文章
Nicky Zhu · 四月 27, 2021 阅读大约需 6 分钟

Open Exchange

在IRIS中联合运用OCR与NLP技术

根据IDC的报道，超过80%的信息是基于NoSQL的，尤其是文本文件。当数字服务或应用程序不能处理所有这些信息时，企业就会遭受损失。为了面对这一挑战，可以使用OCR技术。OCR使用机器学习和/或训练的图像模式将图像像素转化为文本。这一点很重要，因为许多文件被扫描成PDF格式的图像，或者许多文件中包含有文本的图像。因此，OCR是一个重要的步骤，可以从文件中获得所有可能的数据。

为了实现OCR，可以使用开源解决方案Google Tesseract，这是Python和Java社区中最流行的解决方案。Tesseract支持100多个习语，并且可以用新的模型进行训练以识别车牌、验证码等等。Tesseract是在C++中创建的，可以通过Java套用Tess4J构成一个中介层来使用它。下面的代码展示了调用过程。

```
private String extractTextFromImage(File tempFile) throws TesseractException {
        ITesseract tesseract = new Tesseract();
       tesseract.setDatapath("/usr/share/tessdata/"); //directory to trained models
       tesseract.setLanguage("eng+por"); // choose your language/trained model
        return tesseract.doOCR(tempFile); //call tesseract function doOCR()
                                  //passing the file to be processed with OCR technique
    }
```

为了让IRIS使用这个Java类并从Java获得结果，我们需要使用PEX和Java网关解决方案。

首先，有必要在Production中配置Java代理，其次，配置一个PEX业务操作或服务来在Production中连接沟通IRIS和Java。

```
Class dc.ocr.OcrProduction Extends Ens.Production
{
 XData ProductionDefinition
{
<Production Name="dc.ocr.OcrProduction" LogGeneralTraceEvents="false">
 <Description></Description>
 <ActorPoolSize>2</ActorPoolSize>
 <Item Name="OcrService" Category="" ClassName=
"dc.ocr.OcrService" PoolSize="1" Enabled="true"
Foreground="false" Comment="" LogTraceEvents="false" Schedule="">
 </Item>
 <Item Name="JavaGateway" Category="" ClassName=
"EnsLib.JavaGateway.Service" PoolSize="1"
```

```
Enabled="true" Foreground="false" Comment="" LogTraceEvents="false"
Schedule="">
    <Setting Target="Host" Name="ClassPath">
.:/usr/irissys/dev/java/lib/JDK18/*:/opt/irisapp/*
:/usr/irissys/dev/java/lib/gson/*
:/usr/irissys/dev/java/lib/jackson/*:/jgw/ocr-pex-1.0.0.jar
</Setting>
    <Setting Target="Host" Name="JavaHome">
/usr/lib/jvm/java-8-openjdk-amd64/</Setting>
  </Item>
  <Item Name="OcrOperation" Category="" ClassName=
"EnsLib.PEX.BusinessOperation" PoolSize="1"
Enabled="true" Foreground="false" Comment="" LogTraceEvents="false"
Schedule="">
    <Setting Target="Host" Name="%gatewayPort">55555</Setting>
    <Setting Target="Host" Name="%remoteClassname">
community.intersystems.pex.ocr.OcrOperation</Setting>
    <Setting Target="Host" Name=
"%gatewayExtraClasspaths">.:/usr/irissys/dev/java/lib/JDK18/*
:/opt/irisapp/*:/usr/irissys/dev/java/lib/gson/*
:/usr/irissys/dev/java/lib/jackson/*
:/jgw/ocr-pex-1.0.0.jar
</Setting>
  </Item>
</Production>
}
  }
```

现在，任何IRIS Production都可以与Java和Tesseract进行通信了! 如下：

```
 //call ocr method to get text from image, if you want to use pex
        Set pRequest = ##class(dc.ocr.OcrRequest).%New()
        Set pRequest.FileName = file.Filename

// call java pex operation to do ocr, passing file into pRequest and receive ocr text
with pResponse
        Set tSC = ..SendRequestSync("OcrOperation", pRequest, .
```

```
pResponse, 1200)
        //save the results into database to use text analytics - nlp
        Set ocrTable = ##class(dc.ocr.OcrTable).%New()
        Set ocrTable.FileName = file.Filename
        Set ocrTable.OcrText = pResponse.StringValue
   Set tSC = ocrTable.%Save()
```

所有的代码细节，连同注释都可以在我的OCR服务库（https://openexchange.intersystems.com/package/OCR-Service）中找到。

现在，随着文本的提取，我们需要使用IRIS NLP引擎来分析文本数据，并获得支持决策的见解。为此，当文本被提取后，它被保存到一个表中，这个表被NLP引擎用作文本源。请看上面的表%Save()，请看下面的代码，NLP引用OCRTable（有文本提取的地方）。如下：

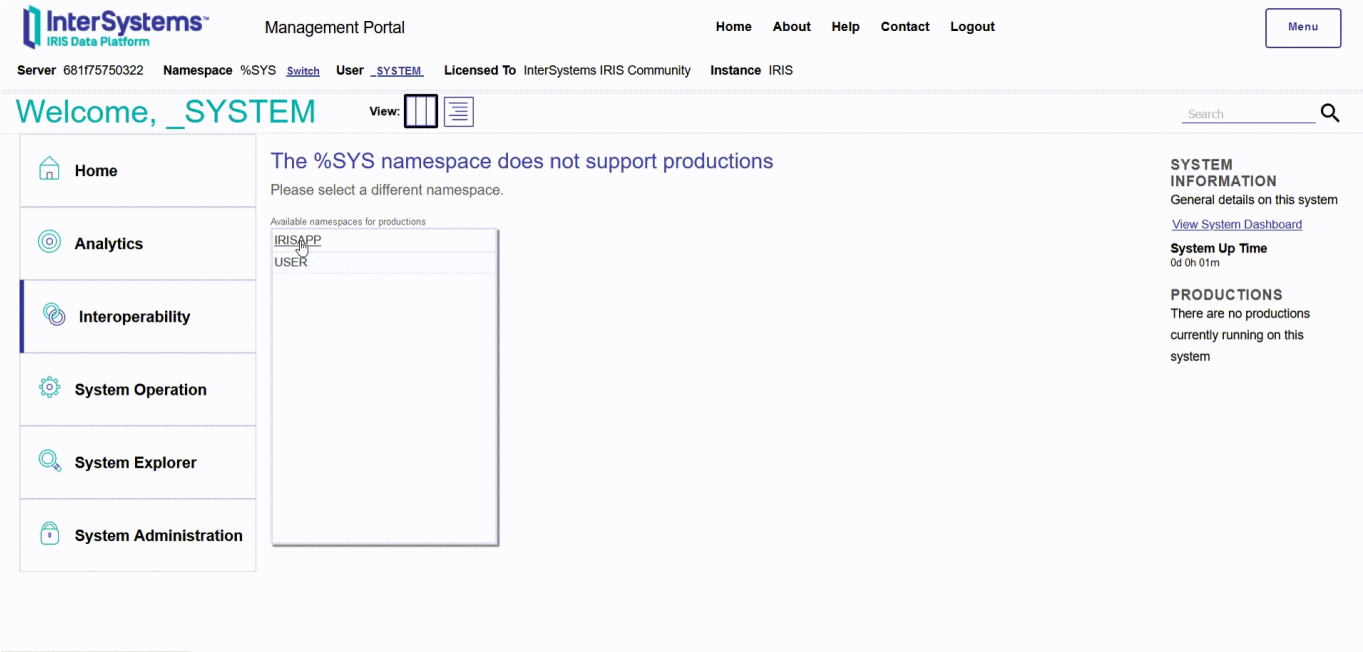
```
 Class dc.ocr.OcrNLP Extends %iKnow.DomainDefinition [ ProcedureBlock ]
{
  XData Domain [ XMLNamespace = "http://www.intersystems.com/iknow" ]
{
<domain name="OcrNLP" disabled="false" allowCustomUpdates="true">
<parameter name="DefaultConfig" value="OcrNLP.Configuration" isList="false" />
<data dropBeforeBuild="true">
<table listname="OcrNLPTable" batchMode="true" disabled="false"
listerClass="%iKnow.Source.SQL.Lister" tableName=
"dcocr.OcrTable"   idField="ID"
groupField="ID" dataFields="OcrText" metadataColumns="FileName"
metadataFields="filename" />
</data>
<matching disabled="false" dropBeforeBuild="true" autoExecute="true"
ignoreDictionaryErrors="true" />
<metadata>
<field name="filename" operators="=" dataType="STRING" storage="0"
caseSensitive="false" disabled="false" />
</metadata>
<configuration name="OcrNLP.Configuration" detectLanguage="true"
languages="en,pt"
userDictionary="OcrNLP.Dictionary#1" summarize="true"
maxConceptLength="0" />
<userDictionary name="OcrNLP.Dictionary#1" />
```

```
</domain>
}
  }
```

在我的OCR服务github资源库中看到完整的细节和配置。

现在我们可以上传一些文件，到资源管理器中查看概念和生成的CRC。

请参阅动画与这里讨论的所有步骤。



欢迎尝试 OCR/NLP!

#Java #互操作性 #分析 #InterSystems IRIS
在 InterSystems Open Exchange 上检查相关应用程序

源URL:
https://cn.community.intersystems.com/post/%E5%9C%A8iris%E4%B8%AD%E8%81%94%E5%90%88%E8%BF%90%E7%94%A8ocr%E4%B8%8Enlp%E6%8A%80%E6%9C%AF