**第四课 课程总结和心得体会**

**刘超**

这周人工智能高阶人才培训的课程分为两部分，分别是微软的资深项目经理郑春蕾老师为我们讲解“AI道德与伦理”以及微软亚洲研究院高级研究开发工程师宋驰老师指导我们熟悉微软在认知计算方面的平台与工具。以下将分别介绍这两部分的课程总结和心得体会。

**一、AI伦理道德**

AI伦理道德课程主要分以下几个部分：什么是伦理道德，技术人员为什么要讲伦理道德，什么是AI伦理以及世界各国的应对。

**1、什么是伦理道德**

郑老师告诉我们，“伦理道德”总的来说就是风俗、习惯的意思。在西方世界，主要分为叔本华的“唯意志主义伦理”，詹姆斯的“实用主义伦理”和海德格尔的“存在主义伦理”。其中海德格尔的“存在主义伦理”对西方影响最大，他强调“自由是价值存在的唯一源泉”。他的理论在20世纪50年代非常流行，经常通过文学作品进行宣传，以至于影响了西方的一代人，英国五十年代“垮掉的一带”以及美国六十年代的“嬉皮士运动”都与之有关。

在我们中国，公元前六世纪《周易》、《尚书》已经提出“伦理”的概念。“伦”就是“三纲五常”，即“君为臣纲，父为子纲，夫为妻纲”和“仁、义、礼、智、信”。“理”就是要遵守的道理。在中国古代，伦理道德的流派是“儒家”、“墨家”和“法家”。墨家讲究“兼相爱”，法家倡导法制。孔子“仁”的思想对我们影响最大，孔子说“克己复礼”。孟子发展了孔子的理论，强调“教以人伦”和“仁之有道”，强调教育的作用。

总之，伦理是哲学的分支，可以建立人与人的关系，通过潜在的价值观对人产生制约。

**2、技术人员为什么要讲伦理道德**

郑老师通过“魏则西”事件形象生动地指出百度竞价排名、公立医院私自出售名头，用手机端悄悄推送给用户等恶劣作法，并指出这些做法造成了魏则西的死亡，社会对百度的负面印象加剧以及百度股价大跌等一系列负面结果，百度也认识到这些问题，并采取了一系列措施，例如搜索医院的时候不再以价格作为排序标准，不再在公立医院搜索的时候出现广告等。通过百度的事例，郑老师想传达给我们一个技术企业要负社会责任的观点。而与此形成鲜明对照的是，比尔盖茨积极从事慈善事业。郑老师在这里强调了“格局”的重要性，并指出一个企业“格局有多大舞台就有多大”。

**3、什么是AI伦理以及各国的应对**

郑老师以斯皮尔伯格2001年的电影《人工智能》中提出的问题作为开头，抛出“当人工智能有了情感，人类将如何对待”的伦理问题。她指出沈向洋等一系列专家已经意识到，目前AI时代最重要的问题就是伦理问题，因为科技已经走在伦理道德和法律的前面。人工智能的专家已经纷纷开始思考AI伦理道德问题及其应对。

接下来，郑老师为我们举例分析了AI伦理道德带来的9个问题，例如失业问题（在未来的20-30年，有50%的工作机会会被人工智能代替）、不平等问题（如何分配机器创造的财富）、算法歧视、安全、机器人权力、隐私等。郑老师讲的两个例子让我印象最深，第一个是责任和安全问题。“无人汽车内坐着两个乘客，车外有两个行人，当车辆不得不发生碰撞事故时，优先保护谁？”一个美国的调查机构，找了很多受访者做调查。结果是，大部分人会选择应该去保护车内的乘客。但随着车外被牺牲的行人的增加，大部分人开始转向保护车外的行人，当车外人数增加到10人时，几乎所有人都选择保护车外的人。而与此同时，受访者表示坚决不会购买无人驾驶汽车，这体现了受访者在伦理问题中的两难困境。另外，在现实世界中，特斯拉无人汽车有一次因为行车环境阳光耀眼，没有识别到白色卡车，造成了车祸和伤亡。这到底是谁的责任？是设计系统的工程师还是车主，至今也没有答案。第二个例子是隐私的保护，facebook去年有5000万的用户数据被剑桥分析公司所泄露。而公民的隐私在西方是非常受到重视的，因此从来都是身穿黑色T恤的扎克伯格也很罕见地穿上西装出席听证会接受听证会问询。

在应对AI的伦理道德问题方面，2018年5月《通用数据保护条例》（General Data Protection Regulation，简称GDPR）正式在欧洲生效。该条例中提到了几点，一个是被遗忘权，还有一个限制使用权保护。该条例保护非常严格，罚款非常高，因此大公司有专门的团队不断审查公司的产品与服务是否触犯GDPR，而很多小企业干脆关闭了欧盟的业务。另外，微软在沈向洋的《计算未来》的书中提出了，“公平、可靠和安全、隐私和保障、包容、透明、负责”的六原则。2016年12月,IEEE首家发布了第一版《合乎伦理的设计:将人类福祉与人工智能和自主系统优先考虑的愿景》，2019年4月8日，欧盟发布《人工智能伦理准则》，微软也有线上课堂，讲述AI伦理道德的内容。

总之，郑老师教了我们什么是伦理道德，为什么技术人员要关心伦理道德，然后通过9个问题具体分析了AI伦理道德的内涵，最后讲解了世界各国对AI伦理进步所做的努力。最后，老师用“世界很复杂，永远别低头”与我们共勉。感谢郑老师让我们系统性地了解了AI的伦理道德问题，我们在今后的项目开展和产品设计中会在这方面多加留意，“以人为本”地发挥科技力量。

**二、微软认知服务的使用**

课程的下半段，由宋驰老师为我们讲解微软认知服务，并带着我们实践体验。宋驰老师认为，世界就是一台计算机，从物理构成上它是一体的，但从数据角度来看它是分散的，如何让这台世界计算机有智能就是要解决的问题。大部分认知服务就解决一件事，就是“**把人类能识别的非结构化数据转变为结构化的数据**”，例如目标检测，OCR（文字识别），计算机视觉，人脸识别，情感分析，语言认知，语音生成，拼写检查，文本翻译，个性化推荐，异常检测（工业上的预测性维护）。我们这次主要学习两方面的认知计算服务，即“看图识熊”（图片分类）和“智能家居”（语音识别和语义理解）。下面将分章节进行回顾。

**1、看图识熊**

这次课程使用的微软认知计算平台是“定制化视觉”（https://customvision.ai）。该平台是微软为了满足企业定制化API需求而研发的图形化的定制化模型服务和模型下载平台。整个定制开发流程也非常简单，只需上传一些数据进行训练，然后通过REST端点来调用模型，最后再将调整好的模型应用到实际程序中即可。以创建一个对黑熊、棕熊图片进行分类的程序为例，步骤如下：

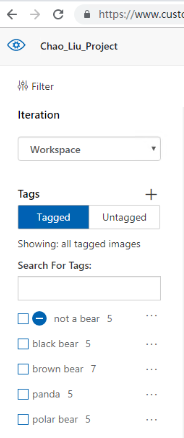
1）新建project

用课上提供的用户名和密码登录<https://customvision.ai>，单击“NEW PROJECT”。

在”Project Type”处，选择”Classification”，然后取一个名字，比如“Chao\_Liu\_Project”，点击”OK”

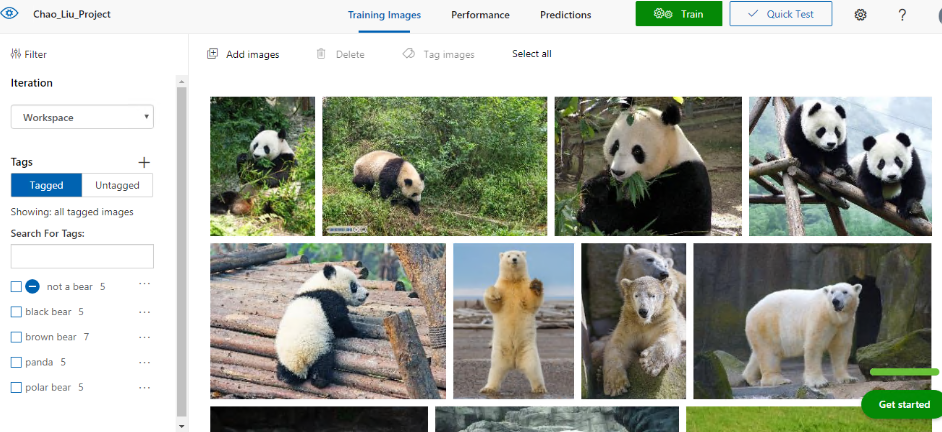
2）创建标签

点击”Tags“右边的”+”号，创建标签，至少要创建两个标签才能分类。这里我们创建“not a bear”（负例），”black bear”， “brown bear”, “polar bear”这四类。



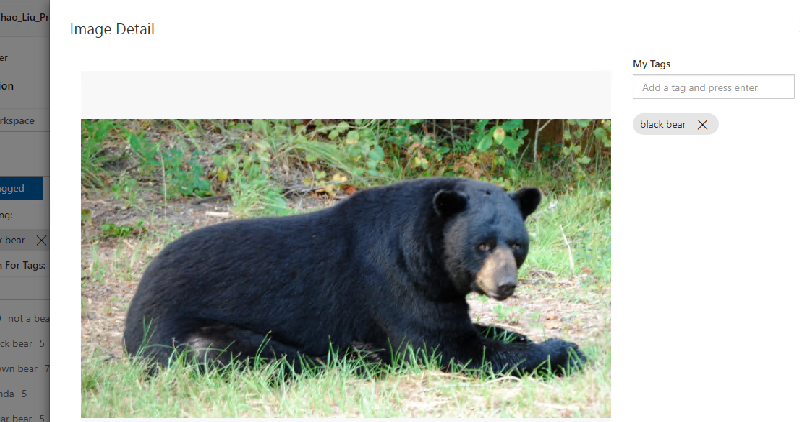
3）上传图片（数据）

点击”Add Images”上传图片，这里我们使用老师已经为我们事先准备好的图片数据。



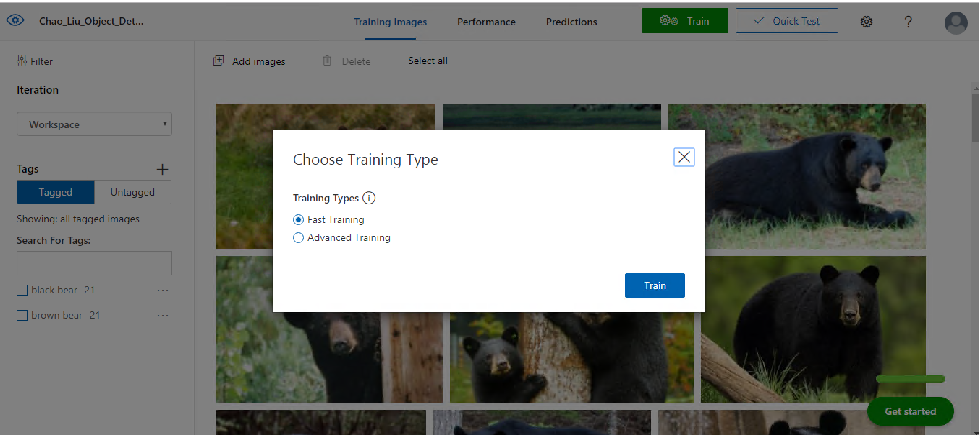
4）标记数据

对于神经网络来说，**数据决定了算法最后的质量。**因此选择图片和标记图片非常重要。真正做应用的时候，要考虑如何把那些我们认不好的图片手动标记后重新训练，提高我们的模型精度。这里我们可以点击每张图片，选择它是属于哪个类别的。

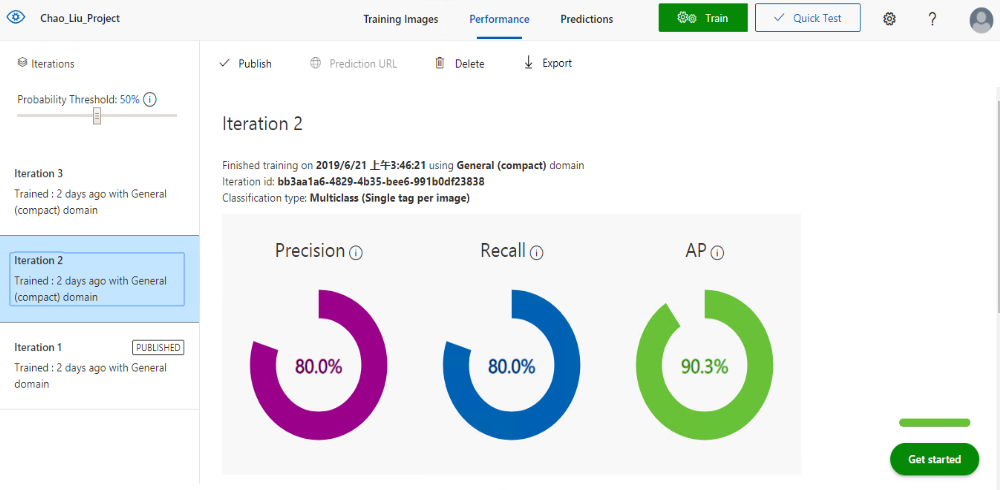


5）训练

训练步骤非常简单，只要点击” Train “按钮，然后选择训练类型。



训练类型分为Fast Training，即通过少量数据和迁移学习进行训练；另一种是Advanced Training，即从头开始训练，这会消耗大量的算力，也需要有非常多的数据。我们这里采用Fast Training. 训练完后得到的模型性能如下。



要看懂该指标，需要了解以下概念：

* 真正例（TP）：实际上是正例的数据点被识别为正例
* 假正例（FP）：实际上是反例的数据点被识别为正例
* 真反例（TN）：实际上是反例的数据点被识别为反例
* 假反例（FN）：实际上是正例的数据点被识别为反例

召回率(Recall)和精度(Precision)衡量指标：

* 召回率（R）：分类模型识别所有相关实例的能力
* 精度（P）：分类模型仅仅返回相关实例的能力

**召回率=真正例/（真正例+假反例）。**“**宁可错杀一千不可放过一个**”，这个要的就是召回率。也就是正识别为反的概率最小化。

**精度=真正例/（真正例+假正例）。**又叫查准率，是指在所有被预测为正的样本中实际为正的样本的概率。有些分类算法造成某些人是“混进革命的队伍”。老师说，现在判刑的无罪推定，不要造成冤案，这个就是要提高精度。

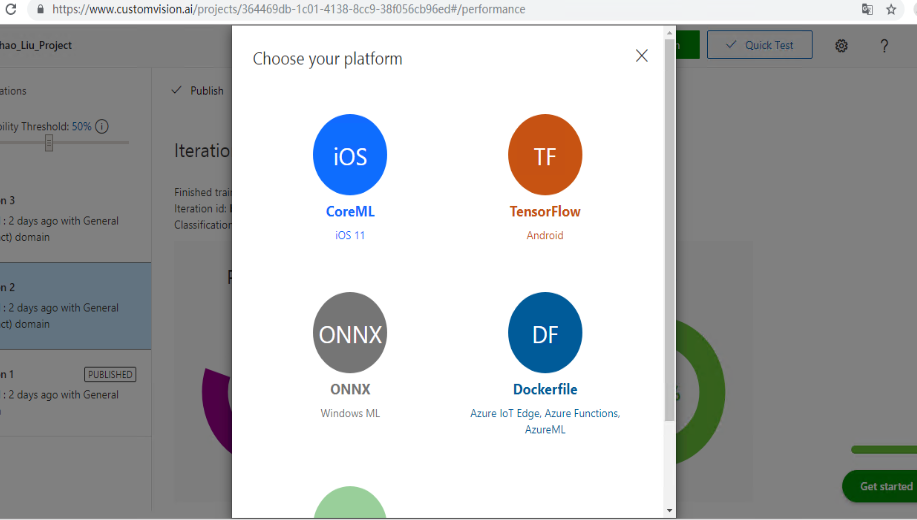
**一般模型通过不断地迭代训练，这两个参数都能提高到90%，但当要再往上提高的时候，往往需要下降某一个才能达到目标**。

6）预测

发布后，就可以看到一个predict URL，你如果不想下载本地就可以在直接在线用了。也可以点Quick Test做在线测试。

7）导出模型

导出的模型必须要是压缩(general compact)的，如果更改了模型必须要重新训练一下。然后在Performance的地方就可以点”export”了，会弹出如下图。

平台选择”ONNX”，ONNX的全称是Open Neural Network Exchange（ONNX，开放神经网络交换）格式，是一个用于表示深度学习模型的标准，可使模型在不同框架之间进行转移。我们会在后面的Windows Mache Learning库中使用它。

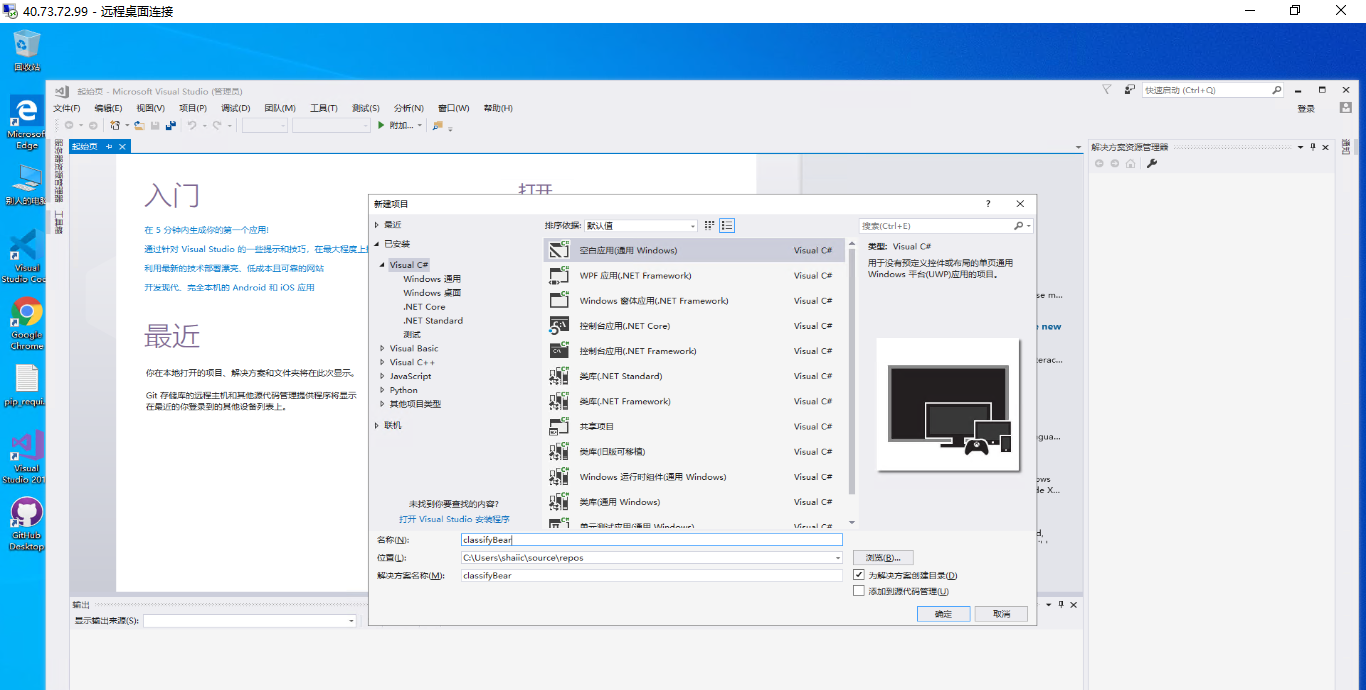
8）使用Windows Machine Learning库进行熊的识别

接下来，用下载的ONNX模型在本地做熊图片的分类识别。

8.1） 创建项目

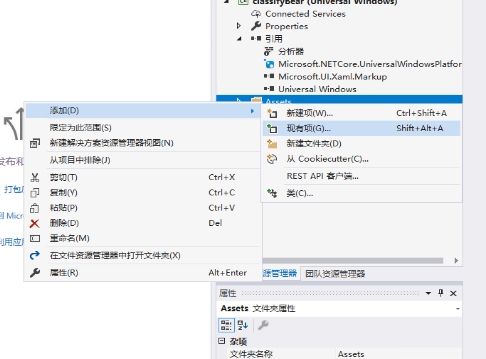
Windows Machine Learning库需要在Windows 10上运行Visual Studio 2017。并要安装相关组件才能使用。课程事先已经帮学员们每人准备了一个开发环境，每位学员只需要使用远程桌面（MSTSC）即可连接Azure中国上的Visual Stodio 2017环境。

进入环境后，我们先创建一个项目，选择“空白应用（通用Windows）”，项目名为“classifyBear”。环境选择最新的“17763”，如下图所示。

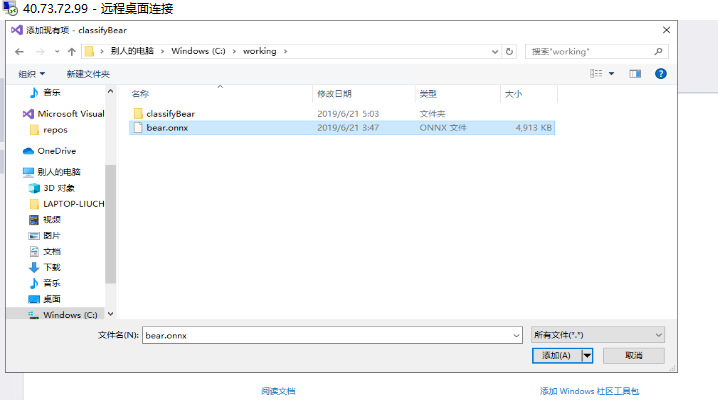


8.2）添加模型

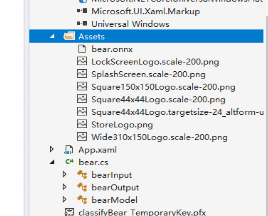
在“通用解决方案资源管理器”中，右键点击Assets选择添加🡪现有项，



选模型（以ONNIX为扩展名的文件）



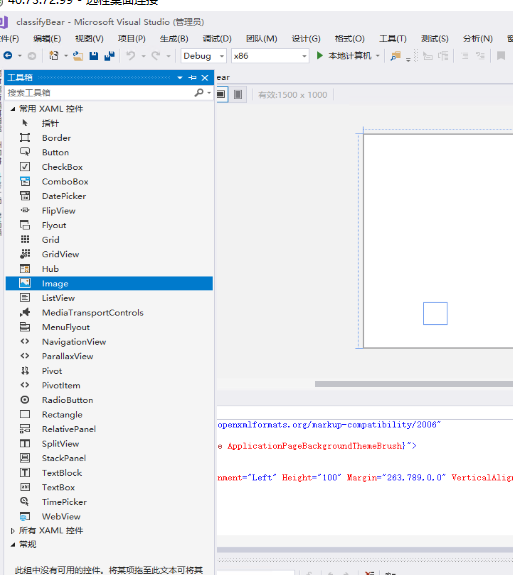
选完模型后就能自动生成了bear.cs，如下图所示。



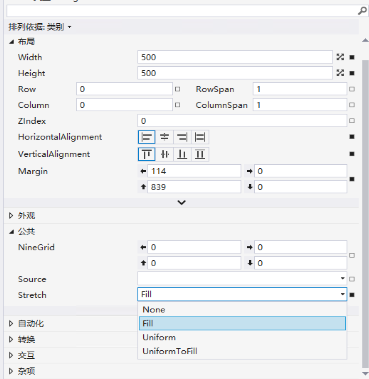
bear.cs文件包含三个类，输入，输出和模型类。用的时候，模型类是通过它来初始化模型，在初始化的时候，需要从一个ONNIX文件打开这个模型，这都是C#的异步操作，效率很高。还有一个去评估一个类型，就是根据输入bear input输出是bear output。

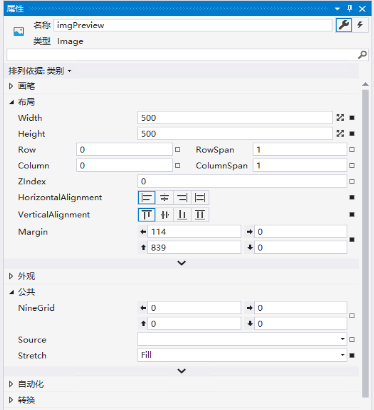
8.3）设计界面

接下来我们创建一个界面。生成的空白项目有MainPage.xmal，双击这个文件就会有一个“界面设计器”，左边有一个工具箱。



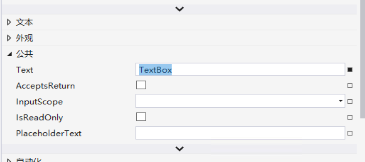
将Image的属性变成正方形，因为训练时候的图像尺寸和预测是一样的，但实际的图像尺寸都不一样，那么为了我们把信息都保留，一般采用拉伸而不是切的方法。另外把图像的Strethch设置为Fill，就是把图全部填满空间。



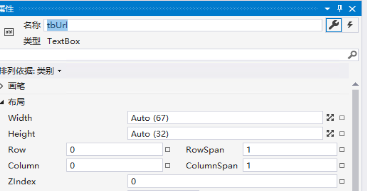


接下来把Image改名字为imagePreview。这样图片的控件就做好了。

我们再加一个textbox.



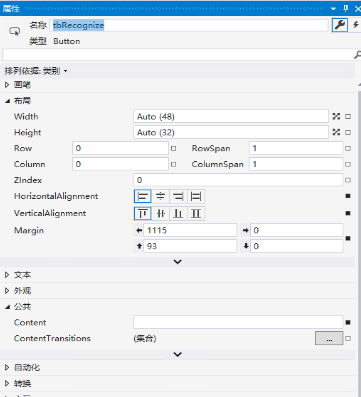
把原先的内容清空。名字改为tbUrl.



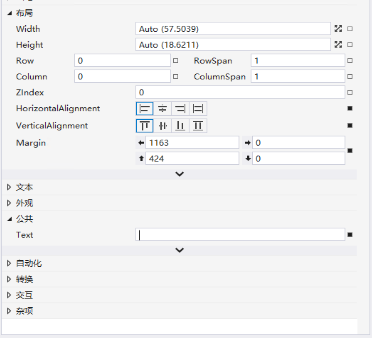
再拉一个button按钮，在Content中叫“识别”

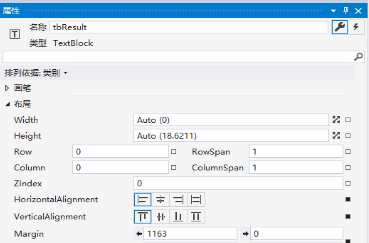


名字叫tbRecognize.

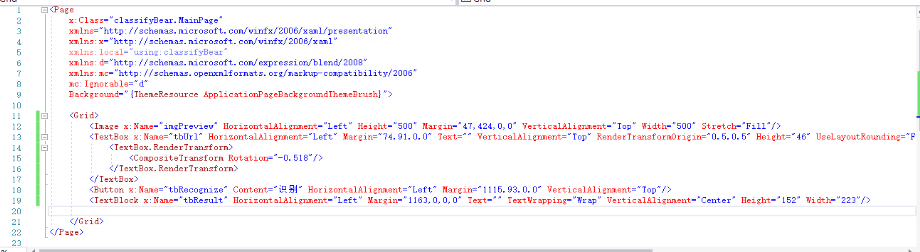


再添加textblock，叫tbResult



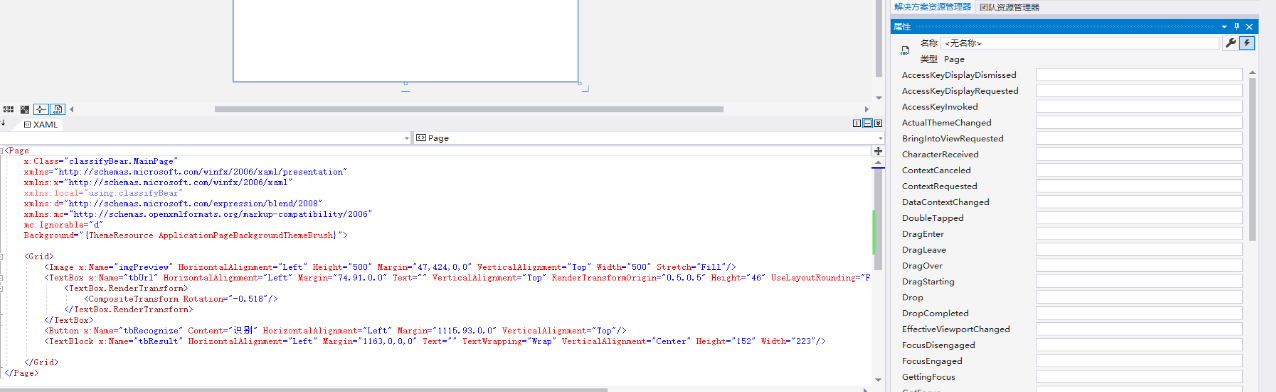


这些图形化操作也会自动生成代码，如下。

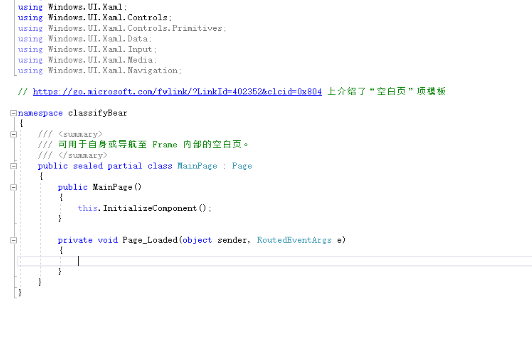


8.4）Loaded事件的回调函数

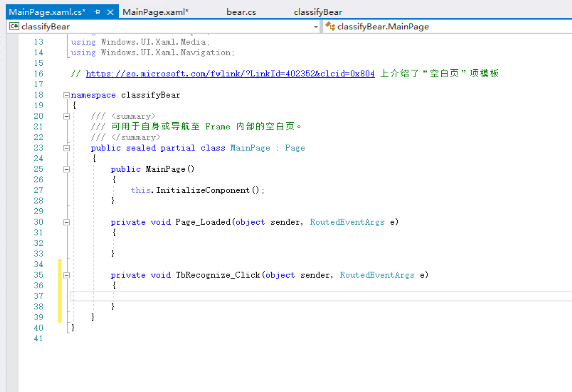
选择Page，点右上角闪电，找到Loaded事件。



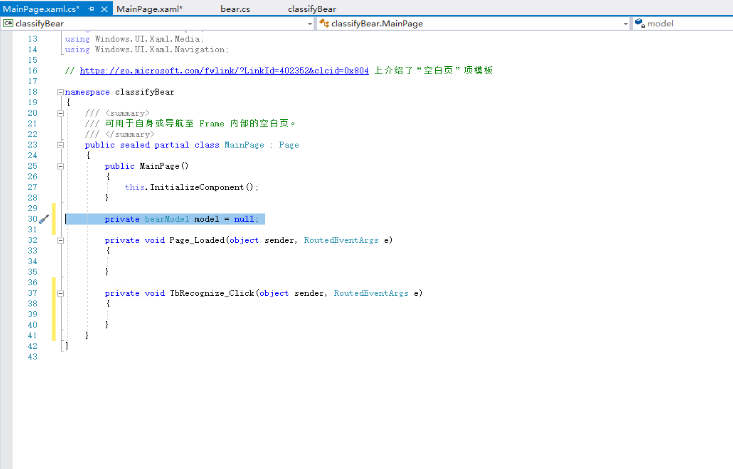
双击Loaded，自动生成函数框架。



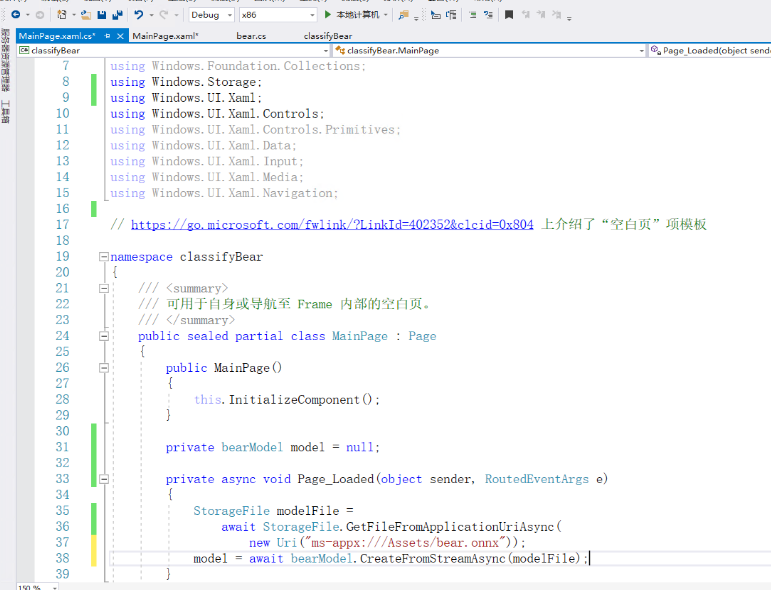
再回到页面，双击button，增加了事件响应函数。我们这个程序只需要这两个事件响应函数即可。



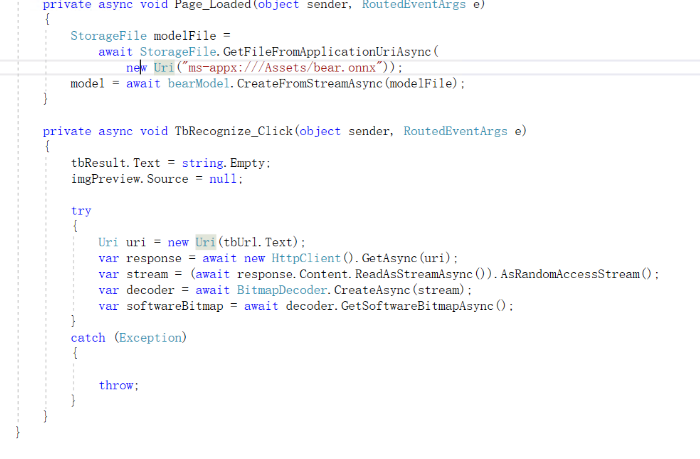
第一步，在MainPage.xaml.cs中，加一行model的变量定义。



在Page\_Loaded中，输入以下代码：



Page\_Loaded函数的作用是在程序界面载入的时候，载入我们训练的模型。值得注意的是其中用到了await，await是把异步资源变得更高效的方法。当执行GetFileFromApplicationUriAsync的时候，一遇到await命令程序就不会等资源，就直接往下走，这种程序一般是多线程，线程就去执行别的了，等资源准备好后，这个函数返回，就进入队列去排，排到之后，又从这里执行下一步。Await是把异步操作更高效的方法。异步操作要高效最简单就是开很多个线程，大家做同一件事情，大家竞争，但多线程也是占有资源的，更好的方法是，大家等的都是CPU，CPU走到这一句，进到I/O操作后，CPU就返回，这里I/O完成了之后，我就回调，回调也不是马上执行，等CPU的空闲了之后，从这里取任务执行。我们可以看到很多async（函数名上的async） awit的标志，这样性能比较高。await是一种更高效的回调。Async和await是配对的，它会等到await完成后在执行。



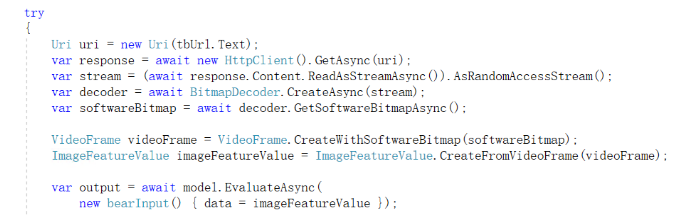
8.4）识别按钮的事件响应

接着我们处理TbRecognize\_Click这个回调函数。首先清空控件，接着处理输入、推理和输出。

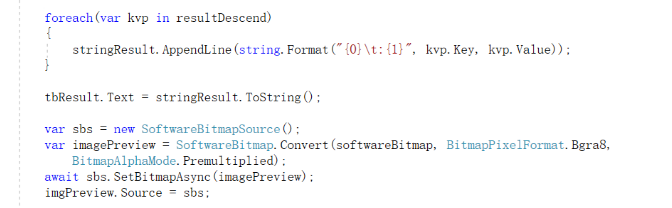
对于输入而言，图像对象有很多方法可以存，而机器学习需要点阵图，video frame是可以处理视频的，这里就需要转换各种stream。原先stream是不随机存储的，只有随机存储才能decoder。在键入代码时，如果发现有红色波浪线，可以采用Shift+Alt+F10让Visual Stidio自动补全包的引入。



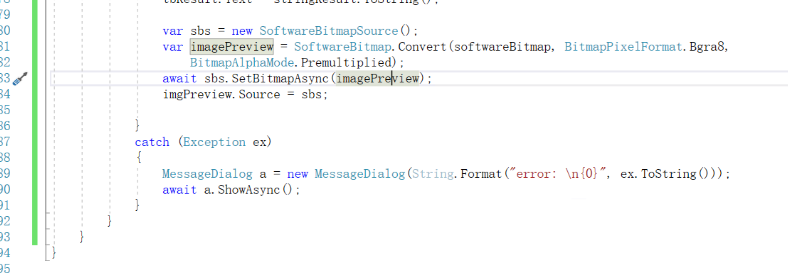
接着实现推理，直接调用model.EvaluateAsync然后把Input传入即可。



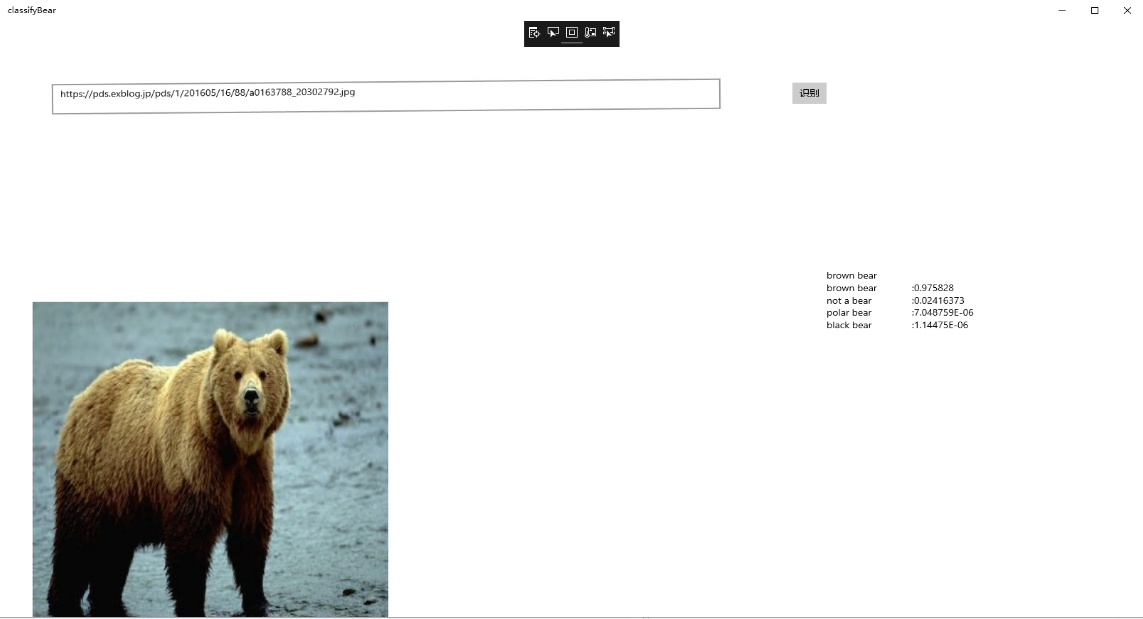
最后处理output，即美化输出，倒序输出，再做格式化美化



最后还有一个收尾的工作，如果输错，要有一个消息框弹出。

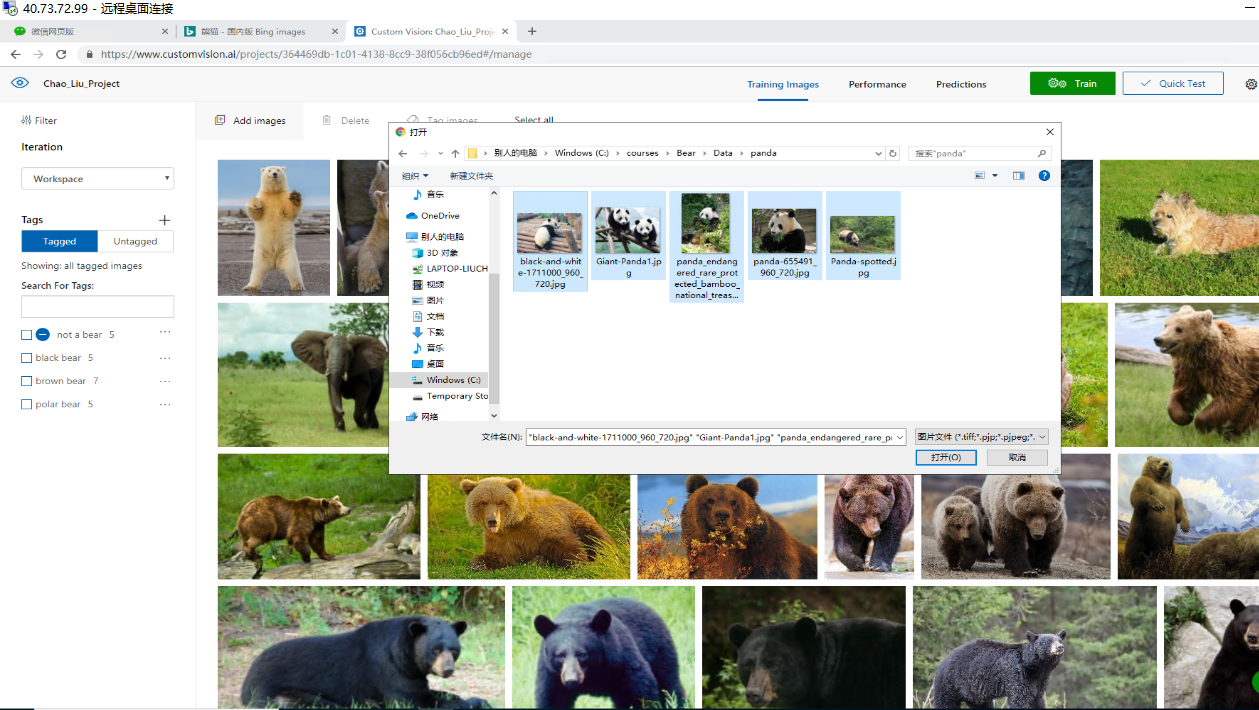


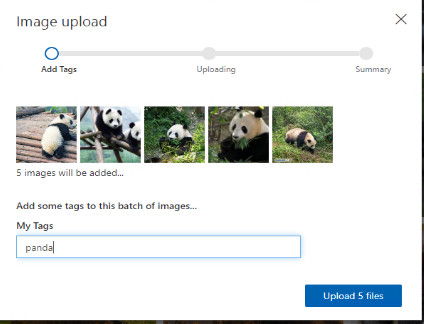
运行效果如下：

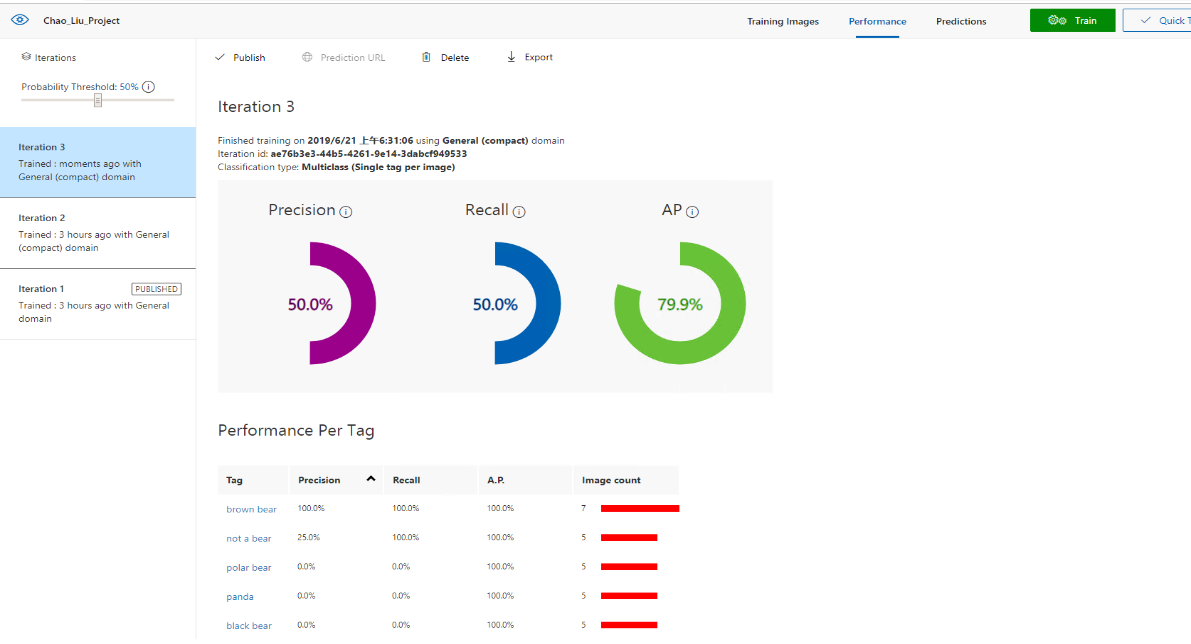


8.5）更新数据和模型，实现熊猫的识别

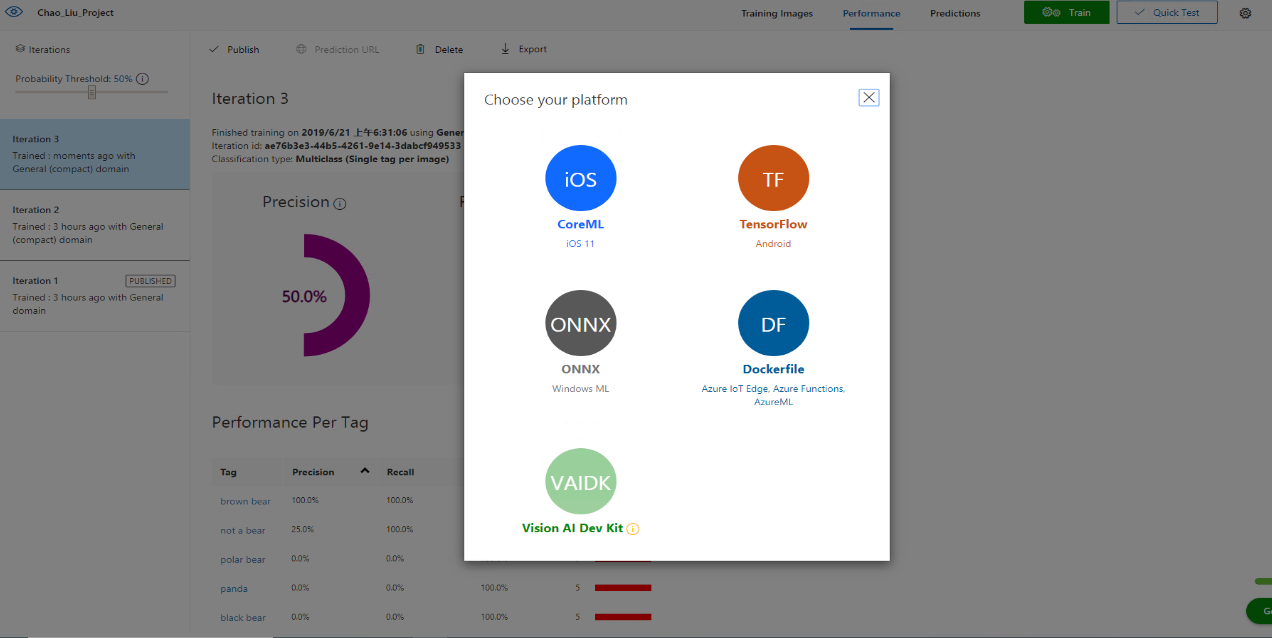
老师进一步指导我们怎么识别熊猫。通过上传熊猫照片，新建熊猫的标记，标记熊猫图片，迭代训练模型。



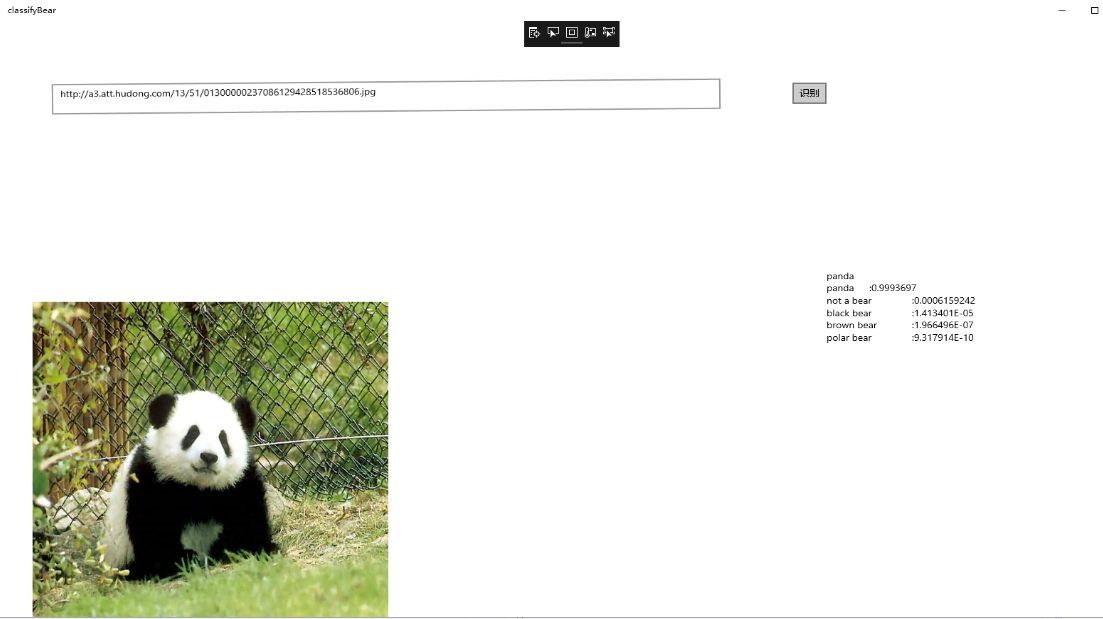




保存到代码的目录中，替换掉原来的模型

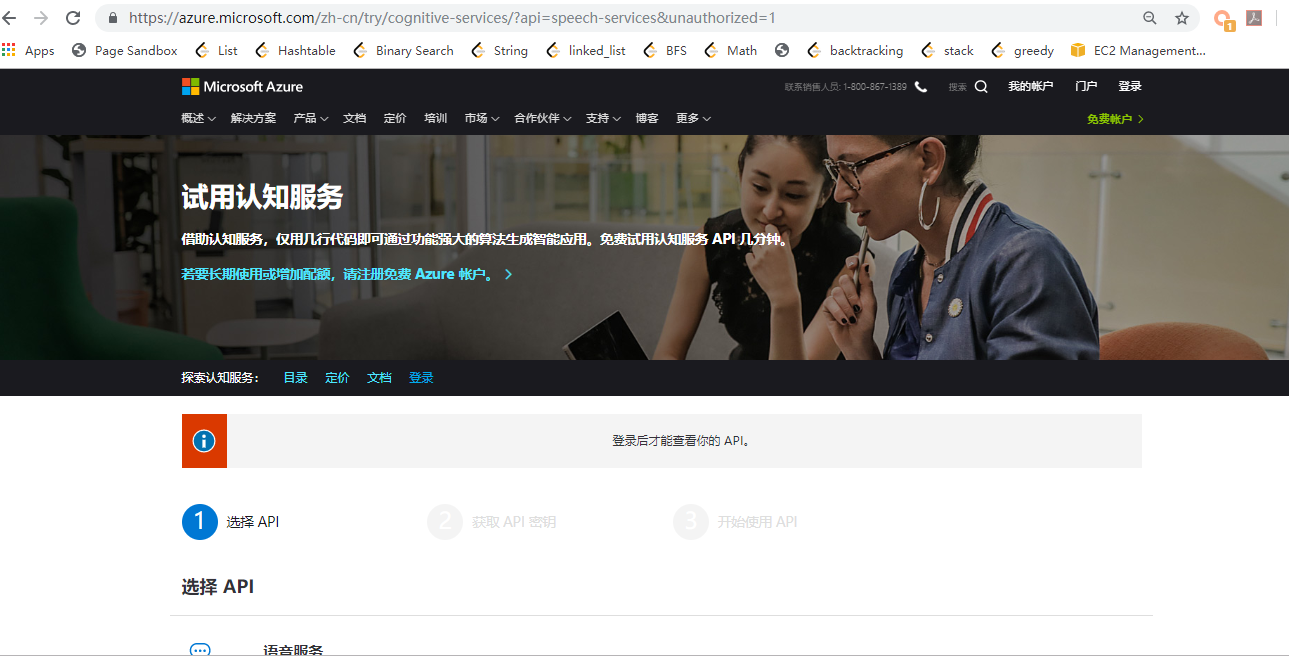


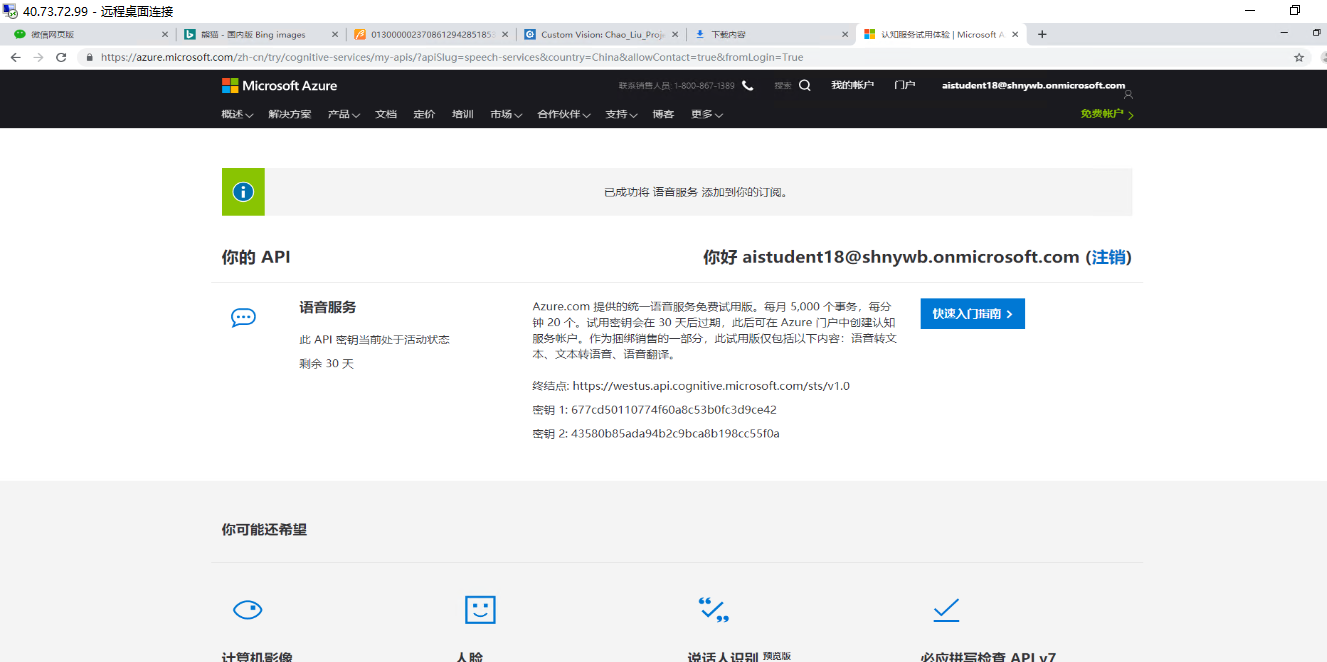
可以发现，这里真正改的是数据和模型，代码没有任何改动的，整个程序就具有识别熊猫的能力了。



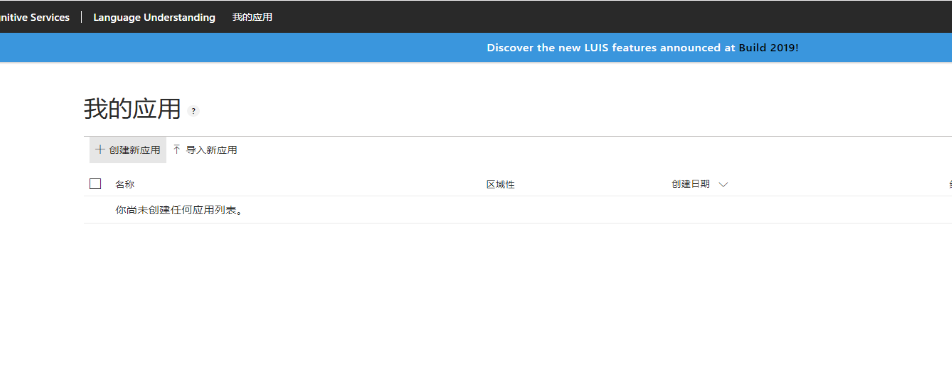
**2、语音服务**

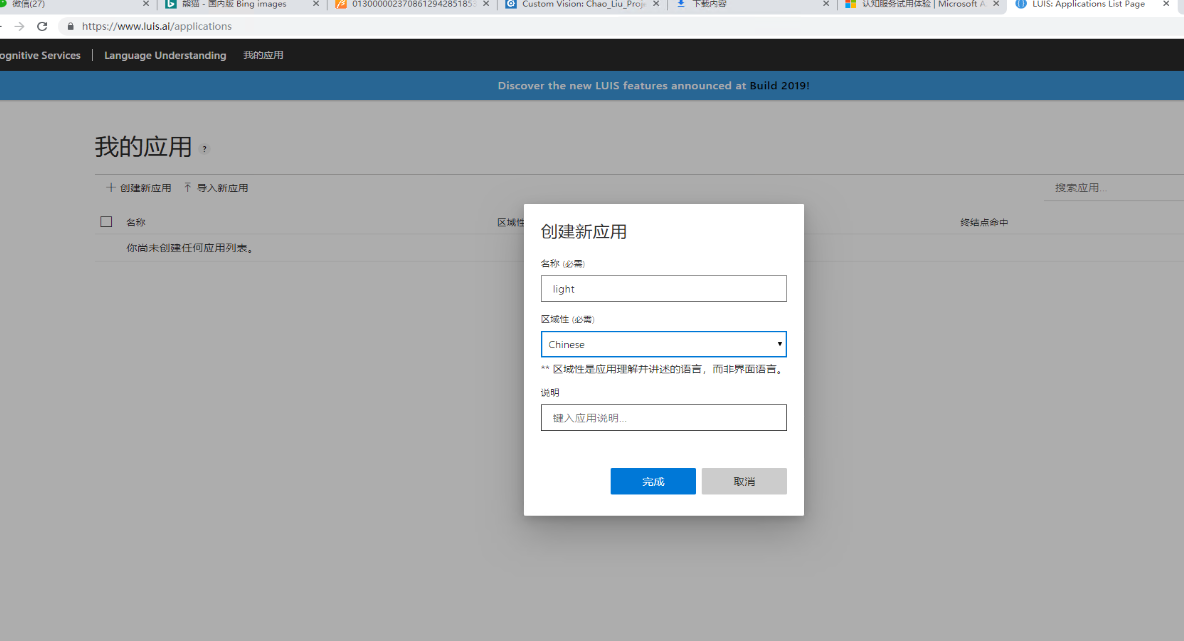
语音服务涉及到两个认知服务即“语音转文字”和“语言理解”。但由于是纯服务端的，对客户端的版本要求不高。这次的练习场景类似于通过“小爱音响”来开关电灯，国外类似有”IFTTT”的网站都是可以做到的。语音转文本，要做到非常自然的人机体验是很难的。例如说了多少时间算对方说完了，叫“静默时长”。另外，“定制化需求”，7里香还是七里香，一九八零还是1980，5点6、5.6还是05：06？另外一个服务叫“自然语言理解服务”，即把文字转换为结构体。开关灯的实体就是灯。意图就是开和关。语音识别可以登陆微软的认知服务网站：<https://aka.ms/trystt>，如下图。

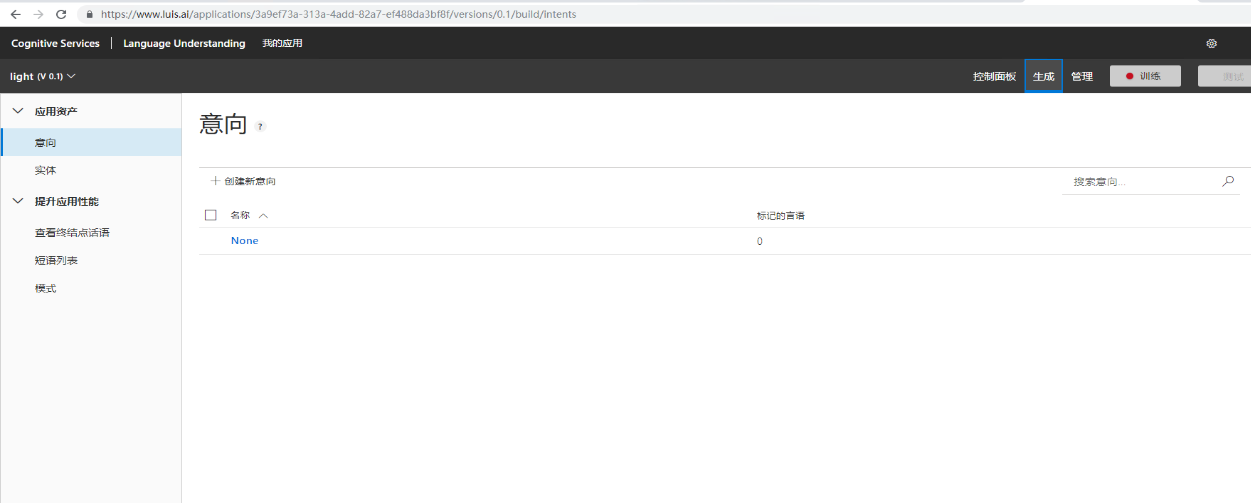




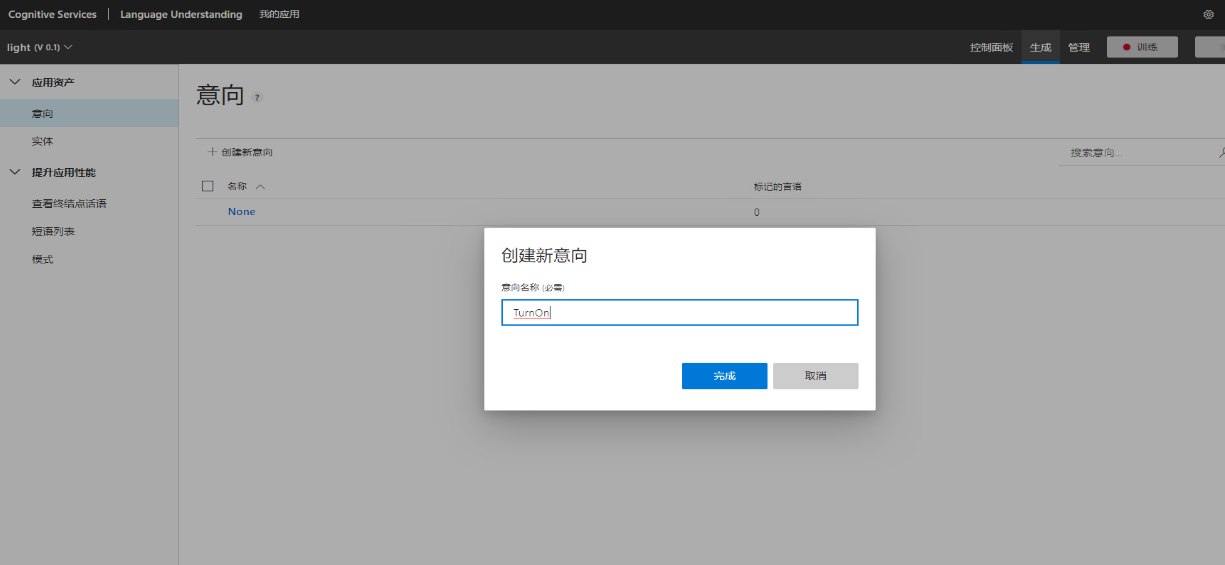
定制语言理解服务跟图像类似，见<https://www.luis.ai>



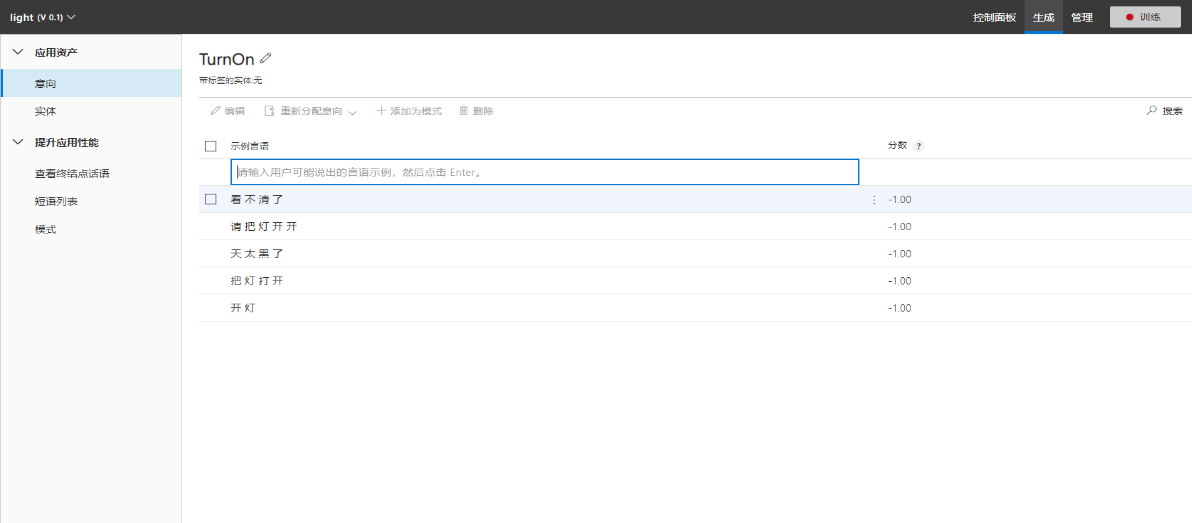




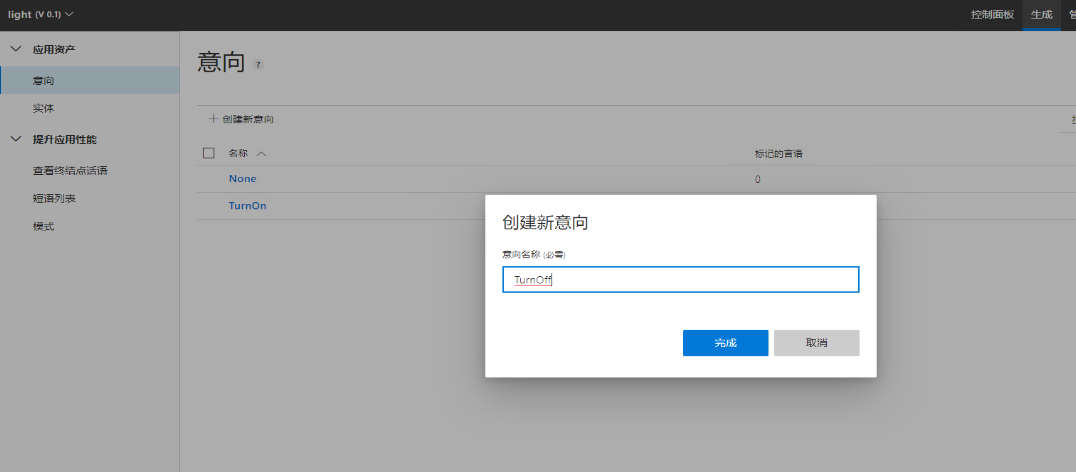
默认 None是负例，创建一个新的意向，例如”TurnOn”。

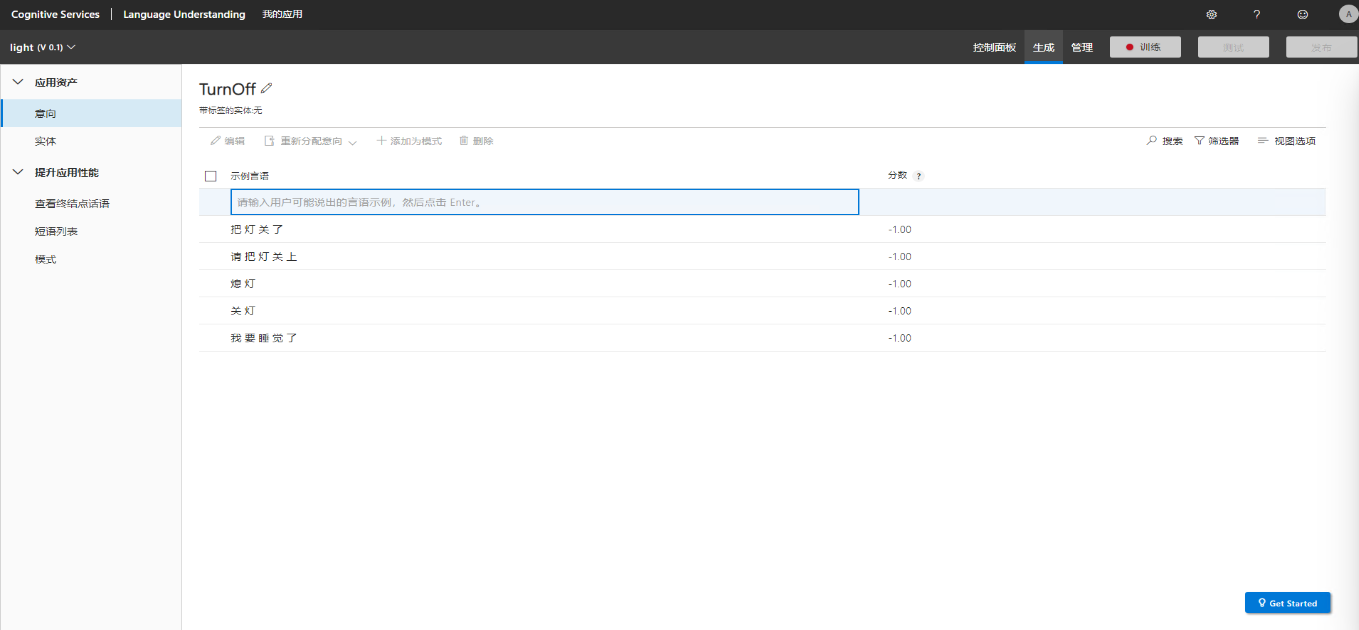


然后在TurnOn这个意图下输入所映射的语言，例如“看不清了”，“请把灯开开”等，如下。

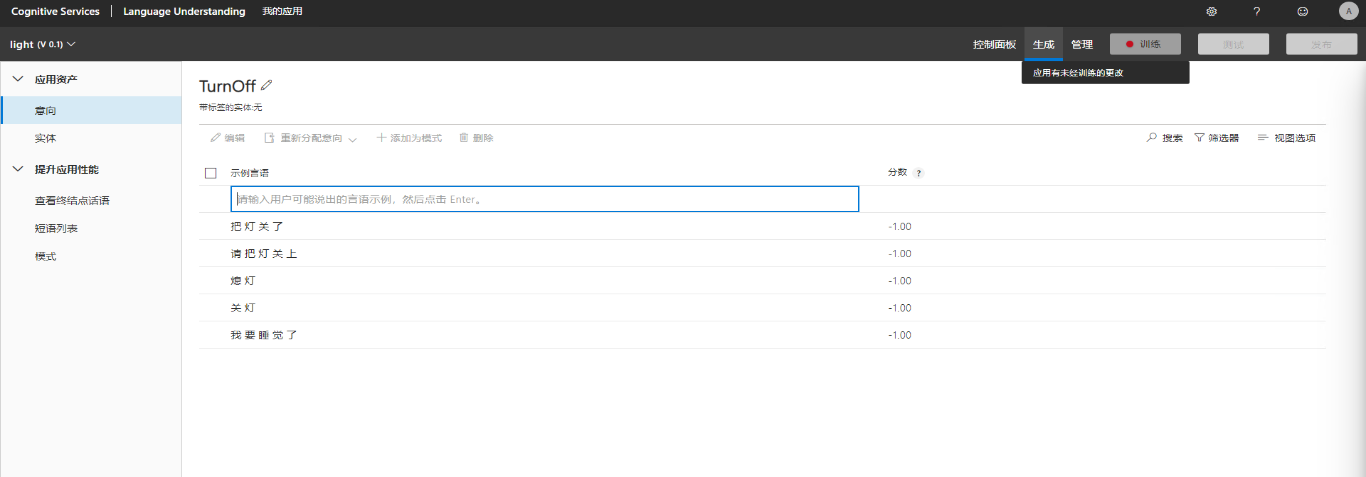


同理，创建一个“TurnOff”的意向，加入“把灯关了”，“请把灯关上”这些语句。

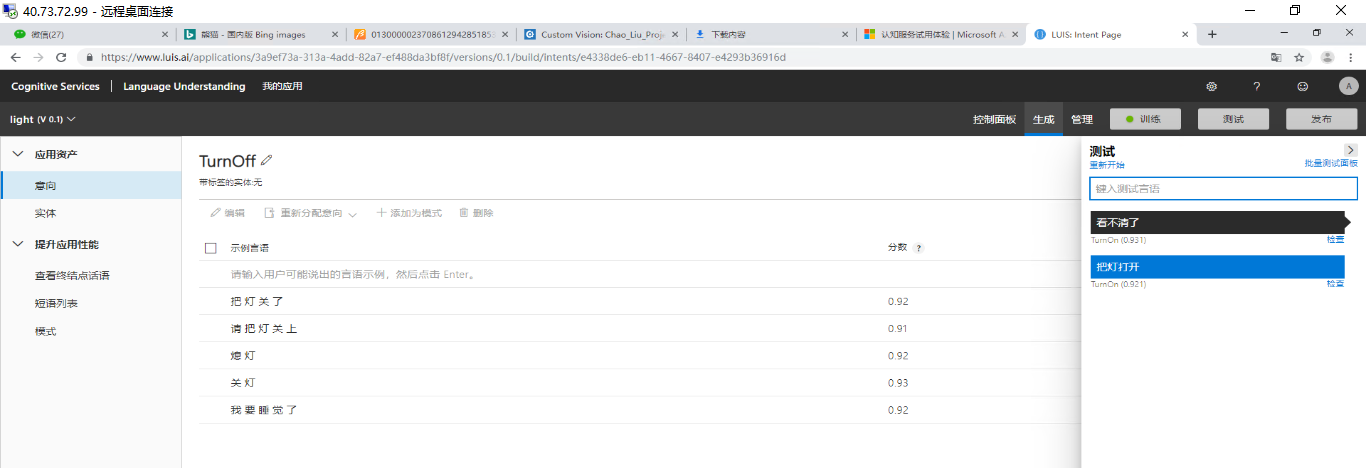




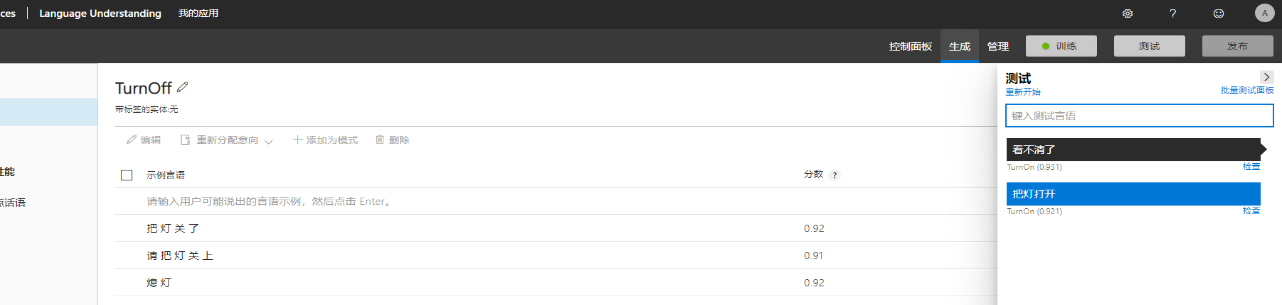
点“训练”对语言识别模型进行训练。

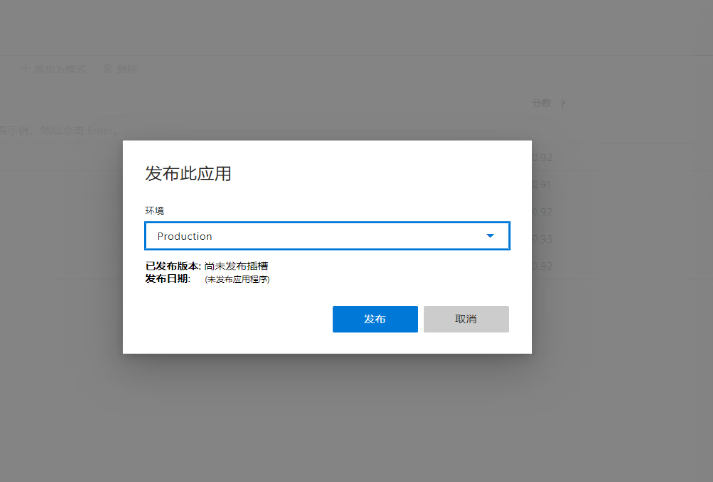


点“测试”，可以测试模型的效果，例如可以输入一个之前没有写的文字“看不清了”，它会调用TurnOn.



点“发布”可以发布并获得这个服务的终结点。

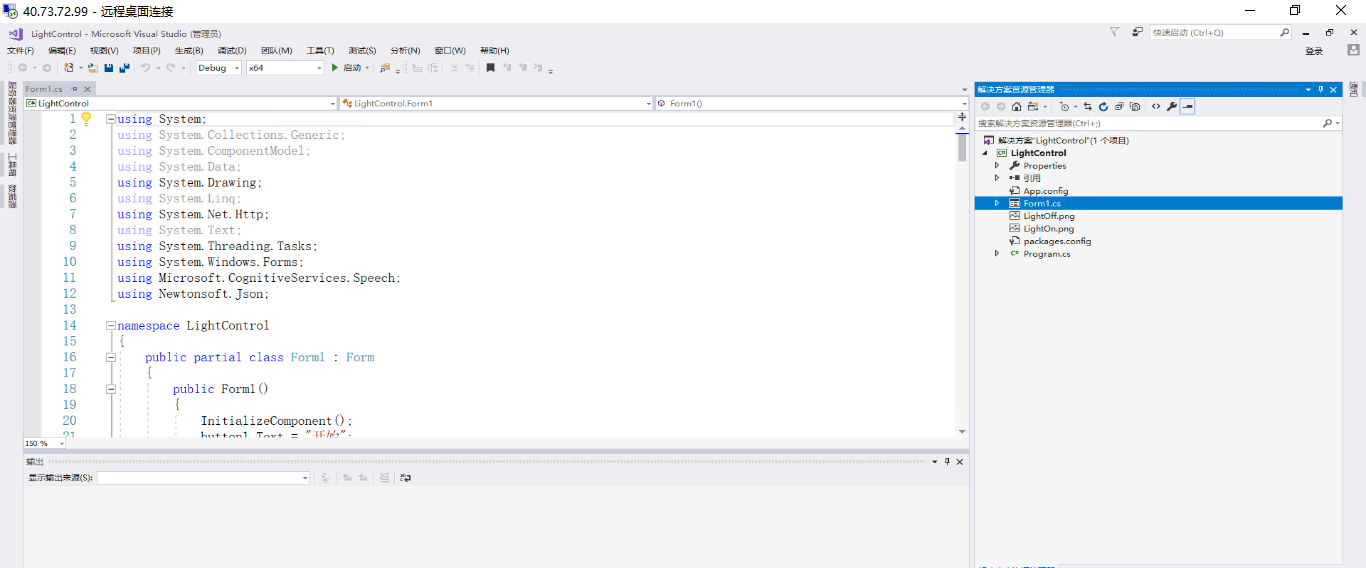


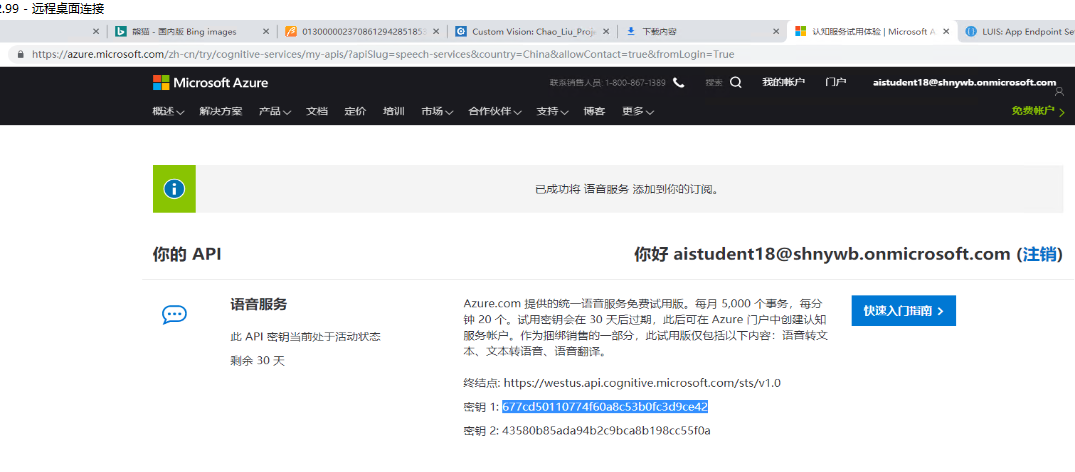




管理🡪密钥和终结点，可以看到终结点的地址。接下来我们用这个终结点的地址，在本地创建一个本地应用程序。

到course的文件夹里面，把light control的目录复制一份到c:\working下，然后找到sln文件，打开。右击forms.cs，选择查看代码。

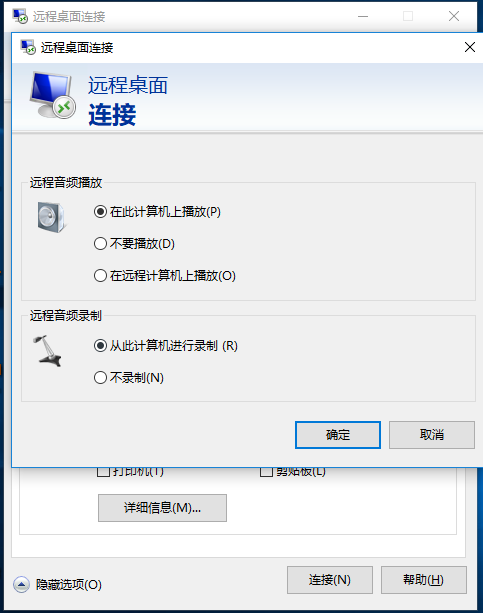




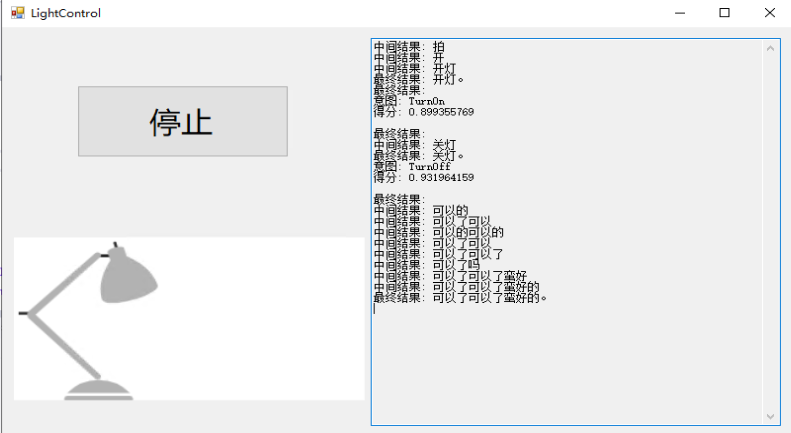




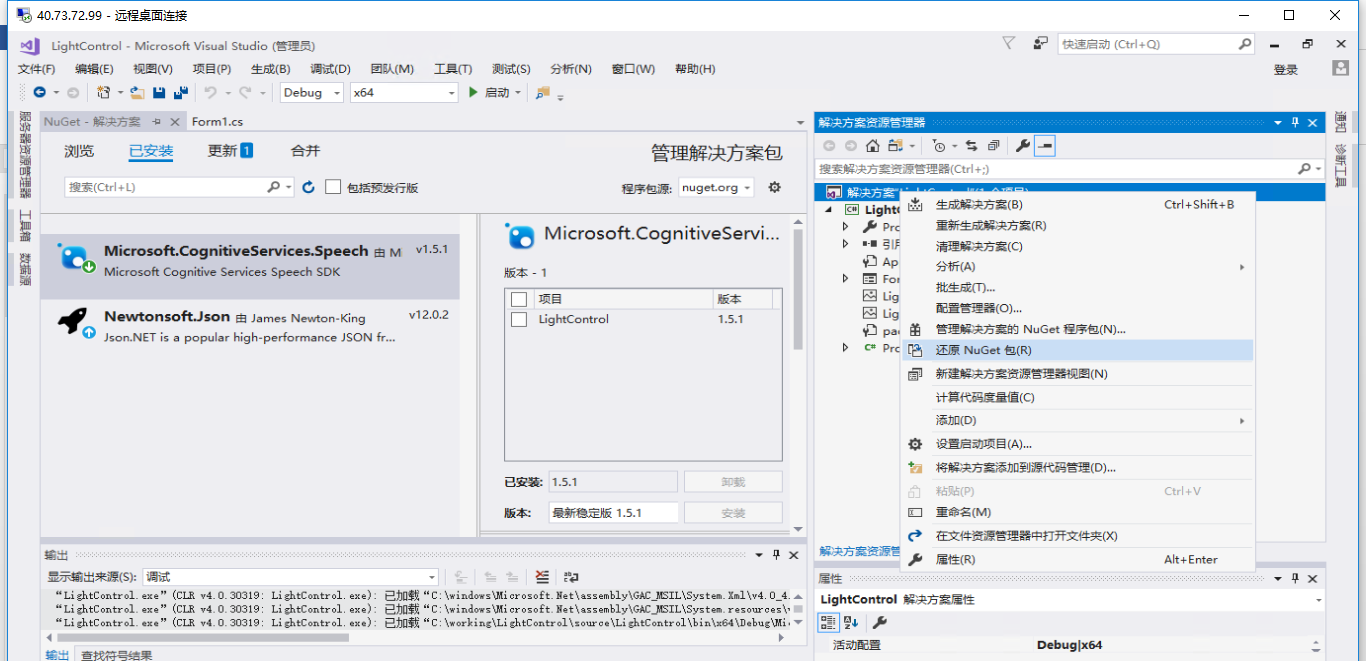
更新speechKey和speechRegion，并打开远程桌面的语音穿透。



然后就可以使用了，如下图：



还有一种通过SDK的方式使用



**三、总结与感悟**

这周的课程虽然只有一天，但信息量极大，主要让我感性地认识到我们所学习的AI课程最后可以实现什么目标（图像分类、目标检测、语义分割、语音识别、语义分析），有哪些应用（智能家居），创建AI应用的流程（数据准备，数据标记，训练，预测），也让我学习到微软在AI方面的实力，我印象最深的就是数据的重要性，就像老师说的“再好的模型，只要数据不够，也无法做出很好的结果”。

课后，老师给我们布置了两个作业：熊的目标检测和灯实体的语音识别支持。作业有一定的难度，难点主要在于我们没有C#和Windows桌面应用的编程基础和Visual Studio 2017的调试基础，但经过自学和试错，我也克服了困难，自主完成了作业的内容，让我对课程上学到的内容有了更深入的理解。