus Distal subtotal gastrectomy for surgical treatment of early gastric cancer: A Meta-Analysis. Hepatogastroenterology.</p> <p>参考文献数量: 119</p> <p>论文全文名称: 100482盛薇155634.pdf</p> <p>培养单位: 医学院</p> <p>专业: Su</p> <p>论文16: 100482盛薇155634_16.pdf</p>	

页面 2

具体功能实现过程:

i. 步骤 1:

获取页面 1 中的论文中问题名下面对应的 url, 并存入数组为后续使用做准备。

由于该网站使用 JavaScript 来动态生成网页内容, 故需要使用 selenium 模拟用户访问网页来获取内容。

ii. 步骤 2:

读取所有的 url 并对页面 2 需要内容进行爬取, 再对爬取的内容进行清洗处理, 构建数据库。

主要使用了 BeautifulSoup 进行标签内容的提取, 用正则表达式进行数据的清洗, 以下是具体代码与实现过程。

由于爬取数据中有部分缺失, 所以使用了 try...except 语句进行处理, 同时由于网页源码中所有数据标签一致, 这里采用先将数据全部爬下来再用正则表达式进行处理。

iii. 步骤 3:

将数据库写入 Excel 表格中。

前后端数据整合部分:

总体框架: 用 Django+Python 制作网页的后端部分, 用 ElasticSearch 数据库存放论文数据, 用 Kibana 可视化管理系统管理



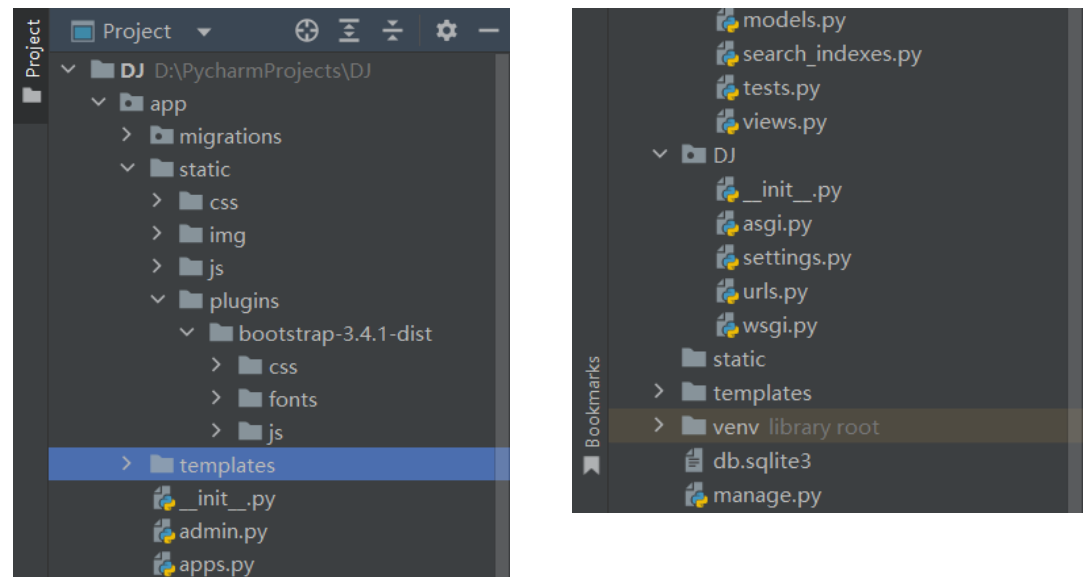
ElasticSearch 数据库。

一、Django 框架

介绍部分: Django 是高水准的 Python 编程语言驱动的一个开源模型 M. 视图 V, 控制器风格 T 的 Web 应用程序框架, 它起源于开源社区。使用这种架构, 程序员可以方便、快捷地创建高品质、易维护、数据库驱动的应用程序。这也正是 OpenStack 的 Horizon 组件采用这种架构进行设计的主要原因。另外, 在 Django 框架中, 还包含许多功能强大的第三方插件, 使得 Django 具有较强的可扩展性。

实现部分:

Django 在 python 中创建的工程框架如下:



其中 M,V,T 分别指的是:

Model:在设计项目时, 使用了 Django 预设的模型。

View:在设计视图部分, 通过自行设计函数, 设计了网页的后端处理部分。

例如:

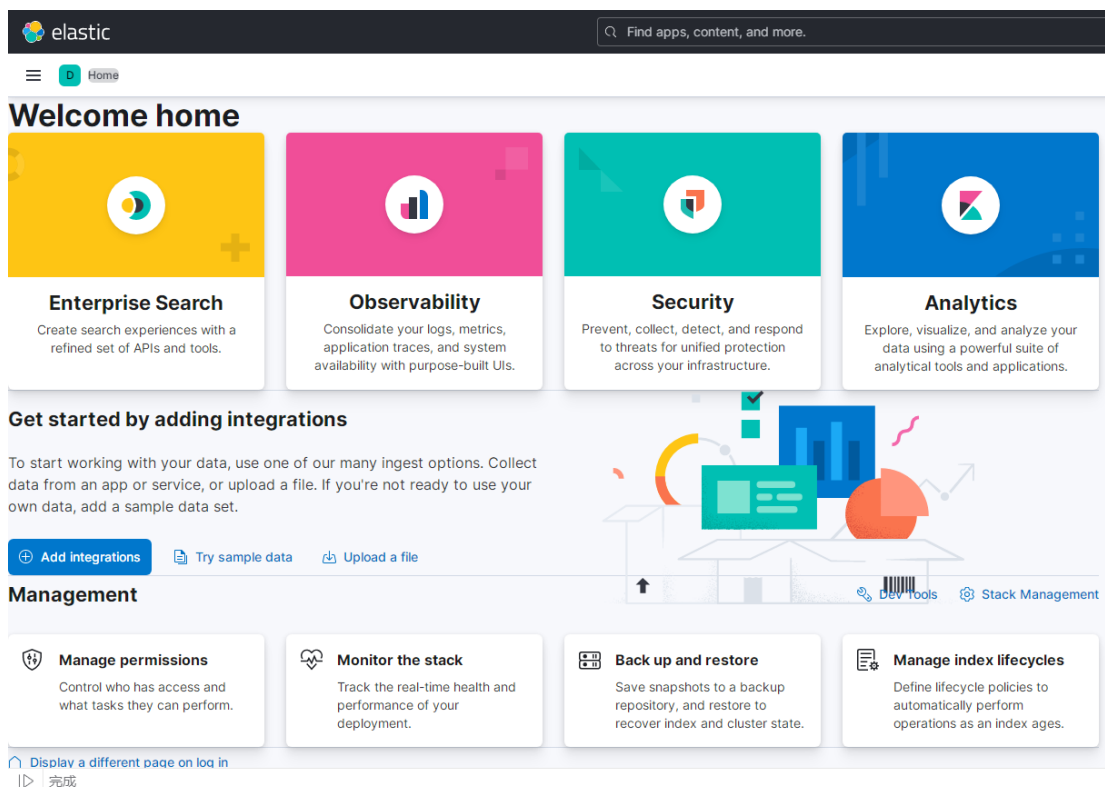
```
def es_index(request):
    if request.method == "GET":
        return render(request, "index.html")
    index = request.POST.get("input_index")
    doc_id = request.POST.get("input_id")
    name = request.POST.get("input_name")
    writer = request.POST.get("input_writer")
    teacher = request.POST.get("input_teacher")
    majority = request.POST.get("input_majority")
    time = request.POST.get("input_time")
    if index == "" or doc_id == "" or name == "" or writer == "" or teacher == "" or majority == "" or time == "":
        return render(request, "response.html", {"type": "输入数据不可为空"})
    es = Elasticsearch("http://localhost:9200/", http_auth=("elastic", "123456"))
    body = {
        "name": name,
        "writer": writer,
        "teacher": teacher,
        "majority": majority,
```

Templates:模板部分, 使用了 html 文件作为模板, 将视图部分功能与模板相关联。

二、ElasticSearch 数据库+Kibana 数据可视化

介绍部分: Elasticsearch 是一个分布式、高扩展、高实时的搜索与数据分析引擎。它能很方便的使大量数据具有搜索、分析和探索的能力。充分利用 Elasticsearch 的水平伸缩性, 能使数据在生产环境变得更有价值。Elasticsearch 的实现原理主要分为以下几个步骤, 首先用户将数据提交到 Elasticsearch 数据库中, 再通过分词控制器去将对应的语句分词, 将其权重和分词结果一并存入数据, 当用户搜索数据时候, 再根据权重将结果排名, 打分, 再将返回结果呈现给用户。

实现部分: 通过下载 ElasticSearch 以及 Kibana,并且配置了相关文件, 在启动 ElasticSearch 和 Kibana 后就可以通过在浏览器中输入 localhost:5601 访问 Kibana 管理系统。



```
GET index1/_search
{
  "query": {
    "term": {
      "_id": {
        "value": "30000"
      }
    }
  }
}
```

25 ▾

26

27

28

29

30

31

32

33

34

35

36

37

38

39

```
"_source": {
  "name": "人工湿地处理公路初期径流设计方法研究",
  "writer": "王哲",
  "teacher": "傅大放",
  "majority": "NO DATA",
  "time": "2012-3-14",
  "studentID": "091023",
  "EKW": "Highway initial runoff;Constructed wetland,Filled
;Drying out",
  "degree": "2",
  "FTU": "东南大学",
  "STN": "NO DATA",
  "STU": "NO DATA",
  "pages": 57,
  "FD": "2012-3-14",
  "DB": "2012-3-8"
```

三、实际操作部分

第一点: Django 部分的设计视图(view)部分。通过前端网页的传参, 确定后端使用到的函数。其中, 函数都与 ES 数据库进行了联动, 在建立与 ES 数据库的连接后, 通过编写 ES 操作语句与调用 Python 的 ES 函数库, 对 ES 数据库进行目标操作, 附带编写了异常处理部分。

例: 插入文档(Doc)函数

```
def es_index(request):
    if request.method == "GET":
        return render(request, "index.html")
    index = request.POST.get("input_index")
    doc_id = request.POST.get("input_id")
    name = request.POST.get("input_name")
    writer = request.POST.get("input_writer")
    teacher = request.POST.get("input_teacher")
    majority = request.POST.get("input_majority")
    time = request.POST.get("input_time")
    if index == "" or doc_id == "" or name == "" or writer == "" or teacher == "" or majority == "" or time == "":
        return render(request, "response.html", {"type": "输入数据不可为空"})
    es = Elasticsearch("http://localhost:9200/", http_auth=("elastic", "123456"))
    body = {
        "name": name,
        "writer": writer,
        "teacher": teacher,
        "majority": majority,
        "time": time
    }
    if es.exists(index=index, id=doc_id):
        return render(request, "response.html", {"type": "添加失败, id已存在"})

    es.index(index=index, id=doc_id, body=body, ignore=400)
    return render(request, "response.html", {"type": "添加成功"})
```

第二点: 论文数据的导入部分。通过查询相关资料, 下载并安装了 Anaconda, 在 Pycharm 终端安装了其中的 Panda 与 xrls 库, 将爬取到的数据导入 ES 数据库。

其中, 导入部分的详细逻辑是:

(1) Panda 库与 xrls 库联动, 读取 excel 文件数据到缓存中:

```
rd = pd.read_excel("D:\PycharmProjects\DJ\硕士论文全.xls")
```

(2) 通过映射关系, 将读取到的数据通过 ES 语句依次存入 ES 数据库中。


```

for count in range(0, 56708):

    body = {...}
    print("Loading:" + str(count) + "/56708")
    if es.exists(index="index1", id=count+5869):
        return render(request, "response.html", {"type": "添加失败, id已存在"})

    es.index(index="index1", id=count+5869, body=body, ignore=400)

```

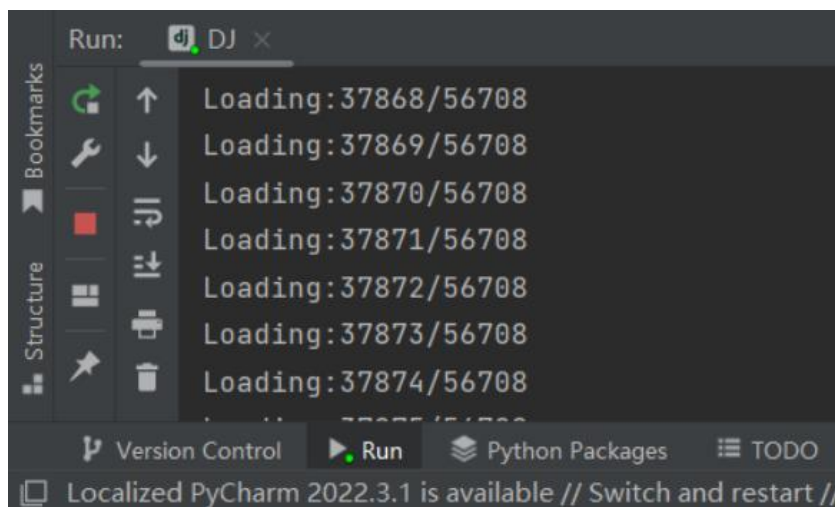
body 中的映射关系例如:

```

"pages": rd.values[count, 10],
"FD": rd.values[count, 12],
"DR": rd.values[count, 13],

```

(3) 通过 ES 服务器通道, 按照最大速率传输数据。



一共传输了不少于 60000 条论文相关数据, 用时约 18 小时。

第三点: Django 框架中 url 部分与设计视图(view)的联动, 以及设计视图中函数与模板 html 文件的联动。

(1) 通过相关语句, 将 url 与设计视图中的函数进行联系, 即用 url 调用函数。

(2)

```

from django.urls import path
from app import views
urlpatterns = [
    # path('admin/', admin.site.urls),
    path('', views.main),
    path('bird/', views.bird),
    path('tpl/', views.tpl),
    path('bilibili/', views.bilibili),
    path('login/', views.login),
    path('es/create', views.es_create),

```

```

<tbody class="tableBody">
{% for user in s_user %}
    {% csrf_token %}
    <tr>
        <td>{{ forloop.counter }}</td>
        <td>{{ user.name }}</td>
        <td>{{ user.writer }}</td>
        <td>{{ user.teacher }}</td>
        <td>{{ user.majority }}</td>
        <td>{{ user.time }}</td>
        <td>
            <span>
                {% csrf_token %}
                <form action="http://127.0.0.1:8000/kwigss2/info" method="post">
                    <input type="hidden" name="cname" value="{{ user.name }}">

```

其中，user 为设计视图中传入的一个类对象：

```

class Info(object):
    def __init__(self, p, d, n, w, te, m, ti, st, ekw, de, f,
                  stn, s, pa, fd, dr, ar, cn, sl, en, ckw, cs,
                  es, rd, ftn, dc, tu, tun, tuc, ss, p16):
        self.pos = p # 在s_user中的位置
        self.id = d # 在索引中的id号
        self.name = n # 论文名
        self.writer = w # 作者名
        self.teacher = te # 第一导师姓名
        self.majority = m # 专业名称
        self.time = ti # 发表时间
        self.studentID = st # 学号
        self.EKW = ekw # 英文关键词

```

以此类对象作为每一条论文信息的容器，方便后续操作。

网页前端部分：

总体上：使用 html 语言制作网页，辅助 css 样式和 javascript 脚本语言，并通过 bootstrap 和 font-awesome 第三方库改进更新网页效果。

网站总体效果：

Try being an interesting Paper Engine

Search for...

Search for...

[illegible]

此处放置推荐内容

蛋白质折叠-去折叠过程中间体的实验研究

한글

生物科学与医学工程学院

摘要: 当前我国重大环境问题频繁发生, 开展环境污染对人体健康风险评估与控制技术研究, 具有重要的科学和现实意义。基于分子印迹聚合物 (Molecularly imprinted polymers, MIPs) 的分子印迹固相萃取 (Molecularly imprinted solid phase extraction, MIPSE) 技术已逐渐取代基于 C18 柱、免疫亲和层析柱等常规样品前处理技术, 在国内食品安全检测中广泛应用。由于大量生物样本中的目标分子均存在于水相体系中, 而亲水性 MIPs 相对于有机相中制备的 MIPs 对目标分子具有更高的特异性和选择性, 使得样品前处理效率更高。

关键词: 牛血清蛋白; 胰蛋白酶; 1-中间体; 荧光光谱; 1CD光谱; 紫外吸收光谱

英文摘要: Using molecularly imprinted polymers (MIPs) as specific adsorbents, molecularly imprinted solid-phase extraction (MISPE) can selectively purify and enrich the analytes efficiently with simplicity and stability in extreme conditions and draw more attention in the application of MIPs in recent decades. With the increasing residual drugs in foods and feeds, water-compatible MIPs and multi-residual analysis technique have been developed to separate drugs in aqueous conditions such as those biological and environmental samples. Unfortunately, preparation of aqueous MIPs in water-containing system and MIPs for multi-residual analysis targeting multiple analytes is still of great challenge.

英文关键词: Bovine serum albumin; Trypsin; Intermediate; Fluorescence spectroscopy; ICD spectra; UV absorption spectra

导师：第一导师：陆祖宏 第二导师：NONE

分类号: R318, UDC575

1. html 语言

使用“标记”（markup）来注明文本、图片和其他内容，以便于在 Web 浏览器中显示。HTML 标记包含一些特殊“元素”

如 [<head>](#)、[<title>](#)、[<body>](#)、[<header>](#)、[<footer>](#)、[<article>](#)、[<section>](#)、[<p>](#)、[<div>](#)、[](#)、[](#)、[<aside>](#)、[<audio>](#)、[<canvas>](#)、[<datalist>](#)、[<details>](#)、[<embed>](#)、[<nav>](#)、[<output>](#)、[<progress>](#)、[<video>](#)、[](#)、[](#)、[](#) 等等。

HTML 元素通过“标签”（tag）将文本从文档中引出，标签由在“<”和“>”中包裹的元素名组成，HTML 标签里的元素名不区分大小写。也就是说，它们可以用大写，小写或混合形式书写。例如，<title> 标签可以写成 <Title>，<TITLE> 或以任何其他方式。然而，习惯上与实践上都推荐将标签名全部小写。

网站使用 html 语言布局

```
1  </head>
2  <body>
3  <div class="container">
4    <div class="page-header">...</div>
5
6    <div class="main-body">
7      <div class="col-xs-6 论文列表">
8        <table class="table table-bordered table-hover table-striped" style="border-radius: 40px;">
9          <nav aria-label="Page navigation">...</nav>
10      </div>
11
12      <div class="col-xs-3 推荐">此处放置推荐内容</div>
13    </div>
14    <div>
15      </div>
16    </div>
17  </div>
```

2. css 样式:

HTML 用于定义内容的结构和语义，CSS 用于设计风格和布局

```
        width: 1200px;
        margin-top: 0px;
    }

    .detail-container {
        margin-top: 20px;
        height: 500px;
    }

    .info-container {
        margin-left: 30px;
    }

    .title-container div {
        text-align: center;
    }

    .title-container {
        margin-top: 30px;
    }

    .title-span {
        font-weight: bold;
        font-size: 20px;
    }

    .author-div {
        margin-top: 15px;
    }

    .author-span {
        font-size: 15px;
        color: darkblue;
    }

    .college-div {
        margin-top: 5px;
    }

    .college-span {
        font-size: 15px;
        color: darkblue;
    }

    .content-container {
        margin-top: 20px;
    }

    .title-摘要 {
        color: black;
    }

    .grey-font {
        color: #666;
    }

    .row-info {
        margin-bottom: 5px;
    }

    .第一导师{
        margin-right: 20px;
    }
</style>
</head>
<body>
<nav class="navbar navbar-default">
    <div class="container-fluid">
```

3. javascript:

JavaScript 是一种脚本，一门编程语言，它可以在网页上实现复杂的功能，网页展现给你的不再是简单的静态信息，而是实时的内容更新，交互式的地图，2D/3D 动画，滚动播放的视频等等。JavaScript 怎能缺席。它是标准 Web 技术蛋糕的第三层

```
<script type="text/javascript">

function show(){
    // 1.找到标签并获取他的内容
    var tag = document.getElementById("txt");
    var dataString = tag.innerText;

    var firstChar = dataString[0];
    var otherString = dataString.substring(1, dataString.length);
    var newText = otherString + firstChar;

    // 3.更新内容
    tag.innerText = newText;
    console.log(newText);
}

// JavaScript中的定时器
setInterval(show, 1000);
```

总结:

网站通过上述三种技术实现，并配合以一些第三方库来达到视觉上更好看的效果:

页面一代码

```

<nav class="navbar navbar-default">
  <div class="container-fluid">
    <!-- Brand and toggle get grouped for better mobile display -->
    <div class="navbar-header">
      <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1" aria-expanded="false">
        <span class="sr-only">Toggle navigation</span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
      <a class="navbar-brand" href="#"> </a>
    </div>

    <!-- Collect the nav links, forms, and other content for toggling -->
    <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
      <ul class="nav navbar-nav">
        <li class="active"><a href="#">首页 <span class="sr-only">(current)</span></a></li>
        <li><a href="#">新闻</a></li>
        <li class="dropdown">
          <a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true" aria-expanded="false">资助我们</a>
        </li>
      </ul>
      <ul class="nav navbar-nav navbar-right">
        <li><a href="#">文档</a></li>
        <li class="dropdown">
          <a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true" aria-expanded="false">下载本站 <span class="sr-only">(current)</span></a>
          <ul class="dropdown-menu">
            <li><a href="#">如何下载</a></li>
            <li><a href="#">视频教程</a></li>
            <li><a href="#">关于</a></li>
            <li role="separator" class="divider"></li>
            <li><a href="#">帮助</a></li>
          </ul>
        </li>
      </ul>
    </div><!-- /.navbar-collapse -->
  </div><!-- /.container-fluid -->
</nav>

<div class="body">
  <div class="container">
    <div class="logo">
      <h1 style="text-align: center;">
        Kwiggs
        <!---->
      </h1>
      <h3 style="color:#3F3F3F; opacity:0.7; height: 25px;">
        <span style="margin: 5px auto 10px; display:inline-block;" class="logo-tip">
          Try being an interesting Paper Engine
        </span>
      </h3>
    </div>

    <div class="row search">
      <div class="col-lg-6">
        <form action="/kwiggs2/s" method="post">
          <div class="input-group search-box">
            <input type="text" class="form-control" placeholder="Search for...">
            <span class="input-group-btn">
              <button class="btn btn-default" type="submit">
                <span class="glyphicon glyphicon-search" aria-hidden="true"></span>
              </button>
            </span>
          </div><!-- /input-group -->
        </form>
      </div><!-- /.col-lg-6 -->
    </div><!-- /.row -->
  </div><!-- /.col-lg-6 -->
</div><!-- /.row -->
</div>

```

页面二代码


```

<div class="page-header">
  <div class="row search">

    <div class="col-lg-6">
      <form action="/kwwgss/s" method="post">
        <div class="input-group search-box">

          <input type="text" class="form-control" placeholder="Search for...">
          <span class="input-group-btn">
            <button class="btn btn-default" type="button">
              <span class="glyphicon glyphicon-search" aria-hidden="true"></span>
            </button>
          </span>
        </div><!-- /input-group -->
      </form>
    </div><!-- /.col-lg-6 -->
  </div><!-- /.row -->
</div>

<div class="main-body">
  <div class="col-xs-6 论文列表">
    <table class="table table-bordered table-hover table-striped" style="border-radius: 40px;"...>
    <nav aria-label="Page navigation"...>
  </div>

  <div class="col-xs-3 推荐">此处放置推荐内容</div>
</div>
<div>
  </div>
</div>

```

页面三代码

```
<!-- Brand and toggle get grouped for better mobile display -->
<div class="navbar-header">
  <button type="button" class="navbar-toggle collapsed" data-toggle="collapse"
    data-target="#bs-example-navbar-collapse-1" aria-expanded="false">
    <span class="sr-only">Toggle navigation</span>
    <span class="icon-bar"></span>
    <span class="icon-bar"></span>
    <span class="icon-bar"></span>
  </button>
  <a class="navbar-brand" href="#">Kwigs</a>
</div>

<!-- Collect the nav links, forms, and other content for toggling -->
<div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
  <ul class="nav navbar-nav">
    <li class="active"><a href="#">首页 <span class="sr-only">(current)</span></a></li>
    <li><a href="#">新闻</a></li>
  </ul>
  <form class="navbar-form navbar-left">
    <div class="form-group">
      <input type="text" class="form-control" placeholder="Search">
    </div>
    <button type="submit" class="btn btn-default">
      <span class="glyphicon glyphicon-search" aria-hidden="true"></span>
    </button>
  </form>
  <ul class="nav navbar-nav navbar-right">
    <li><a href="#">文档</a></li>
    <li class="dropdown">
      <a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true"
        aria-expanded="false">关于 <span class="caret"></span></a>
      <ul class="dropdown-menu">
        <li><a href="#">Action</a></li>
        <li><a href="#">Another action</a></li>
        <li><a href="#">Something else here</a></li>
        <li role="separator" class="divider"></li>
        <li><a href="#">Separated link</a></li>
      </ul>
    </li>
  </ul>
</div><!-- /.navbar-collapse -->
</div><!-- /.container-fluid -->
</nav>

<div class="container">
  <div class="col-xs-3">
    <div class="panel panel-primary detail-container">
      <!-- Default panel contents -->
      <div class="panel-heading">论文相关细节</div>

      <!-- List group -->
      <ul class="list-group">
        <li class="list-group-item">学号: 079187</li>
        <li class="list-group-item">专业名称: none</li>
        <li class="list-group-item">学位: 博士</li>
        <li class="list-group-item">第一导师单位: 东南大学</li>
        <li class="list-group-item">第二导师单位: none</li>
        <li class="list-group-item">论文总页数: 97</li>
        <li class="list-group-item">归档日期: 2010-1-12</li>
        <li class="list-group-item">论文答辩日期: 2010-1-8 0</li>
        <li class="list-group-item">参考文献数量: 408</li>
        <li class="list-group-item"></li>
      </ul>
    </div>
  </div>

  <div class="col-xs-8 info-container">
    <div class="title-container">
      <div class="title-div">
        <span class="title-span">
          蛋白质折叠-去折叠过程中熵体的实验研究
        </span>
      </div>

      <div class="author-div">
        <span class="author-span">
          曹鹏
        </span>
      </div>

      <div class="college-div">
        <span class="college-span">
          生物科学与医学工程学院
        </span>
      </div>
    </div>

    <div class="content-container">
      <div class="row-info">
        <span class="title-摘要">
          摘要
        </span>
      </div>
    </div>
  </div>
</div>
```

173	
174	摘要:
175	
176	
177	当前我国重大环境问题频繁发生,开展环境污染对人体健康风险评估与控制技术研究。
178	具有重要的科学和现实意义。基于分子印迹聚合物(Molecularly imprinted polymers, MIPs)的分子印迹固相萃取
179	(Molecularly imprinted solid phase extraction, MISPE)技术已逐渐取代基于C18小柱、免疫亲和和层析柱等常规样品前处理技术,
180	在国内外食品安全检测中广泛应用。由于大多数生物样本中的目标分子均残存于水相体系中,
181	而亲水性MIPs相对于有机相中制备的MIPs对目标分子具有更高的特异性和选择性,使得样品前处理效率更高。
182	
183	</div>
184	
185	<div class="row-info">
186	
187	关键词:
188	
189	
190	牛血清蛋白;胰蛋白酶; \中间体; 荧光光谱; \CD光谱; 紫外吸收光谱
191	
192	</div>
193	
194	<div class="row-info">
195	
196	英文摘要:
197	
198	
199	Using molecularly imprinted polymers (MIPs) as specific adsorbents,
200	molecularly imprinted solid-phase extraction (MISPE) can selectively purify
201	and enrich the analytes efficiently with simplicity and stability in extreme conditions
202	and draw more attention in the application of MIPs in recent decades. With the increasing
203	residual drugs in foods and feeds, water-compatible MIPs and multi-residual analysis
204	technique have been developed to separate drugs in aqueous conditions such as those biological
205	and environmental samples. Unfortunately, preparation of aqueous MIPs in water-containing
206	system and MIPs for multi-residual analysis targeting multiple analytes is still of great challenge.
207	
208	</div>
209	
210	<div class="row-info">
211	
212	英文关键词:
213	
214	
215	Bovine serum albumin; Trypsin; \Intermediate; Fluorescence spectroscopy; \CD spectra; UV absorption spectra
216	
217	</div>
218	
219	<div class="row-info">
220	
221	导师:
222	
223	
224	第一导师: 陆祖宏
225	第二导师: NONE
226	
227	</div>
228	
229	<div class="row-info">
230	
231	分类号:
232	
233	
234	R318. UDC575
235	
236	</div>
237	
238	
239	</div>

附表 1:

东南大学校/院级 SRTP 项目中期检查表

院系名称：

项目名称	基于语义解析的学术论文问答系统				
项目编号	202309044	学生负责人	诸欣扬	指导教师	李慧颖
<p>一、项目进展情况：</p> <ol style="list-style-type: none"> 1. 目前已经完成论文的爬取与数据库的建立，人工智能的语义解析部分正在进行，现在我们的前端网页已经搭好，可以进行简单的互动，并执行非语义解析的模糊搜索，搜索到的论文会分页呈现，点进可看到具体的论文内容和论文信息。人工智能部分正在进行 seqToseq 模型的环境搭建，基础人智部分已经完成，还有待将数据输送进模型中训练。 2. 我们不希望拘泥于简单的论文搜索，从出发点来说，我们希望建立基于自然语言的论文检索系统是为了方便大家的使用，但同学们往往不能够一次就用准确充分的自然言语来描述自己的想法，因此受大数据自然语言模型的启发，我们希望能利用某些人工智能的技术来完善检索信息的发散，以用户提供的搜索语言为本，拓展到一些临近领域的关键词搜索，基于此，我们还在致力于将大数据自然语言模型类人工智能的接口应用到网页中，使得用户在不清楚自己的想法时，可以通过询问 AI 来确定深层信息。 3. 做锦上添花作用，在时间允许的情况下，我们将会内置一些图像或者数学工具提供给检索相关方面论文的同学，使得他们可以在阅览论文的同时使用一些简单的工具更好的理解论文内容，如果时间不够，我们会将对应有用工具的链接内置在网页中以供下载学习。 					
<p>二、存在的问题，及拟采取解决问题的措施</p> <ol style="list-style-type: none"> 1. 前端页面的美化问题，现在的网页稍显简陋，不足以吸引眼球，突出重点，目前已经让组内有相关审美设计经验的同学负责前端页面的整化 2. 人工智能 Bert 模型还不熟悉，需要投入一定的时间讨论，并想办法将模型和前后端搜索完美的整合起来，目前已经进行了部分人员调动，希望能更好的拟合模型。 					
<p>三、指导教师意见：</p> <p>《基于语义解析的学术论文问答系统》目前已经完成学术论文的爬取，前段页面搭建工作，正进一步搭建深度学习模型以实现自然语言解析。项目进展顺利，同意通过中期检查。</p> <p style="text-align: right;">签字：_____ 日期：_____</p>					
<p>四、院系中期检查小组结论及意见：</p> <p style="text-align: center;"> <input type="checkbox"/> 优秀 <input type="checkbox"/> 良好 <input type="checkbox"/> 通过 <input type="checkbox"/> 不通过 </p> <p>推荐排序：</p> <p style="text-align: right;">签字：_____ 日期：_____</p>					

注：表头及第一、第二栏由学生负责人填写。（可附页）