

安装TensorFlow for Java

- 支持的平台
- 使用TensorFlow与Maven项目
 - 例
- 使用TensorFlow与JDK
 - 在Linux或Mac OS上安装
 - 在Windows上安装
 - 验证安装
 - 编译
 - 运行
 - 高级示例
- 从源代码构建

原文链接：https://www.tensorflow.org/install/install_java

译文链接：<http://www.apache.wiki/pages/viewpage.action?pageId=10029603>

贡献者：片刻 ApacheCN Apache中文网

TensorFlow提供用于Java程序的API。这些API特别适合加载使用Python创建的模型，并在Java应用程序中执行它们。本指南介绍如何安装 TensorFlow for Java 并将其用于Java应用程序。

警告：TensorFlow API 不在 TensorFlow API稳定性保证范围内。

支持的平台

以下操作系统支持TensorFlow for Java：

- Linux
- Mac OS X
- Windows
- Android

Android的安装说明在一个单独的 [Android TensorFlow支持页面](#)。安装完成后，请参阅 [Android上的TensorFlow的完整示例](#)。

使用TensorFlow与Maven项目

如果您的项目使用Apache Maven，则将以下内容添加到项目中pom.xml以使用TensorFlow Java API：

```
<dependency>
  <groupId>org.tensorflow</groupId>
  <artifactId>tensorflow</artifactId>
  <version>1.1.0</version>
</dependency>
```

例

例如，这些步骤将创建一个使用TensorFlow的Maven项目：

1. 创建项目的pom.xml：

```
<project>
  <modelVersion>4.0.0</modelVersion>
  <groupId>org.myorg</groupId>
  <artifactId>label-image</artifactId>
  <version>1.0-SNAPSHOT</version>
  <properties>
    <exec.mainClass>HelloTF</exec.mainClass>
    <!-- The sample code requires at least JDK 1.7. -->
    <!-- The maven compiler plugin defaults to a lower version
-->
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
  </properties>
  <dependencies>
    <dependency>
      <groupId>org.tensorflow</groupId>
      <artifactId>tensorflow</artifactId>
      <version>1.1.0</version>
    </dependency>
  </dependencies>
</project>
```

2. 创建源文件 (src/main/java/HelloTF.java) :

```

import org.tensorflow.Graph;
import org.tensorflow.Session;
import org.tensorflow.Tensor;
import org.tensorflow.TensorFlow;

public class HelloTF {
    public static void main(String[] args) throws Exception {
        try (Graph g = new Graph()) {
            final String value = "Hello from " + TensorFlow.version();

            // Construct the computation graph with a single operation, a
            constant
            // named "MyConst" with a value "value".
            try (Tensor t = Tensor.create(value.getBytes("UTF-8"))) {
                // The Java API doesn't yet include convenience functions
                for adding operations.
                g.opBuilder("Const", "MyConst").setAttr("dtype",
                t.dataType()).setAttr("value", t).build();
            }

            // Execute the "MyConst" operation in a Session.
            try (Session s = new Session(g);
                Tensor output =
                s.runner().fetch("MyConst").run().get(0)) {
                System.out.println(new String(output.bytesValue(),
                "UTF-8"));
            }
        }
    }
}

```

3. 编译执行：

```

-qmvn
mvn -q compile execjava

```

前面的命令应该从版本输出Hello。如果这样做，您已经成功地设置了TensorFlow for Java，并准备在Maven项目中使用它。如果没有，请检查 [堆栈溢出](#) 可能的解决方案。您可以跳过阅读本文档的其余部分。

使用TensorFlow与JDK

本节介绍如何使用TensorFlow `java`和`javac` JDK安装中的命令。如果您的项目使用Apache Maven，那么请参考上面更简单的说明。

在Linux或Mac OS上安装

采取以下步骤在Linux或Mac OS上安装TensorFlow for Java：

1. 下载 [libtensorflow.jar](#)，它是TensorFlow Java Archive (JAR)。
2. 决定是否在CPU上运行TensorFlow for Java或者在GPU的帮助下运行。为了帮助您决定，请阅读以下指南之一标题为“确定要安装的TensorFlow”的部分。
 - [在Linux上安装TensorFlow](#)

- [在Mac OS上安装TensorFlow](#)

3. 通过运行以下shell命令，为您的操作系统和处理器支持下载并提取相应的Java Native Interface (JNI) 文件：

```
TF_TYPE="cpu" # Default processor is CPU. If you want GPU, set to
"gpu"
OS=$(uname -s | tr '[:upper:]' '[:lower:]')
mkdir -p ./jni
curl -L \

"https://storage.googleapis.com/tensorflow/libtensorflow/libtensorflow
low_jni-${TF_TYPE}-${OS}-x86_64-1.1.0.tar.gz" |
tar -xz -C ./jni
```

在Windows上安装

请执行以下步骤在Windows上安装TensorFlow for Java：

1. 下载 [libtensorflow.jar](#)，它是TensorFlow Java Archive (JAR)。
2. 在Windows上下载适合于TensorFlow for Java的以下Java Native Interface (JNI) 文件。
3. 解压缩此.zip文件。

验证安装

安装TensorFlow for Java后，通过在下列代码中输入以下代码来验证安装HelloTF.java：

```

import org.tensorflow.Graph;
import org.tensorflow.Session;
import org.tensorflow.Tensor;
import org.tensorflow.TensorFlow;

public class HelloTF {
    public static void main(String[] args) throws Exception {
        try (Graph g = new Graph()) {
            final String value = "Hello from " + TensorFlow.version();

            // Construct the computation graph with a single operation, a
            constant
            // named "MyConst" with a value "value".
            try (Tensor t = Tensor.create(value.getBytes("UTF-8"))) {
                // The Java API doesn't yet include convenience functions for
                adding operations.
                g.opBuilder("Const", "MyConst").setAttr("dtype",
t.dataType()).setAttr("value", t).build();
            }

            // Execute the "MyConst" operation in a Session.
            try (Session s = new Session(g);
                Tensor output = s.runner().fetch("MyConst").run().get(0)) {
                System.out.println(new String(output.bytesValue(), "UTF-8"));
            }
        }
    }
}

```

HelloTF.java

编译

当编译使用TensorFlow的Java程序时，下载的程序.jar 必须是您的一部分classpath。例如，您可以通过使用编译标志将下载.jar到您classpath的内容中-cp，如下所示：

```
javac -cp libtensorflow-1.1.0.jar HelloTF.java
```

运行

要执行依赖于TensorFlow的Java程序，请确保以下两个文件可用于JVM：

- 下载的.jar文件
- 提取的JNI库

例如，以下命令行执行HelloTF程序：

```
java -cp libtensorflow-1.1.0.jar:.. -Djava.library.path=./jni HelloTF
```

HelloTensorFlow for JavaAPI

高级示例

有关更复杂的示例，请参阅 [LabelImage.java](#)，它识别图像中的对象。

从源代码构建

TensorFlow是开源的。您可以通过[单独的文档中的说明](#)从TensorFlow源代码构建用于Java的TensorFlow。