Atitit word结构化提取考试试题读取 poi读取word nlp aiprj

目录

[1.1. 结构化后数据 1](#_Toc16034)

[1.2. 文字读取 1](#_Toc841)

[1.3. 图片读取 1](#_Toc8594)

[1.4. Doc的表格读取 //遍历range范围内的table。 2](#_Toc1133)

[1.5. code 2](#_Toc23401)

## 结构化后数据

{

"linenum":85,

"topicNum":8,

"linenumEnd":95,

"topicType":"8.singlechoice[单选题](3分)68494d9cc2bb4396af50b30fc9e79d63\u00A0",

"topic":"Servlet生命周期的初始化在什么时候会调用init()方法( A )",

"ans":"A"

}

## **文字读取**

文字操作中，可以通过HWPFDocument 直接读取，也可以获取到Range对象后读取

## **图片读取**

Doc获取文档中的所有图片，并存储，其中Picture 对象中，getContent()可获取字节流，然后获取字节数据，自己写出来，也可以通过writeImageContent 来直接写，图片的开始位置可以获取，但是结束位置没有

## Doc的表格读取 //遍历range范围内的table。

TableIterator tableIter = new TableIterator(range);

while (tableIter.hasNext()) {

Table table = tableIter.next();

//开始位置

int start = table.getStartOffset();

//结束位置

int end = table.getEndOffset();

System.out.printf("开始位置%d,结束为止%d\r\n",start,end);

//获取行的数目

int rowNum = table.numRows();

for (int j = 0; j < rowNum; j++) {

//获取每一行

TableRow row = table.getRow(j);

int cellNum = row.numCells();

for (int k = 0; k < cellNum; k++) {

//获取每一列

TableCell cell = row.getCell(k);

// 输出单元格的文本

System.out.println(cell.text().trim());

}

}

}

---------------------

## code

package apkg;

import java.io.File;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import java.io.IOException;

import java.io.InputStream;

import java.rmi.NoSuchObjectException;

import java.text.Collator;

import java.util.Arrays;

import java.util.Collections;

import java.util.Comparator;

import java.util.List;

import java.util.Locale;

import java.util.Map;

import java.util.Set;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

import org.apache.commons.io.FileUtils;

import org.apache.poi.POIXMLProperties.CoreProperties;

import org.apache.poi.hwpf.HWPFDocument;

import org.apache.poi.hwpf.usermodel.Range;

import org.apache.poi.xwpf.extractor.XWPFWordExtractor;

import org.apache.poi.xwpf.usermodel.XWPFDocument;

import com.alibaba.fastjson.JSON;

import com.google.common.collect.Lists;

import com.google.common.collect.Maps;

import com.google.common.collect.Sets;

public class kaosiWordExt {

public static void main(String[] args) throws IOException {

// readTwoMannamesFileJoin();

// String f="C:\\Users\\Administrator\\Desktop\\namelib\\好听的女孩名字2225个.docx";

// List<String> femnames=getList2220femNnames(f,1);

// System.out.println(femnames);

// FileUtils.writeLines(new File("D:\\namelib\\fem2200names.txt"), femnames);

String f="D:\\0db\\人工智能1807javaA-云计算《电子商务》第3周周考理论题.doc";

// List<Map> femnames=getList(f);

HWPFDocument doc = new HWPFDocument(new FileInputStream(f));

//通过 Doc对象直接获取Text

StringBuilder sb = doc.getText();

String string = sb.toString();

System.out.println(string);

FileUtils.write(new File("d:\\doc2txt.txt"), string);

//通过Range对象获取Text

Range range = doc.getRange();

String text = range.text();

System.out.println(text);

List<Map> topics=Lists.newLinkedList();

String[] lines=string.split("\r");

for(int lineIdx=1;lineIdx<=lines.length;lineIdx++)

{

if(lineIdx==7)

System.out.println("d");

int linesListIdex=lineIdx-1;

String line=lines[linesListIdex].trim();

if(line.length()==0)

continue;

System.out.println(" lineIdx :"+lineIdx);

if(lineIdx==404)

System.out.println("d");

if(KaosiUtil.isTopicStart(line.trim()))

{

Map topic=Maps.newLinkedHashMap();

topic.put("linenum", lineIdx);

topic.put("topicNum",KaosiUtil.gettopicNum(line.trim()));

topic.put("linenumEnd", KaosiUtil.getNextTopicLineNum(lines,lineIdx+1));

topic.put("topicType",line.trim());

topic.put("topic",KaosiUtil.gettopic(lines,lineIdx+1));

topic.put("ans",KaosiUtil.getAnsBytopictitle( topic.get("topic").toString() ));

topics.add(topic);

System.out.println(JSON.toJSONString(topic, true));

}

}

}

(9+条消息)POI之Word文档读取-yellowcong - yelllowcong的专栏 - CSDN博客.html