Artificial Intelligence-as-a-Service with IBM Watson
For developers

Roman Boiko,
Clemence Lebrun
May, 2018
Clemence Lebrun, Developer & Developer Advocate Europe
IBM Cloud
clemence.lebrun@fr.ibm.com | @ClemenceLebron | clebrun

Roman Boiko, Hybrid Cloud Integration Technical Sales Specialist at IBM Cloud
1. IBM Cloud Introduction

2. How can Watson help developers to quickly build cognitive apps?

3. Demo: TJBot meeting the human:
   TJBot meets the Human #IBMCloud #Watson #BOT

4. Toolkit for developers to get started with Watson today
Роль облаков

Облачные технологии помогают инновациям и развитию бизнеса

1. Стоимость
   - IaaS-centric
   - Virtual compute
   - Low cost storage
   - Traditional app hosting

2. Инновации
   - PaaS-centric
   - DevOps tooling
   - Web/Mobile apps
   - Basic analytics
   - Hybrid integration

3. Платформа для бизнес идей
   - High value solutions
   - Cognitive apps
   - Advanced analytics
   - Internet of Things
Различные варианты сред исполнения в IBM Cloud

**Bare-Metal**
Максимальная производительность и контроль

**Виртуальный сервер или Vmware**
Использование своих инструментов и языков

**Контейнеры**
Максимальная тиражируемость и переносимость

**Cloud Foundry**
PaaS среда исполнения

**OpenWhisk**
Среда для безсерверных приложений

**ПРОИЗВОДИТЕЛЬНОСТЬ И КОНТРОЛЬ**

**ТИРАЖИРУЕМОСТЬ**

**СКОРОСТЬ РАЗРАБОТКИ**
IBM Cloud

DevOps Tooling

Your Own Hosted Apps / Services

Catalog of Services that Extend Apps’ Functionality

Flexible Compute Options to Run Apps / Services

Integration & API Mgmt

Platform Deployment Options to meet Workload Requirements

© 2017 IBM Corporation
<table>
<thead>
<tr>
<th>IBM Watson APIs for developers &amp; tools</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Speech</strong></td>
</tr>
<tr>
<td>Text To Speech</td>
</tr>
<tr>
<td>Speech To Text</td>
</tr>
<tr>
<td><strong>Language</strong></td>
</tr>
<tr>
<td>Language Translator</td>
</tr>
<tr>
<td>Natural Language Classifier</td>
</tr>
<tr>
<td><strong>Empathy</strong></td>
</tr>
<tr>
<td>Tone Analyzer</td>
</tr>
<tr>
<td>Personality Insights</td>
</tr>
<tr>
<td><strong>AI Assistant</strong></td>
</tr>
<tr>
<td>Watson Assistant</td>
</tr>
<tr>
<td><strong>Vision</strong></td>
</tr>
<tr>
<td>Visual Recognition</td>
</tr>
<tr>
<td><strong>Knowledge</strong></td>
</tr>
<tr>
<td>Discovery</td>
</tr>
<tr>
<td>Natural Language Understanding</td>
</tr>
<tr>
<td>Knowledge Studio</td>
</tr>
<tr>
<td><strong>Data</strong></td>
</tr>
<tr>
<td>Watson Studio</td>
</tr>
<tr>
<td>Watson Machine Learning</td>
</tr>
<tr>
<td>Watson Knowledge Catalog</td>
</tr>
</tbody>
</table>
IBM Watson: the Visual Recognition example

A set of available built-in models provides highly accurate results without training:

- **General model**: Default classification from thousands of classes.
- **Face model**: Facial analysis with age and gender.
- **Explicit model (Beta)**: Whether an image is inappropriate for general use.
- **Food model (Beta)**: Specifically for images of food items.
- **Text model (Private beta)**: Text extraction from natural scene images.

You can also train custom models to create specialized classes.

**How To use the service:**

Prepare images
Gather images to analyze.

Analyze images
Use the built-in capabilities or your own custom models.

View Results
Review the insights into your visual content.

Prepare training data
Sort images into positive or negative images.

Train and create new models
Upload your examples as training data.
IBM Watson: the Visual Recognition example

**Visual Recognition**

The IBM Watson™ Visual Recognition service uses deep learning algorithms to identify scenes, objects, and faces in images you upload to the service. You can create and train a custom classifier to identify subjects that suit your needs.

For more information about this service, see the API reference:


### General

- **GET** `/v3/classify`  
  Classify an image

- **POST** `/v3/classify`  
  Classify images

### Face

- **GET** `/v3/detect_faces`  
  Detect faces in an image

- **POST** `/v3/detect_faces`  
  Detect faces in images

### Custom

- **GET** `/v3/classifiers`  
  Retrieve a list of classifiers

- **POST** `/v3/classifiers`  
  Create a classifier

- **DELETE** `/v3/classifiers/{classifier_id}`  
  Delete a classifier

- **GET** `/v3/classifiers/{classifier_id}`  
  Retrieve classifier details

- **POST** `/v3/classifiers/{classifier_id}`  
  Update a classifier
IBM Watson: the Visual Recognition example

Insurance (Custom Classifier)
Custom Classifier trained on insurance images

<table>
<thead>
<tr>
<th>Category</th>
<th>Confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>flat_tire</td>
<td>0.91</td>
</tr>
<tr>
<td>vandalism</td>
<td>0.00</td>
</tr>
<tr>
<td>broken_windshield</td>
<td>0.00</td>
</tr>
<tr>
<td>motorcycle_accident</td>
<td>0.00</td>
</tr>
</tbody>
</table>

International vehicle glass repair company Belron uses Custom Models to automatically generate estimates of repair costs based on customer-submitted images of car damage.

Select an image on the left to evaluate how this Custom Model analyzes different images.

Build your own Custom Model for free
IBM Watson: the Watson Assistant example

1. **Create Intents:** An intent is a category that defines a user's goal or purpose. You can think of intents as the actions your users might want to perform with your application.

   ➔ You can add pre-defined intents from a content catalog

2. **Create Entities:** Watson's way of handling significant parts of an input that should be used to alter the way it responds to the intent

   ➔ You can use existing trained-entities such as currency, dates ..
3. Try it out! (But wait until training is done)

If the phrase is wrongly identified, you have the ability to change it in the Try it out panel and the service will retrain with your updates.

4. Design the dialog flow and responses:
A dialog defines the flow of your conversation in the form of a logic tree. Each node of the tree has a condition that triggers it, based on user input. A Dialog uses Intents and Entities identified, plus Context of the application to interact with the user, and automatically provide a response.

You can add a response condition.
IBM Watson Maker Kits are a collection of DIY open source templates to connect to Watson services in a fun way. TJBot is the first maker kit in the collection and was created as an experiment to find the best practices in the design and implementation of cognitive objects.

**Build TJBot:**
You can download the design files an laser cut or 3D print TJBot. Here is an instructable to help you with the details of how to build your Bot

**Bring TJBot to life!**
Recipes are step by step instructions to help you connect your TJBot to Watson services. The recipes are designed based on a Raspberry Pi. You can either run one of the available recipes or create your own.
View recipes on github and Instructables.
Demo: TJBot, how does it works?

**Hardware:** Raspberry Pi, Camera, Servo Motor, Speakers, Microphone, SenseHat (optional)

**Software:**
- Node-RED running on the Raspberry Pi
- **IBM Watson APIs** called with Node-RED
- Raspberry integration with Node-RED (GPIO, camera, microphone, speakers nodes...)

Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways.
A Toolkit for developers to get started with Watson today

Tech talks, Open source, Tutorials …
• ibm.com/developerworks
• developer.ibm.com/code/
• medium.com/ibm-watson
• ibm.com/watson/
• console.bluemix.net/docs

TJBot
https://ibmtjbot.github.io/
https://github.com/binnes/tobyjnr/wiki

Events
• developer.ibm.com/code/community/events
Thank you!