

XINCHECK® 文本查重

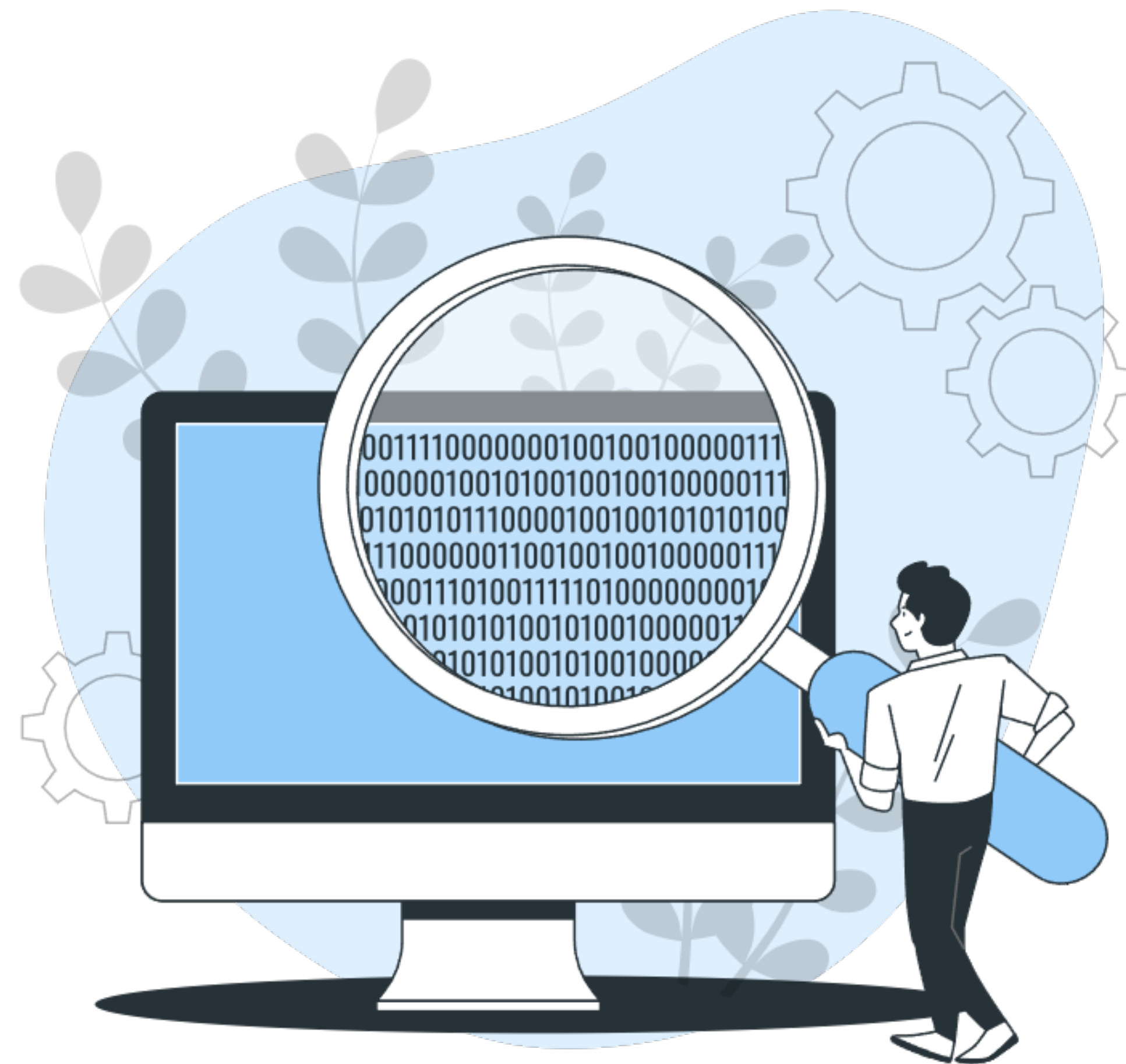
产品官网：<https://www.xincheck.com>



目录



- 产品简介
- 使用教程
 - 查重 SDK 快速使用教程
 - 查重 SDK 进阶使用教程
 - 文档查重软件快速使用教程
 - 更新日志
- 采购报价
 - 查重 SDK
 - 查重软件
- Q & A



产品简介



XINCHECK 文本查重基于自研的、业界认可的查重算法及查重引擎，提供易使用、易拓展、高性能的离线查重 **SDK** 和查重软件，可用于论文查重、标书查重/辅助串标检测、项目申报书查重、文档查重、文本去重、作业查重等多个场景。

产品核心竞争力

- 完全离线使用
- 线性的算法时间复杂度，任务自动并发处理，充分利用 CPU 性能
- 支持基于自建库查重、支持横向查重
- 支持设置排除白名单，排除不需要查重的文本内容
- 支持添加重点关注关键词，查重时重点标注关键词
- 切换查重算法(文本指纹算法/分句语义算法)
- 支持图像查重



产品简介 查重 SDK



基于 JAVA 语言的离线查重开发工具包

- ☑ 产品核心竞争力
- ☑ 丰富的接口，完善的开发文档，开发者可以获取查重报告中的全部数据
- ☑ 生成详细的、多种类型的查重报告，且查重报告样式可以通过接口进行部分自定义
- ☑ 20 余项可定制化配置项

产品简介 查重软件



基于 SDK 开发的桌面端软件

专注标书查重应用场景，提供白名单排除功能和添加重点关注关键词功能

☑ 产品核心竞争力

☑ 支持对文档属性和图片查重并生成图片重复报告等详细的、多种类型的查重报告

使用教程 查重 SDK 快速使用教程



通过 maven 将本 SDK 0.5.5 版本引入到项目中

```
1 <repository>
2   <id>XINCHECK</id>
3   <name>XINCHECK Public Repository</name>
4   <url>https://maven.xincheck.com/repository/pro/</url>
5 </repository>
```

然后在 <dependencies> 中添加依赖

```
1 <dependency>
2   <groupId>com.xincheck</groupId>
3   <artifactId>duplicate-check-pro</artifactId>
4   <version>0.5.7</version>
5 </dependency>
```

除 maven 外同样支持 Gradle、Ivy 等，修改对应引入语法即可。

使用教程 查重 SDK 快速使用教程



我们为商业用户提供免费的评估许可证¹，您可先通过以下方法获取并打印服务器或 PC 机的机器指纹

```
1 System.out.println(CheckManager.INSTANCE.getMachineCode());
```

将机器指纹输入到下面文本框中获取许可证

机器指纹:

获取评估授权许可证

授权许可证是一串较长的包含字母数字和符号的字符串。获取后通过以下方法对 SDK 授权

```
1 CheckManager.INSTANCE.setRegCode("授权许可证");
```

¹授权时常为 3 天，可无限次续期，每次续期时长 3 天。用于教育、科研或其它合理非商业用途可申请长有效期)，详情可访问[查重 SDK 采购报价表](#)

使用教程 查重 SDK 快速使用教程



为了便于部分简单应用场景下的开发，SDK 0.5.0 以上版本内置了简易启动器 EasyStarter，通过如下所示的一行代码即可完成 SDK 调用。参数介绍如下

- 参数 1: **待查文件**所在的文件夹路径(如果待查文件只有一个，可以传文件路径);
- 参数 2: **比对库文件**所在的文件夹路径(如果比对库中只有一个文件，可以传文件路径);
- 参数 3: 保存**查重报告**的文件夹路径。如果不需要导出查重报告可以传空字符串;
- 参数 4: **白名单文本**。对于标书查重等场景，有一些文本是允许重复的，这些文本可以通过该参数传入。

该参数可选，如不需要可以不传或传 null。

```
1 List<Reporter> reporters = EasyStarter.check(new File("参数1"), new File("参数。2"), "参数3", "参数4");
```

横向查重应用场景下参数 1 和参数 2 可以相同，查重时如果待查文件和比对库中的某一个文件完全相同会自动跳过比对，不会出现重复率 100

使用教程 查重 SDK 进阶使用教程



以下内容对开发过程中较为关键的部分做出介绍，详细的接口和参数说明请查阅完整[开发者接入文档](#)，完整实例代码参见 GitHub 项目中的 Sample 部分。

- ① 使用原生方式同步启动任务
- ② 通过 CheckState 观察者异步处理查重任务
- ③ 实例化文本对象(Paper)
- ④ 为本文对象补充额外信息
- ⑤ 为 Paper 设置 Payload

使用教程 使用原生方式启动查重任务



1 加载比对库²

将所有要作为比对库的文件放到一个文件夹中(支持 doc、docx、xls、xlsx、pdf、rtf、txt 格式), 实例化 PaperLibrary 时将文件夹路径传入构造方法。

```
1 LocalPaperLibrary paperLibrary = LocalPaperLibrary.load(new File("对比库文件夹路径"));
```

将待查重的文件加载为 Paper。Paper 支持多种加载方式, 可以通过文件、字符串、输入流加载, 也支持批量加载。

2 加载待查重文件

```
1 Paper paper = Paper.load(new File("文件路径")); //加载单个文件
```

²加载比对库会花费一定时间, 通常只需要在服务启动的时候加载一次, 后续的所有查重操作都不需要再重新加载比对库。

使用教程 使用原生方式启动查重任务



3 启动查重任务

```
1 //构建并启动任务
2 CheckTask checkTask = CheckManager .INSTANCE
3     .getCheckTaskBuilder () //获 取 查 重 任 务 构 建 器
4     .addLibrary(paperLibrary) //添加比对库。可以添加多个
5     .addCheckPaper(paper) //添加待查文本。可以添加多个
6     .build(); //构建任务，返回checkTask对象
7 checkTask.start(); //启动任务线程
8 checkTask.join(); //等待查重结束(阻塞)
```

该方式为同步启动查重任务，并通过 join 方法等待查重任务结束。SDK 支持异步调用。

使用教程 使用原生方式启动查重任务



4 保存查重报告

查重任务结束后可以通过以下方式将查重报告保存，SDK 可以生成三种查重报告。

```
1 checkTask.getReporters().get(0).saveAsFile("C:\\Report\\report1.html",
    ReportType.TEXT_WITH_CITATION); //保存全文标红查重报告
2 checkTask.getReporters().get(0).saveAsFile("C:\\Report\\report2.html",
    ReportType.TEXT_WITH_ORIGINAL); //保存原文对照查重报告
3 checkTask.getReporters().get(0).saveAsFile("C:\\Report\\report3.html",
    ReportType.SAMPLE); //保存简洁查重报告
```

可不将结果保存为文件，直接通过接口获取到查重文件，如：

```
1 String reportId = reporter.getReportId(); //获取查重报告id
2 String copyRate = reporter.getCopyRate(); //获取总重复率
3 String copyWords = reporter.getCopyWords(); //获取重复字数
```

查重报告的样式也可以通过接口进行一定程度的自定义。

使用教程 通过 CheckState 观察者异步处理查重任务



通过实现 CheckState 接口，可以实现查重任务的异步处理。

CheckState接口中包含start、finish、fail三个方法，分别对应查重任务提交后的启动、完成和失败。使用setCheckState方法将该CheckState注册到查重任务中。这样在查重任务的各个状态就会回调CheckState的不同方法。

```
1 //构建并启动任务
2 CheckManager.INSTANCE
3     .getCheckTaskBuilder() //获取构建者
4     .setUid("1") //设置任务id。如不设置会随机生成uuid
5     .addCheckState(new CheckStateImp(), "test") //设置回调处理和自定义信息。如不设置将无法收到回调
6     .addLibrary(paperLibrary) //设置比对库
7     .addCheckPaper(toCheckPaper) //设置待查Paper
8     .build() //构建任务
9     .submit(); //启动任务。submit: 将任务提交到线程池中。start: 直接启动任务
```

使用教程 实例化文本对象 (Paper)



① 通过 File 对象实例化

```
1 Paper paper = Paper.load(new File("文件路径"));
```

② 通过 File 对象实例化，并指定文件类型

```
1 Paper paper = Paper.load(new File("文件路径"), FileType.TXT);
```

③ 通过文本实例化

```
1 Paper paper = Paper.load("文本内容");
```

使用教程 为文本对象补充额外信息



对于每一个 Paper 对象，包含 id、标题、作者、其它信息四项可选信息，这四项信息的设置不会影响查重结果。id 可由开发者按需标记、使用，用以唯一标识一个 Paper 对象，不会在查重报告中展示;其余三项信息设置后会在查重报告中展示。如果不设置这些信息，将默认使用文件名作为标题，其它信息为空。有以下几种方式可以对这四项信息进行设置。

① 通过 set 方法直接设置

```
1 paper.setId("001").setTitle("标题").setAuthor("作者").setInfo("其它需要展示的信息");
```

② 通过格式化文件名设置通过 File 对象加载 Paper 对象时，如果文件名中包含分隔符“@”且数量符合以下任一种规则，将自动从文件名中读取标题、作者、来源和年份信息³。

³@”只是 SDK 的默认分割符，分隔符可由开发者自行修改，修改方式参见“详细文档-接口文档-高级配置项”部分。

分割规则

- 1) 如果文件名中包含 1 个分隔符，SDK 会将文件名按分隔符分割为数组后依次读取为 id、标题，如文件名为“001@ 标题.docx”；
- 2) 如果文件名中包含 2 个分隔符，SDK 会将文件名按分隔符分割为数组后依次读取为 id、标题、作者，如文件名为“001@ 标题 @ 作者.docx”；
- 3) 如果文件名中包含 3 个分隔符，SDK 会将文件名按分隔符分割为数组后依次读取为 id、标题、作者、其它信息，如文件名为“001@ 标题 @ 作者 @ 其它需要展示的信息.docx”；
- 4) 如果文件名中不包含分隔符，SDK 会将文件名读取为标题，如文件名为“标题.docx”。

使用教程 为 Paper 设置 Payload



Payload 可用于传递上下文信息，也可以用于为 Paper 补充不希望在查重报告中展示的信息。对于每一个 Paper 对象，可以设置一个 Payload，Payload 不会展示到查重报告中，但可以用来存储额外信息或上下文信息。Payload 可以是任意对象，但该对象必须实现了 Serializable 接口。设置 payload 后，可以通过 get 方法获取。

```
1 paper.setPayload("001");  
2 Object payload = paper . getPayload () ;
```

使用教程 文档查重软件快速使用教程



① 下载、安装^a

② 获取并使用免费授权许可证启动应用程序，勾选“已完整阅读并同意遵守《软件许可协议》”。点击“免费获取”自动获取授权，然后点击“开始免费试用”即可进入主程序^b。

^a目前只支持 win7 及以上 64 位操作系统，不支持苹果系统

^b免费版支持对 8000 字以下的文档进行查重，如果超过 8000 字会被自动截断

试用授权

机器指纹：
KGY2LSRuWRc0aTNwqy8m4ixPH8D0sC 复制

评估版授权许可证：
 免费获取

商务合作请联系微信：di18810760681
QQ：1253332375

☒ 已完整阅读并同意遵守 《软件许可协议》

开始免费试用

使用教程 文档查重软件快速使用教程



③ 开始查重

选择查重方式(支持同批次间查重^a、使用比对库查重两种方式^b)、选择待查文件所在的文件夹、保存查重报告的文件夹，点击开始查重按钮即可。

^a同批次间查重:比较同一批次内是否存在横向抄袭的问题

^b使用比对库查:将文件夹中的文件和比对库中的文件比对，检查是否有与比对库中文本重复的情况。



使用教程 文档查重软件快速使用教程



查重完毕后会生成如下样式的查重报告：

文本复制检测报告单 (简洁)		
编号:2022D_1655454701930000000000000000004718888	检测时间：	2022-06-17 16:31:41
检测文献： 查重文件.docx		
作者：		
检测结果		
总文字复制比：	34.9%	单篇最大文字复制比：
34.9%		34.9%
重复字数： [14967]	总字数： [42936]	单篇最大重复字数： [14967]
总段落数： [5]	前部重合字数： [11203]	疑似段落最大重合字数： [4160]
疑似段落数： [5]	后部重合字数： [3764]	疑似段落最小重合字数： [982]
34.7%	(4007) 查重文件.docx_第1部分 (总11535字)	
35.2%	(4160) 查重文件.docx_第2部分 (总11826字)	
35.1%	(3036) 查重文件.docx_第3部分 (总8653字)	
34.6%	(2782) 查重文件.docx_第4部分 (总8050字)	
34.2%	(982) 查重文件.docx_第5部分 (总2872字)	
<div><div></div><div></div></div> <div>(注释：无问题部分文字复制部分)</div>		
1. 查重文件.docx_第1部分		总字数： 11535
相似文献列表		文字复制比： 34.7% (4007)
1	基础比对库文件 (一) .pdf	34.7% (4007)
2. 查重文件.docx_第2部分		总字数： 11826
相似文献列表		文字复制比： 35.2% (4160)
1	基础比对库文件 (一) .pdf	35.2% (4160)
3. 查重文件.docx_第3部分		总字数： 8653
相似文献列表		文字复制比： 35.1% (3036)
1	基础比对库文件 (一) .pdf	35.1% (3036)
4. 查重文件.docx_第4部分		总字数： 8050
相似文献列表		文字复制比： 34.6% (2782)
1	基础比对库文件 (一) .pdf	34.6% (2782)
5. 查重文件.docx_第5部分		总字数： 2872
相似文献列表		文字复制比： 34.2% (982)
1	基础比对库文件 (一) .pdf	34.2% (982)
说明： 总文字复制比：被检测文章总重合字数在总字数中所占的比例。		

简洁查重报告

文本复制检测报告单 (全文标明引文)		
编号:2022D_1655454701930000000000000000004718888	检测时间：	2022-06-17 16:31:41
检测文献： 查重文件.docx		
作者：		
检测结果		
总文字复制比：	34.9%	单篇最大文字复制比：
34.9%		34.9%
重复字数： [14967]	总字数： [42936]	单篇最大重复字数： [14967]
总段落数： [5]	前部重合字数： [11203]	疑似段落最大重合字数： [4160]
疑似段落数： [5]	后部重合字数： [3764]	疑似段落最小重合字数： [982]
34.7%	(4007) 查重文件.docx_第1部分 (总11535字)	
35.2%	(4160) 查重文件.docx_第2部分 (总11826字)	
35.1%	(3036) 查重文件.docx_第3部分 (总8653字)	
34.6%	(2782) 查重文件.docx_第4部分 (总8050字)	
34.2%	(982) 查重文件.docx_第5部分 (总2872字)	
<div><div></div><div></div></div> <div>(注释：无问题部分文字复制部分)</div>		
1. 查重文件.docx_第1部分		总字数： 11535
相似文献列表		文字复制比： 34.7% (4007)
1	基础比对库文件 (一) .pdf	34.7% (4007)
原文内容		
尊敬的各位老师：		
大家好。我是汇报人盛荣辉，就读于北京交通大学。今天由我来分享我们在 acm mm 2021 上的工作：bridge net: a learning network of depth map super-resolution and monocular depth estimation。		
消费级深度相机的普及为获取深度图像提供了便利。但由于成像能力的限制，深度图像的分辨率通常较低。面对诸多应用领域对高质量深度图像的需求，深度图像超分辨率重建技术获得了广泛关注。深度图像超分辨率重建是在深度相机不变的前提下，通过算法重建出高质量、高分辨率的深度图像。随着深度学习的快速发展，该任务的性能得到了很大提升。		
在实际应用中，高分辨率彩色图像易于获得，且与深度图像具有结构相似性，因而可以为深度图像超分辨率重建提供先验信息。现有的颜色指导的深度图像超分辨率重建算法通常需要一个分支来从彩色图像获取信息，然后用于指导重建分支。但彩色图像与深度图像的边缘并非一一对应，这种不一致性可能会造成纹理复制等问题。为了寻找解决方案，我们将目光聚焦在单目深度估计。单目深度估计实现了彩色图像到深度图像的跨模态转换，因而面向单目深度估计学习到的彩色特征更适合指导深度图像超分辨率重建。在网络设计方面，为了替换现有方法中的颜色分支，拟采用多任务学习的方式进行不同任务间的信息传递。且单目深度估计和深度图像超分辨率重建联合学习无需引入额外的监督信息。		
基于理论分析和实验验证，我们设计了联合学习网络bridgenet，该网络具有高度的可移植性，可以为关联深度图像超分辨率重建和单目深度估计提供范例。在探讨任务相关性后，提出了用于联合学习的高频注意力桥和内容引导桥。在不引入监督信息的前提下，我们的方法在多个基准数据集上达到了具有竞争力的性能。		

全文标红查重报告

© 2022年 北京芯锋科技有限公司

文本复制检测报告单 (原文对照)		
编号:2022D_1655454701930000000000000000004718888	检测时间：	2022-06-17 16:31:41
检测文献： 查重文件.docx		
作者：		
检测结果		
总文字复制比：	34.9%	单篇最大文字复制比：
34.9%		34.9%
重复字数： [14967]	总字数： [42936]	单篇最大重复字数： [14967]
总段落数： [5]	前部重合字数： [11203]	疑似段落最大重合字数： [4160]
疑似段落数： [5]	后部重合字数： [3764]	疑似段落最小重合字数： [982]
34.7%	(4007) 查重文件.docx_第1部分 (总11535字)	
35.2%	(4160) 查重文件.docx_第2部分 (总11826字)	
35.1%	(3036) 查重文件.docx_第3部分 (总8653字)	
34.6%	(2782) 查重文件.docx_第4部分 (总8050字)	
34.2%	(982) 查重文件.docx_第5部分 (总2872字)	
<div><div></div><div></div></div> <div>(注释：无问题部分文字复制部分)</div>		
1. 查重文件.docx_第1部分		总字数： 11535
相似文献列表		文字复制比： 34.7% (4007)
1	基础比对库文件 (一) .pdf	34.7% (4007)
原文内容	相似内容来源	
1 此处有 20 字相似 深度图像提供了便利。但由于 成像能力的限制，深度图像的分辨率通常较低。面对诸多应用领域对高质量	基础比对库文件 (一) .pdf 。但是，由于当前深度相机 成像能力的限制，深度图像的分辨率通常较低，无法与同场景的高分辨率彩	
2 此处有 20 字相似 ，深度图像的分辨率通常较低。面对诸多应用领域对高质量深度图像的需求，深度图像超分辨率重建技术	基础比对库文件 (一) .pdf 的高分辨率彩色图像相匹配。面对诸多应用领域对高质量深度图像的需求 [3-5]，深度图像超分辨	
3 此处有 13 字相似 领域对高质量深度图像的需求，深度图像超分辨率重建技术 获得了广泛关注。深度图像超	基础比对库文件 (一) .pdf 量深度图像的需求[3-5]，深度图像超分辨率重建技术作为 解决方案获得了越来越	
4 此处有 17 字相似 相机不变的前提下，通过算法 重建出高质量、高分辨率的深度图像。随着深度学习的快速发展，该	基础比对库文件 (一) .pdf 制图像中的随机噪声，从而 重建出高质量、高分辨率的深度图像。传统的基于滤波的方法[6-	
此处有 17 字相似	基础比对库文件 (一) .pdf	

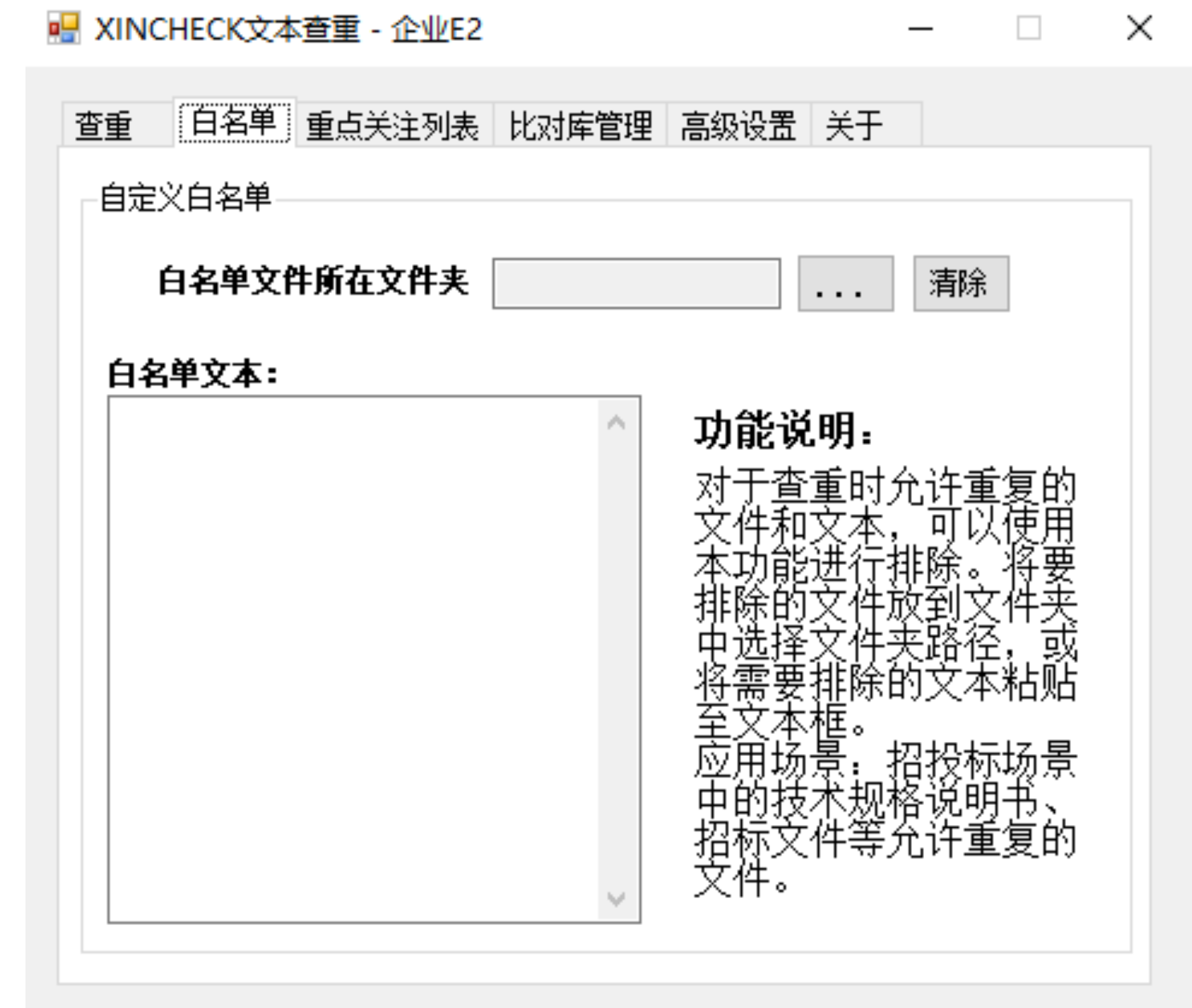
原文对照查重报告

XINCHECK® 文本查重 2022/4/28

18 北京芯锋科技有限公司

④ 排除部分文本

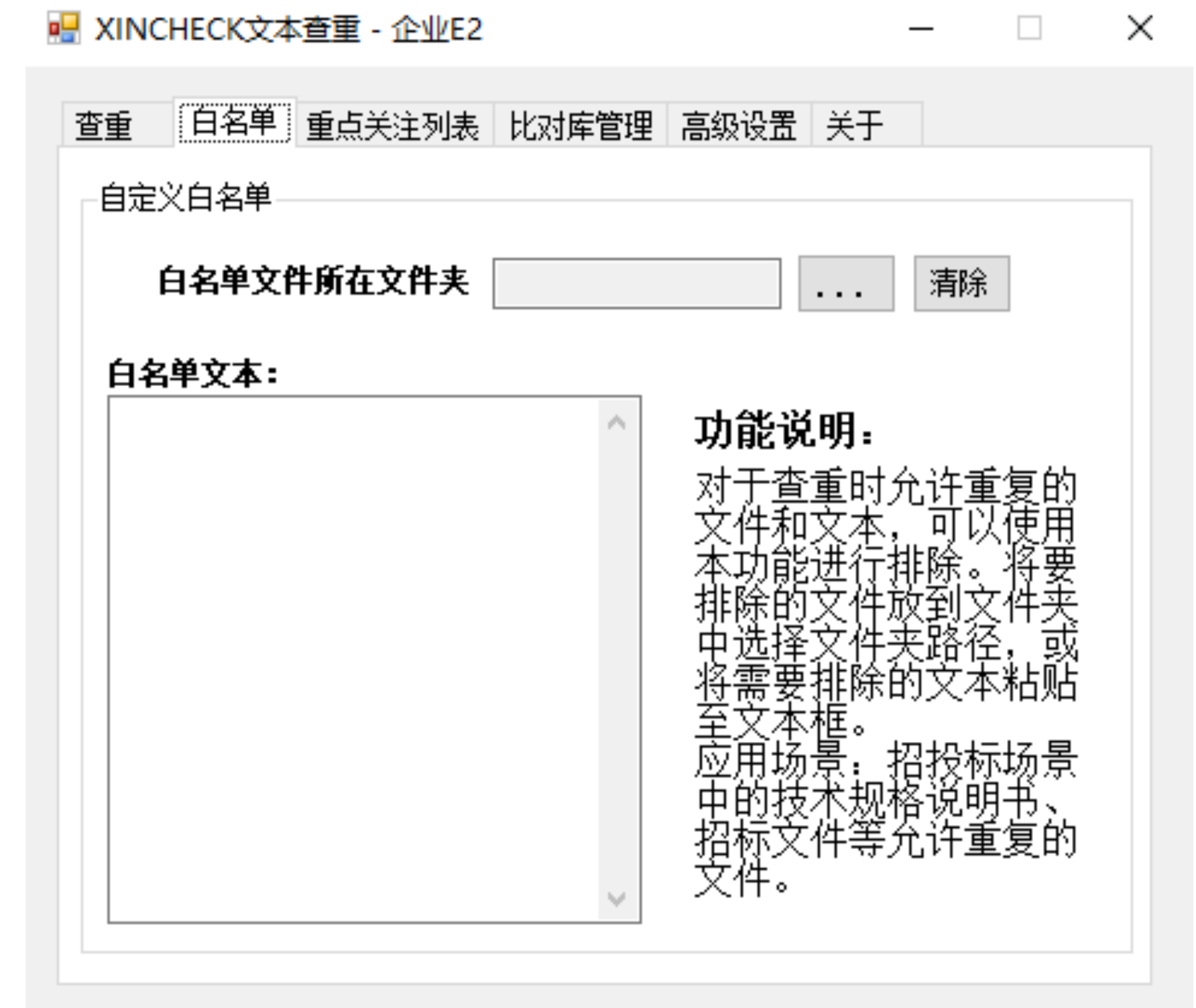
在进行标书查重或论文查重时，对于技术规格说明书、招标文件、专有名词列表中的内容，是允许重复的，XINCHECK 文档查重软件提供了白名单排除功能，将允许重复的文件或文本添加进去，则可以在最后的查重报告中进行排除，适合标书查重。



⑤ 添加重点关注关键词

在制作标书或检查串标时，地名、供应商、关联公司等，很容易在复制时被遗漏，通过将这类关键词^a 添加至重点关注列表，在查重时将以紫色和黄色重点标注这些关键词，便于工作人员检查。

^a重点关注列表中的关键词长度不能超过 8 个字符



使用教程 文档查重软件快速使用教程



查重报告中紫色和黄色标注的部分即为命中重点关注列表的关键词(重复部分命中标记为紫色，非重复部分命中标记为黄色)。

文本复制检测报告单

(全文标明引文)

编号:2022D_165545470193000000000000000004718888

检测时间:2022-06-17 16:31:41

检测文献:查重文件.docx

作者:

检测结果

总文字复制比:34.9%

单篇最大文字复制比:34.9%

重复字数:[14967]

总字数:[42936]

单篇最大重复字数:[14967]

总段落数:[5]

前部重合字数:[11203]

疑似段落最大重合字数:[4160]

疑似段落数:[5]

后部重合字数:[3764]

疑似段落最小重合字数:[982]

34.7% (4007) 查重文件.docx_第1部分 (总11535字)

35.2% (4160) 查重文件.docx_第2部分 (总11826字)

35.1% (3036) 查重文件.docx_第3部分 (总8653字)

34.6% (2782) 查重文件.docx_第4部分 (总8050字)

34.2% (982) 查重文件.docx_第5部分 (总2872字)

(注释:无问题部分文字复制部分)

1. 查重文件.docx_第1部分

总字数:11535

相似文献列表

文字复制比:34.7% (4007)

1基础比对库文件(一).pdf

34.7% (4007)

原文内容

[illegible]

文本复制检测报告单

(全文标明引文)

编号:2022D_1655456762479000000000000000007740378

检测时间:2022-06-17 17:06:02

检测文献:查重文件.docx

作者:

检测结果

总文字复制比:0%

单篇最大文字复制比:0%

重复字数:[0]

总字数:[42936]

单篇最大重复字数:[0]

总段落数:[5]

前部重合字数:[0]

疑似段落最大重合字数:[0]

疑似段落数:[0]

后部重合字数:[0]

疑似段落最小重合字数:[0]

0%

(0)查重文件.docx第1部分(总11535字)

0%

(0)查重文件.docx第2部分(总11826字)

0%

(0)查重文件.docx第3部分(总8653字)

0%

(0)查重文件.docx第4部分(总8050字)

0%

(0)查重文件.docx第5部分(总2872字)

(注释:无问题部分文字复制部分)

1. 查重文件.docx_第1部分

总字数:11535

相似文献列表

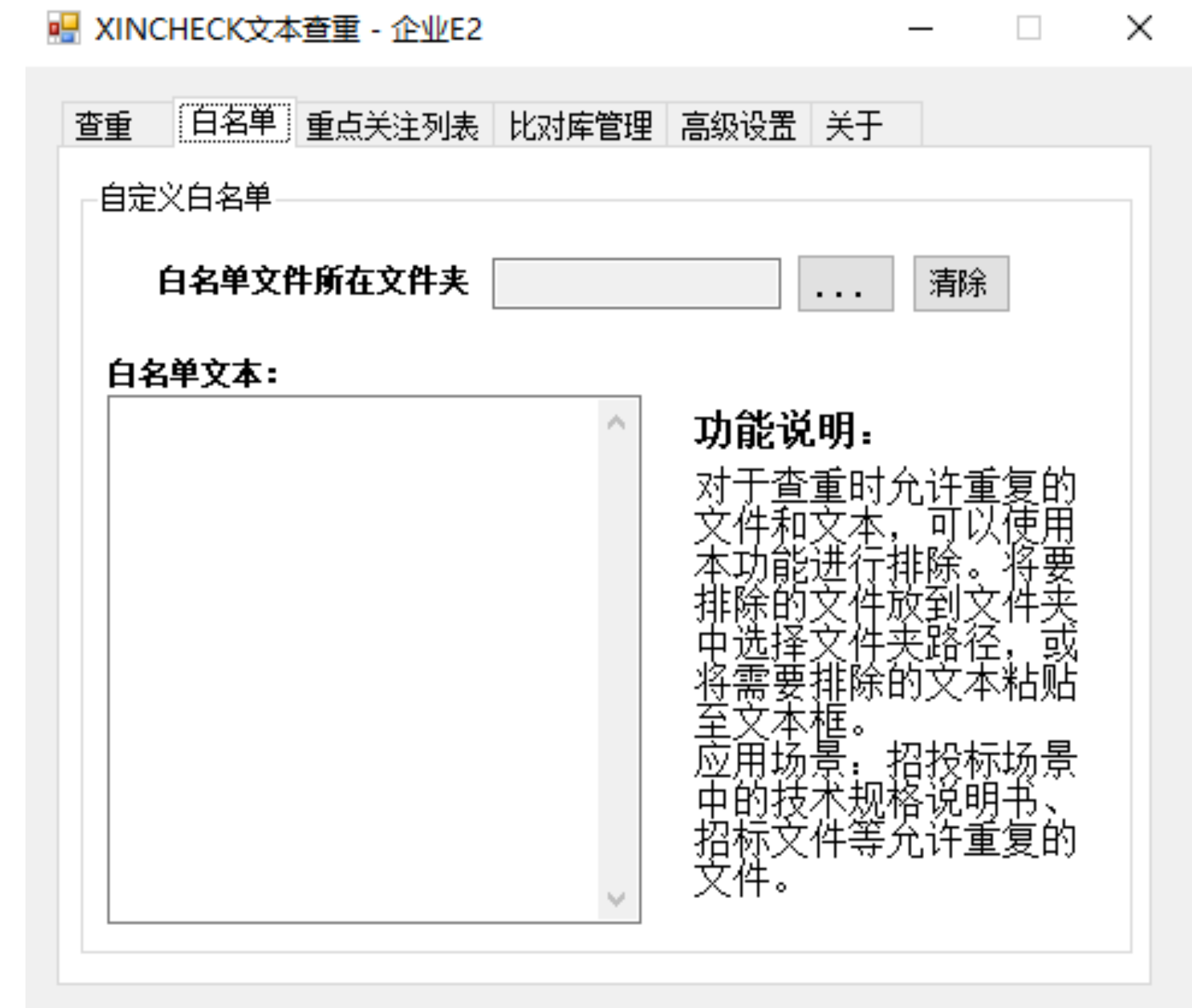
文字复制比:0%(0)

原文内容

尊敬的各位老师:

⑥ 其他功能

支持自建比对库、切换查重算法(文本指纹算法/分句语义算法)、选择查重严格性、对文档属性和图片查重并生成图片重复报告等功能。



使用教程 查重 SDK 更新日志



- 2021.11.13 发布 0.5.5 beta 版

增加了实验特性:图片查重并生成图片重复报告。目前仅面向付费用户进行公测，需要通过高级配置项打开该功能。【评估版可以进行该项配置但功能不生效】

- 2021.11.25 发布 0.5.5

修复若干 bug;支持基于语义的查重(实验性)。

- 2021.12.12 发布 0.5.5 patch1

修复超过字符限制、实际不相同、前 n 字相同的文件被判定为相同文件的问题。

使用教程 查重 SDK 更新日志



- 2021.11.13 发布 0.5.5 beta 版
增加了实验特性:图片查重并生成图片重复报告。目前仅面向付费用户进行公测，需要通过高级配置项打开该功能。【评估版可以进行该项配置但功能不生效】
- 2021.11.25 发布 0.5.5
修复若干 bug;支持基于语义的查重(实验性)。
- 2021.12.12 发布 0.5.5 patch1
修复超过字符限制、实际不相同、前 n 字相同的文件被判定为相同文件的问题。

- 2021.11.13 发布 0.5.5 beta 版
增加了实验特性:图片查重并生成图片重复报告。目前仅面向付费用户进行公测，需要通过高级配置项打开该功能。【评估版可以进行该项配置但功能不生效】
- 2021.11.25 发布 0.5.5
修复若干 bug;支持基于语义的查重(实验性)。
- 2021.12.12 发布 0.5.5 patch1
修复超过字符限制、实际不相同、前 n 字相同的文件被判定为相同文件的问题。

使用教程 查重软件更新日志



- 2021.10.22 发布 0.9.31

修复一个文件属性查重的 bug。

- 2021.11.15 发布 0.9.32

支持切换文本块指纹、分句语义两种查重算法;支持对图片查重并生成图片重复报告(实验性)。

- 2021.12.11 发布 0.9.33

修复超过字符限制、实际不相同、前 n 字相同的文件被判定为相同文件的问题;修复横向查重多个相同文件只生成一份查重报告的问题。



- 2021.10.22 发布 0.9.31

修复一个文件属性查重的 bug。

- 2021.11.15 发布 0.9.32

支持切换文本块指纹、分句语义两种查重算法;支持对图片查重并生成图片重复报告(实验性)。

- 2021.12.11 发布 0.9.33

修复超过字符限制、实际不相同、前 n 字相同的文件被判定为相同文件的问题;修复横向查重多个相同文件只生成一份查重报告的问题。



- 2021.10.22 发布 0.9.31
修复一个文件属性查重的 bug。
- 2021.11.15 发布 0.9.32
支持切换文本块指纹、分句语义两种查重算法;支持对图片查重并生成图片重复报告(实验性)。
- 2021.12.11 发布 0.9.33
修复超过字符限制、实际不相同、前 n 字相同的文件被判定为相同文件的问题;修复横向查重多个相同文件只生成一份查重报告的问题。

使用教程 查重 SDK



许可证版本	企业E1	企业E2	企业E3	企业E4
授权时长	当前版本永久授权	当前版本永久授权	当前版本永久授权	当前版本永久授权
发布地点数	1个。支持离线部署	同一地级市内3个。支持离线部署	同一地级市内不限。支持离线部署	不限地级市、发布地点。支持离线部署
测试服务器	数量不限	数量不限	数量不限	数量不限
字符数限制	比对库5亿内；白名单2000内；重点关注列表200内	比对库30亿内；白名单3000内；重点关注列表300内	比对库不限；白名单5000内；重点关注列表500内	均不限
待查文本限制	均不限	均不限	均不限	均不限
查重报告样式	支持自定义	支持自定义	支持自定义	支持自定义
高级配置项及特性	支持全部	支持全部	支持全部	支持全部
水印及广告	无	无	无	无
软件更新	1年	1年	1年	1年
技术支持	180天邮件支持。30天在线技术支持	1年邮件支持。45天在线技术支持	1年邮件支持。45天在线技术支持	1年邮件支持。90天在线技术支持
价格	¥12800	¥29800	¥49800	议价 ¹

使用教程 查重软件



许可版本	个人P1	企业E1	企业E2	企业E3
授权时长	当前版本永久授权	当前版本永久授权	当前版本永久授权	当前版本永久授权
发布地点数	1个。需要 联网授权 使用	同一地级市内3个。支持 内网 离线 使用	同一地级市内30个。支持 内网 离线 使用	不限地级市。 发布地点数3000个。支持 内网 离线 使用
比对库限制	总字数400万以内，篇数上限30	总字数5亿以内，篇数上限1000	均不限	均不限
待查文本限制	单篇40万字以内。单次查重篇数10以内、400万字内。	单篇不限。 单次查重篇数20以内	均不限	均不限
其它字符数限制	白名单1000字符内；重点关注列表100字符内	白名单2000字符内；重点关注列表150字符内	白名单3000字符内；重点关注列表300字符内	均不限
水印及广告	无	无	无	无
软件更新	1个月	1年	1年	1年
技术支持	30天邮件支持	180天邮件支持；1次使用培训；30天在线技术支持	1年邮件支持；1次使用培训；30天在线技术支持	1年邮件支持；1次使用培训；90天在线技术支持
价格	¥1980	¥12800	¥29800	20万起，详请议价

Q & A Thank you



1 Question

查重性能如何？

- 内存占用：SDK 启动后会占用约 50MB 常驻内存。比对库中每一千万字次需要约 150MB 常驻内存。此外，在加载比对库时解析 word 和 pdf 文档时所需的动态内存约 12GB，动态内存存在 GC 后会被回收。
- CPU 占用：在没有任何任务运行时，SDK 不会占用 CPU。在进行比对库导入时，如果开启了多线程导入通常会占用约 50% 的 CPU，如果未开启多线程导入通常会占用一个 CPU 核心。在进行查重任务时，每个待查文本的文本段将占用一个 CPU 核心，SDK 默认将每 1 万字划分为一个文本段。如果您需要预留部分服务器的 CPU 资源用于其它服务，可以对 SDK 进行并发配置，限制 SDK 的 CPU 占用。
- 查重速度：待查文本每 1 万字、比对库每 1 千万字次的情况下秒级出结果。时间随数据量线性增加。

2 Question

SDK 的发布地点是什么意思？

终端的定义为：“具备存储功能且可以进行逻辑计算的实体或虚拟设备，包括但不限于服务器、台式电脑、手提电脑、手持移动终端、电视机、机顶盒、单片机、开发板等，以及使用虚拟化技术生成的上述实例(如虚拟机等)。”发布地点的定义为：“嵌入许可软件的应用程序用于商业用途时的分发地点。如果是桌面应用程序 (C/S)，每个终端均被视为一个发布地点。若是服务器应用程序，分发到内网或私有云平台，且许可软件提供的能力通过公网不可以直接或间接访问，则每个终端均被视为一个发布地点；分发到公有云平台(如：阿里云、腾讯云、华为云等)，且许可软件提供的能力通过公网可以直接或间接访问，则被视为无限发布地点；分发到有公云平台且许可软件提供的能力无法通过公网直接或间接访问，则每个终端被视为一个发布地点。

3 Question

桌面端的发布地点是什么意思？

可以简单理解为装机数量(即安装到几台电脑上)。详细定义如下：

终端的定义为：“具备存储功能且可以进行逻辑计算的实体或虚拟设备，包括但不限于服务器、台式电脑、手提电脑、手持移动终端、电视机、机顶盒、单片机、开发板等，以及使用虚拟化技术生成的上述实例(如虚拟机等)。”

发布地点的定义为：“每一个安装 XINCHECK 文档查重桌面端软件的终端即为一个发布地点。如果终端的主要硬件(主板和网卡)发生变化，则视为产生了一个新的发布地点。”

Q & A 常见问题



4 Question

在哪些领域有过合作案例？
有过电子招标系统、科研管理平台、论文查重系统、舆情系统等合作案例。

文本查重

文本指纹算法：全匹配，按设置的最低阈值字符数匹配，比对库中存在即可判定为重复

分句语义算法：先分句，然后计算句间相似度，句间相似度超过阈值判定为重复

图像查重

图像指纹算法：计算图像的均值哈希，不同图像哈希的汉明距离若超过阈值判定为重复