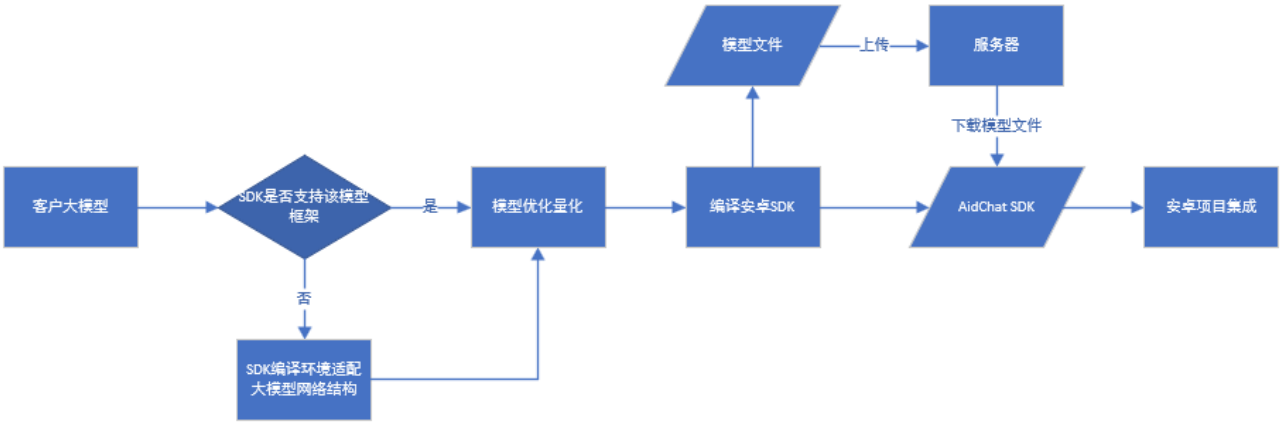
AidChat SDK 开发者文档

介绍

AidChat SDK是阿加犀智能科技有限公司封装的针对LLM（大模型）在边缘端场景推理部署的AI推理SDK，能够直接在单块SoC上本地运行多种主流大模型。

AidChat SDK包含了大模型编译到SDK封装的整个流程，如下图所示：



> SDK预置了6种主流大模型，开发者可以直接使用SDK进行推理。如果需要通过SDK部署自己的大模型，需要根据上图的流程，针对该模型进行编译和SDK封装。

大模型编译封装SDK请联系：AidLux商务联系方式

SDK设备授权方式

SDK授权采用第一次在线授权验证（需联网），验证通过后会生成授权文件至本地，以后可以离线授权使用。

授权信息获取方式

- 使用手机号注册一个AidLux平台账号：AidLux账号注册地址
- 联系我们激活该账号并获取User ID：AidLux商务联系方式

> AidChat SDK目前处于内测阶段，暂未开放给所有注册用户使用，需要联系我们激活，注册激活的用户有免费三台设备授权数量

SDK集成指南

硬件及系统要求

目前SDK支持7B以下的大模型部署，针对不同尺寸的大模型对硬件RAM的要求也不同。

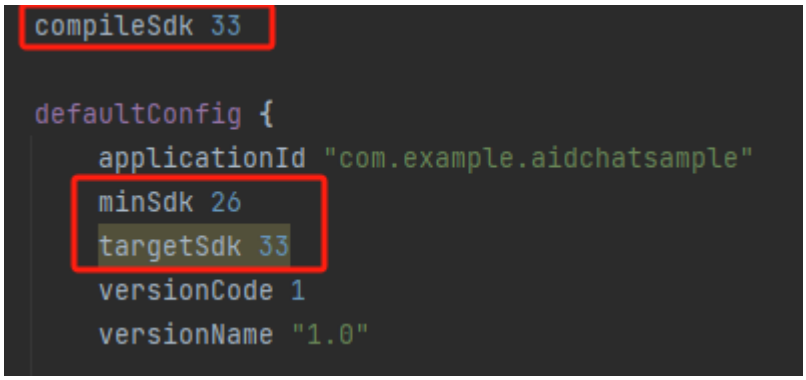
系统：安卓10及以上

3B大模型：RAM ≥ 8GB

7B大模型：RAM ≥ 12GB

项目集成步骤

1. 保证项目build.gradle文件中的compileSdk=33，minSdk=26，targetSdk=33



2. 导入SDK：在android studio 的project 视图下的主应用app下新建libs文件夹，并将AidliteSDK-release.aar导入到libs
3. 添加dependencies依赖：在主应用app 目录下的build.gradle的的dependencies下添加依赖

```
implementation fileTree(include: ['*.jar', "*.aar"], dir: 'libs')
implementation "com.blankj:utilcodex:1.30.6"
implementation 'com.elvishew:xlog:1.10.1'
//网络通信
implementation 'com.squareup.retrofit2:retrofit:2.9.0'
implementation 'com.squareup.retrofit2:converter-gson:2.9.0'
implementation "com.squareup.okhttp3:logging-interceptor:4.9.0"
implementation 'org.jetbrains.kotlinx:kotlinx-coroutines-android:1.3.9'

implementation 'com.google.code.gson:gson:2.10.1'
implementation 'androidx.activity:activity-compose:1.7.1'
implementation platform('androidx.compose:compose-bom:2022.10.00')
implementation 'androidx.lifecycle:lifecycle-viewmodel-compose:2.6.1'
implementation 'androidx.compose.ui:ui'
implementation 'androidx.compose.ui:ui-graphics'
implementation 'androidx.compose.ui:ui-tooling-preview'
implementation 'androidx.compose.material3:material3:1.1.0'
implementation 'androidx.compose.material:material-icons-extended'
implementation 'net.java.dev.jna:jna:5.10.0'
```

4. 在Android视图下的app/mainifests/AndroidManifest.xml文件中添加权限和属性配置

```
<uses-permission android:name="android.permission.READ_PRIVILEGED_PHONE_STATE" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
```

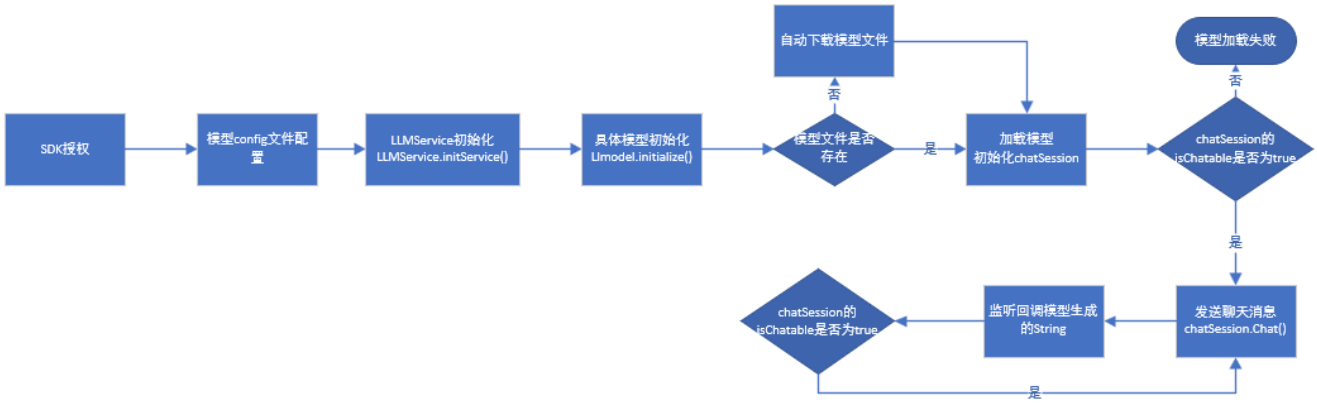
```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

5. 在Android视图下的app/mainifests/AndroidManifest.xml 设置 usesCleartextTraffic

```
<application
    android:allowBackup="true"
    android:dataExtractionRules="@xml/data_extraction_rules"
    android:fullBackupContent="@xml/backup_rules"
    android:icon="@drawable/logo"
    android:label="AidChatSample"
    android:name=".LLMApplication"
    android:roundIcon="@drawable/logo"
    android:supportsRtl="true"
    android:theme="@style/Theme.AidChatSample"
    android:usesCleartextTraffic="true">
    <activity
```

开发步骤

流程图



模型文件下载地址

链接：https://pan.baidu.com/s/1KbFrDg-KRsj9RmAtYpd4Qg?pwd=cqd9
提取码：cqd9

配置app-config.json文件

在安卓工程assets目录下新建app-config.json文件，app-config.json文件指定了需要加载的模型名称以及下载模型的URL地址

```
{
  "model_list": [
    {
      "model_url": "model resource url",
```

```
      "local_id": "Llama-2-7b-chat-hf"
    },
    {
      "model_url": "model resource url",
      "local_id": "Qwen-7B-Chat"
    },
    {
      "model_url": "model resource url",
      "local_id": "Baichuan2-7B-Chat-LLaMAfied"
    },
    {
      "model_url": "model resource url",
      "local_id": "chatglm2-6b"
    },
    {
      "model_url": "model resource url",
      "local_id": "RedPajama-INCITE-Chat-3B-v1"
    },
    {
      "model_url": "model resource url",
      "local_id": "vicuna-7b-v1.1"
    }
  ]
}
```

> SDK暂不提供公网的模型下载地址，只提供模型文件的百度网盘链接，开发者可以自行下载后放在自己的服务器地址上，并填入model_url中。前期开发过程中开发者也可以跳过模型下载步骤，直接将模型通过adb push方式放入指定目录下即可（具体操作参考SDK 安卓Demo）

AidChat SDK 接口调用流程说明

```
private List<LLModel> modelList;
// --------------------0. AidSDK authorization and initialization----------------- //

Aidlite.INSTANCE.initialize(getApplication(), "your user id", true);

// --------------------1. Initialize LLMService----------------- //

LLMService.INSTANCE.initService(getApplication(), new OnModelListLoadListener() {
    @Override
    public void onModelListLoad(@NonNull List<LLModel> list) {
            modelList = list;
        // code here
    }
});

// --------------------2. Initialize LLModel----------------- //

// Register listener
modelList.get(0).registerModelStateListener(new OnModelStateListener() {
```

```
    @Override
    public void onModelParamsDownload(@NonNull ModelInitState modelInitState, @NonNull
MutableState<Integer> progressNow, @NonNull MutableState<Integer> progressTotal) {
                // code here
    }

    @Override
    public void onClearFinish(@NonNull ModelInitState modelInitState) {
        // code here
    }

    @Override
    public void onDeteteFinish(@NonNull ModelInitState modelInitState) {
        // code here
    }
});

//model initialize, download and check model file
modelList.get(0).initialize();

// --------------------3. Load model and initialize ChatSession ----------------- //

ChatSession chatSession = LLMService.INSTANCE.loadModel(modelList.get(0), new
ChatSession.ReloadChatCallback() {
    @Override
    public void success() {
                // code here
    }

    @Override
    public void fail(@Nullable String error) {
                // code here
    }
});

// Register listener
        chatSession.registerReplyCallback(new ChatSession.ReplyCallback() {
            @Override
            public void onReply(@NonNull String reply) {
                // code here
            }
        });

// --------------------4. chat ----------------- //

chatSession.chat(message);
```

SDK安卓Demo运行步骤

导入project

1. 打开Android Studio，在菜单栏中：File->New->Import Project，选择项目导入

2. 根据自身Android Studio和项目需求修改version
3. 导入成功后点击sync按钮进行编译同步

> 如果编译时出现"ERROR: Plugin with id 'com.android.application' not found."错误，请在build.gradle文件中添加以下代码。

```
buildscript {
    repositories {
        google()
        jcenter()
    }
    dependencies {
         //版本号请根据自己的gradle插件版本号自行更改
        classpath 'com.android.tools.build:gradle:3.4.0'
        // NOTE: Do not place your application dependencies here; they belong
        // in the individual module build.gradle files
    }
}
```

4. 编译同步成功后，将有效的user ID填入SDK初始化函数中

```
Aidlite.INSTANCE.initialize(getApplication(), "your user id", true);
```

5. 点击run即可运行demo

此时APP的app-config.json文件填写的URL是空的或者无效地址，开发者可以通过两种方式置入模型文件

a. 将模型文件放入自己的服务器地址上，然后填入app-config.json的url中，再次运行APP即可通过UI点击方式下载模型

b. 将模型文件直接通过adb push方式推送至指定目录下，并修改文件夹权限，再次运行APP即可直接加载模型，无需下载

```
adb push E:\mlcllm_resource\file_encry\Llama-2-7b-chat-hf /sdcard/
adb shell
        cd /sdcard
        mv Llama-2-7b-chat-hf
/storage/emulated/0/Android/data/com.example.aidchatsample/files/
        cd /storage/emulated/0/Android/data/com.example.aidchatsample/files/
        chmod -R 777 Llama-2-7b-chat-hf
```

常见问题

1. adb push模型文件后无法加载模型

—— 检查模型文件名称是否完整，如果PC的模型文件路径包含中文，则adb push进去的文件夹名称会丢失，导致无法识别模型文件夹