

ff55fff的博客

目录视图

摘要视图

RSS 订阅

个人资料



ff55fff

关注

发私信

访问：5581次

积分：252

等级：BLOG 2

排名：千里之外

原创：0篇

转载：110篇

译文：0篇

评论：0条

文章搜索

文章存档

2017年09月 (43)

2017年08月 (65)

2017年07月 (2)

阅读排行

[Python][小知识] Python字符... (190)

开源分享 Unity3d客户端与C#... (175)

64位win10+cuda8.0+vs2013+cu... (167)

angular 4 实战开发--安卓兼容... (166)

deepin系统下部署Python3.5的... (149)

nodejs-ORM 操作数据库中间... (106)

【Spring】16、注解事务 @Tr... (92)

Redis笔记——技术点汇总 (91)

共享洗车、洗车O2O，都将被... (90)

fread读取文件（二进制文件） (87)

评论排行

Calico 的网络结构是什么？ - ... (0)

fread读取文件（二进制文件） (0)

cydia闪退解决方案 (0)

异步赠书：10月Python畅销书升级

【线路图】人工智能到底学什么？！

程序员9月书讯

节后

PyQt5、Kotlin（评论送书）

Android辅助功能原理与基本使用详解-AccessibilityService

2017-08-31 18:13

47人阅读

评论(0)

收藏

举报

目录(?)

[+]

辅助功能原理与基本使用详解

本文主要介绍辅助功能的使用

1. 辅助功能基本原理

2. 辅助功能基本配置和框架搭建

3. 辅助功能实战解析

一、辅助功能基本原理

辅助功能（AccessibilityService）其实是一个Android系统提供给的一种服务，本身是继承Service类的。这个服务提供了增强的用户界面，旨在帮助残障人士或者可能暂时无法与设备充分交互的人们。

从开发者的角度看，其实就是提供两种功能：查找界面元素，实现模拟点击。实现一个辅助功能服务要求继承AccessibilityService类并实现它的抽象方法。自定义一个服务类AccessibilitySampleService（这个命名可以随意），继承系统的AccessibilityService并覆写onAccessibilityEvent和onInterrupt方法。编写好服务类之后，在系统配置文件（AndroidManifest.xml）中注册服务。完成前面两个步骤就完成了基本发辅助功能服务注册与配置，具体的功能实现需要在onAccessibilityEvent中完成，根据onAccessibilityEvent回调方法传递过来的AccessibilityEvent对象可以对事件进行过滤，结合AccessibilitySampleService本身提供的查找节点与模拟点击相关的接口即可实现权限节点的查找与点击。

AccessibilityService

```
public abstract class AccessibilityService
extends Service

java.lang.Object
├── android.content.Context
│   ├── android.content.ContextWrapper
│   │   ├── android.app.Service
│   │   │   └── android.accessibilityservice.AccessibilityService
```

二、辅助功能基本配置和框架搭建

创建自定义辅助功能服务类

http://blog.csdn.net/ff55fff/article/details/77750750

1/7

C++语句	(0)
C++异常处理	(0)
C++作用域与标识符的可见性	(0)
ASP.NET Web Pages &#8211; ...	(0)
angular 4 实战开发--安卓兼容...	(0)
android-minSdkVersion 详解	(0)
64位win10+cuda8.0+vs2013+cu...	(0)

推荐文章

- \* 【观点】第二十三期：程序员应该如何积累财富？
- \* Android检查更新下载安装
- \* 动手打造史上最简单的 RecyclerView 侧滑菜单
- \* TCP网络通讯如何解决分包粘包问题
- \* SDCC 2017之大数据技术实战线上峰会
- \* 快速集成一个视频直播功能

最新评论

PHP 文件处理  
wmz545546 :讲的太好了，棒棒哒！！！

Java继承中方法的覆盖和重载  
long0801 :感谢博主分享

Java中文件的压缩处理  
dadahahaha :不错的文章

fread读取文件（二进制文件）  
kof2019 :博主写得太好了。

C++语句  
44pq 学习了。

C++异常处理  
bmw601055 :讲的太好了，棒棒哒！！！！

C++作用域与标识符的可见性  
daguguji :学习了~ 谢啦。

64位win10+cuda8.0+vs2013+cuDNN V5下Caffe的编译安装教程并配置matlab2014a 接口  
ogog123 :讲的太好了。

```
import android.accessibilityservice.AccessibilityService;

import android.view.accessibility.AccessibilityEvent;

import com.accessibility.utils.AccessibilityLog;

public class AccessibilitySampleService extends AccessibilityService {

    @Override

    protected void onServiceConnected() {

        super.onServiceConnected();

    }

    @Override

    public void onAccessibilityEvent(AccessibilityEvent event) {

        // 此方法是在主线程中回调过来的，所以消息是阻塞执行的

        // 获取包名

        String pkgName = event.getPackageName().toString();

        int eventType = event.getEventType();

        // AccessibilityOperator封装了辅助功能的界面查找与模拟点击事件等操作

        AccessibilityOperator.getInstance().updateEvent(this, event);

        AccessibilityLog.printLog("eventType: " + eventType + " pkgName: " + pkgName);

        switch (eventType) {

            case AccessibilityEvent.TYPE_WINDOW_STATE_CHANGED:

                break;

        }

    }

    @Override

    public void onInterrupt() {

    }

}
```

注册辅助功能服务

```
// 注册辅助功能服务

<service android:name=".AccessibilitySampleService"

    android:label="@string/accessibility_tip"

    android:exported="true"

    android:permission="android.permission.BIND_ACCESSIBILITY_SERVICE"

    android:process=":BackgroundService">

    <intent-filter>
```

关闭

```

        <action android:name="android.accessibilityservice.AccessibilityService" />
    </intent-filter>

    // 通过xml文件完成辅助功能相关配置，也可以在onServiceConnected中动态配置

    <meta-data
        android:name="android.accessibilityservice"
        android:resource="@xml/accessibility_config"/>
</service>

```

上面android:label="@string/accessibility\_tip"是配置此辅助功能服务在系统辅助功能页面的提示文字。

**accessibility\_config文件内容如下：**

```

<?xml version="1.0" encoding="utf-8"?>
<accessibility-service xmlns:andro
    android:accessibilityEventTypes="typeAllMask"
    android:accessibilityFeedbackType="feedbackGeneric"
    android:canRetrieveWindowContent="true"
    android:description="@string/accessibility_desc"
    android:notificationTimeout="100" />

```

### 跳转到系统辅助功能页面，开启辅助功能服务

完成上面配置之后，辅助功能服务就注册成功了，在系统辅助功能页面就能找到这个服务，但是默认是关闭的，也就是说，这个服务要开始为我们服务，还需要去系统界面开启那个开关。下面是跳转到辅助功能页面的代码，跳转过去之后，手动点击开关按钮。开关打开之后，这个辅助功能服务就开始工作了，系统开始回调onAccessibilityEvent方法。我们可以在onAccessibilityEvent方法中处理查找节点与点击操作。

```

public class OpenAccessibilitySettingHelper {

    private static final String ACTION = "action";

    private static final String ACTION_START_ACCESSIBILITY_SETTING =
        "action_start_accessibility_setting";

    public static void jumpToSettingPage(Context context) {
        try {
            Intent intent = new Intent(context, AccessibilityOpenHelperActivity.class);
            intent.putExtra(ACTION, ACTION_START_ACCESSIBILITY_SETTING);
            intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
            context.startActivity(intent);
        } catch (Exception ignore) {}
    }
}

```

下图是小米手机开启辅助功能的界面

关闭



三、辅助功能实战解析

实现界面自动点击操作，动画有点模糊，将就看吧



界面节点查找与模拟点击

AccessibilityOperator封装了辅助功能的界面查找与模拟点击事件等操作，下面介绍几个关键技术点。

界面节点查找操作

AccessibilityNodeInfo提供两种查找View节点的方法

1. 根据View的ID精确查找，但是要求SDK\_INT >= 18才能用

```
/**
 * 根据View的ID搜索符合条件的节点,精确搜索方式;
 * 这个只适用于自己写的界面，因为ID可能重复
 * api要求18及以上
 * @param viewId
 */
public List<AccessibilityNodeInfo> findNodesById(String viewId) {
```

关闭

```
AccessibilityNodeInfo nodeInfo = getRootNodeInfo();

if (nodeInfo != null) {

    if (Build.VERSION.SDK_INT >= 18) {

        return nodeInfo.findAccessibilityNodeInfosByViewId(viewId);

    }

}

return null;

}
```

## 2. 根据View的Text文本进行模糊查找

```
/**
 * 根据Text搜索所有符合条件的节点, 模糊搜索方式
 */
public List<AccessibilityNodeInfo> findNodesByText(String text) {

    AccessibilityNodeInfo nodeInfo = getRootNodeInfo();

    if (nodeInfo != null) {

        return nodeInfo.findAccessibilityNodeInfosByText(text);

    }

    return null;

}
```

## 模拟界面操作

### 1. 普通的View事件模拟 ( ACTION\_CLICK )

```
private boolean performClick(List<AccessibilityNodeInfo> nodeInfos) {

    if (nodeInfos != null && !nodeInfos.isEmpty()) {

        AccessibilityNodeInfo node;

        for (int i = 0; i < nodeInfos.size(); i++) {

            node = nodeInfos.get(i);

            // 获得点击View的类型

            AccessibilityLog.printLog("View类型 : " + node.getClassName());

            // 进行模拟点击

            if (node.isEnabled()) {

                return node.performAction(AccessibilityNodeInfo.ACTION_CLICK);

            }

        }

    }

    return false;

}
```

### 2. 全局事件模拟 ( 返回键 : AccessibilityService.GLOBAL\_ACTION\_BACK )

关闭

```
public boolean clickBackKey() {  
    return performGlobalAction(AccessibilityService.GLOBAL_ACTION_BACK);  
}  
  
private boolean performGlobalAction(int action) {  
    return mAccessibilityService.performGlobalAction(action);  
}
```

源码地址

https://github.com/PopFisher/AccessibilitySample

顶 0 踩 0

- 上一篇 使用Apriori进行关联分析（二）
- 下一篇 Spring 学习笔记01

相关文章推荐

- Android辅助功能AccessibilityService的使用
- Presto服务治理与架构优化在京东的实践应用--王...
- Android 无障碍辅助功能AccessibilityService（1）
- 【免费直播】Python最佳学习路线--韦玮
- Android 无障碍辅助功能AccessibilityService（2）
- JS-SDK开发与微信支付
- 使用Android辅助服务AccessibilityService实现的微...
- Spring Cloud微服务真实场景实战解析
- Android辅助功能AccessibilityService与抢红包辅助
- 10小时深入掌握 Kubernetes
- Android辅助功能AccessibilityService与抢红包辅助
- JDK9新特性
- 浅谈辅助功能 AccessibilityService
- Android中AccessibilityService（辅助类服务）的用法
- AccessibilityService辅助类用法（Android 中的另类...
- 辅助服务类：AccessibilityService使用

今日优选



红包玩命送!!!  
低价促销

去看看



红包玩命送!!!

去看看



红包玩  
专业实

去看

查看评论

暂无评论

关闭

您还没有登录,请[登录](#)或[注册](#)

\* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

