

Caption Generator for Images powered by Watson Visual Recognition

1. INTRODUCTION

1.1 Overview

In the past few years, computer vision in image processing area has made significant progress, like image classification and object detection. Benefiting from the advances of image classification and object detection, it becomes possible to automatically generate one or more sentences to understand the visual content of an image, which is the problem known as Image Captioning. Generating complete and natural image descriptions automatically has large potential effects, such as titles attached to news images, descriptions associated with medical images, text-based image retrieval, information accessed for blind users, human-robot interaction.

Image Captioning refers to the process of generating textual description from an image based on the objects and actions in the image. The generated text is expected to describe, in a single sentence, what is visually depicted in the image, for example the entities or objects present in the image, their attributes, the actions or activities performed. The ability to recognize image features and generate accurate, syntactically reasonable text descriptions is important for many tasks in computer vision.

1.2 Purpose

Caption generation is an interesting artificial intelligence problem where a descriptive sentence is generated for a given image. It involves the dual techniques from computer vision to understand the content of the image and a language model from the field of natural language processing to turn the understanding of the image into words in the right order. Image captioning has various applications such as recommendations in editing applications, usage in virtual assistants, for image indexing, for visually impaired persons, for social media, and several other natural language processing applications.

2. LITERATURE SURVEY

2.1 Existing Problem

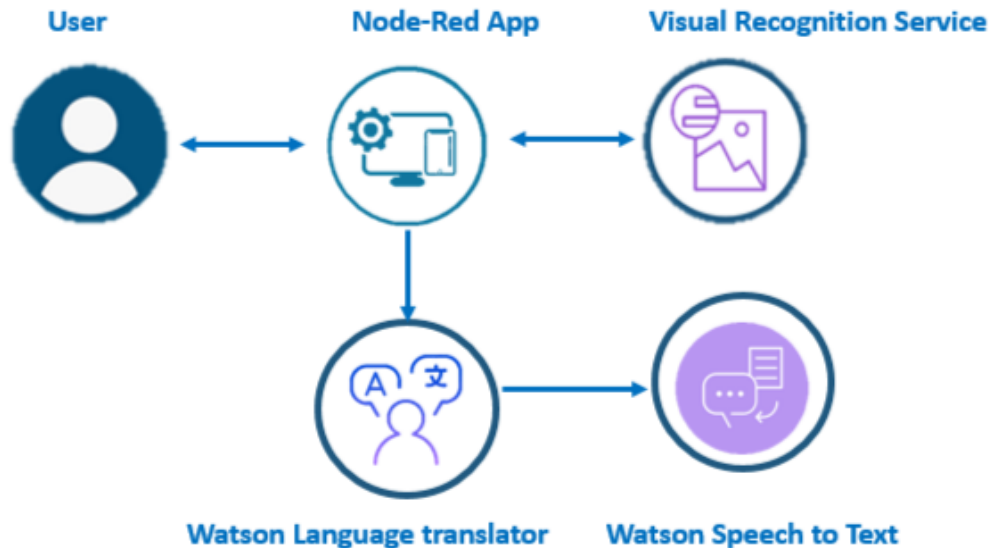
Traditional captioning systems suffer from lack of compositionality and naturalness as they often generate captions in a sequential manner, i.e., next generated word depends on both the previous word and the image feature. This can frequently lead to syntactically correct, but semantically irrelevant language structures, as well as to a lack of diversity in the generated captions.

2.2 Proposed Solution

The project aims at building an application which takes input as image analyses it and generate the captions in the form of speech. To achieve this, I have used IBM Services like node-red service to build a web UI where user uploads a picture. This picture is analyzed by visual recognition service and the analyzed description is then converted in to text to speech service using text to speech service.

3. THEORITICAL ANALYSIS

3.1 Block Diagram



3.2 Software Designing

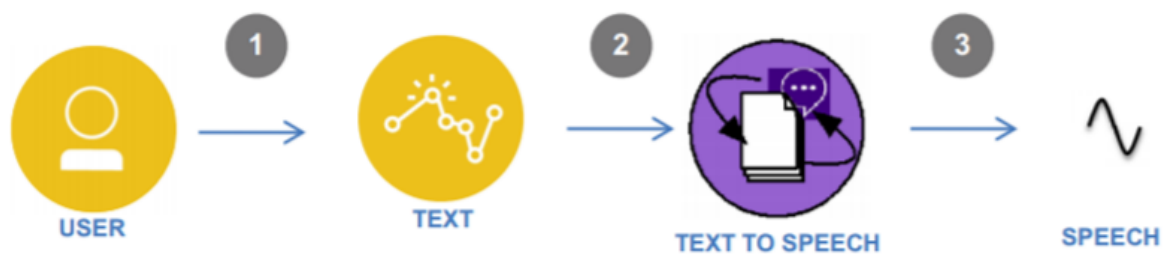
The project uses the following services provided by IBM -

1. IBM Watson Visual Recognition

The IBM Watson Visual Recognition service uses deep learning algorithms to analyze images for content such as objects, scenes, and faces. We can create a Watson Visual Recognition modeler to automatically train a model to classify images for scenes, objects, or our custom content.

2. IBM Watson Text To Speech

Watson Text to Speech is a speech synthesizer API that converts written text into audible speech. It is multilingual, so it accepts text as input and outputs an audio file in various languages. The input text can be plain text or written in Speech Synthesis Markup Language (SSML). Additionally, it outputs various speaking styles, pronunciation, pitch, and speaking rate. We can improve the customer experience and engagement by interacting with users in multiple languages and tones, increase content accessibility for users with different abilities, provide audio options to avoid distracted driving, or automate customer service interactions to increase efficiencies using this service.



3. Node-RED Service

Node-RED is a flow-based development tool for visual programming developed originally by IBM for wiring together hardware devices, APIs and online services as part of the Internet of Things. Node-RED provides a web browser-based flow editor, which can be used to create JavaScript functions. Elements of applications can be saved or shared for re-use. The runtime is built on Node.js. The flows created in Node-RED are stored using JSON. Since version 0.14, MQTT nodes can make properly configured [TLS](#) connections.

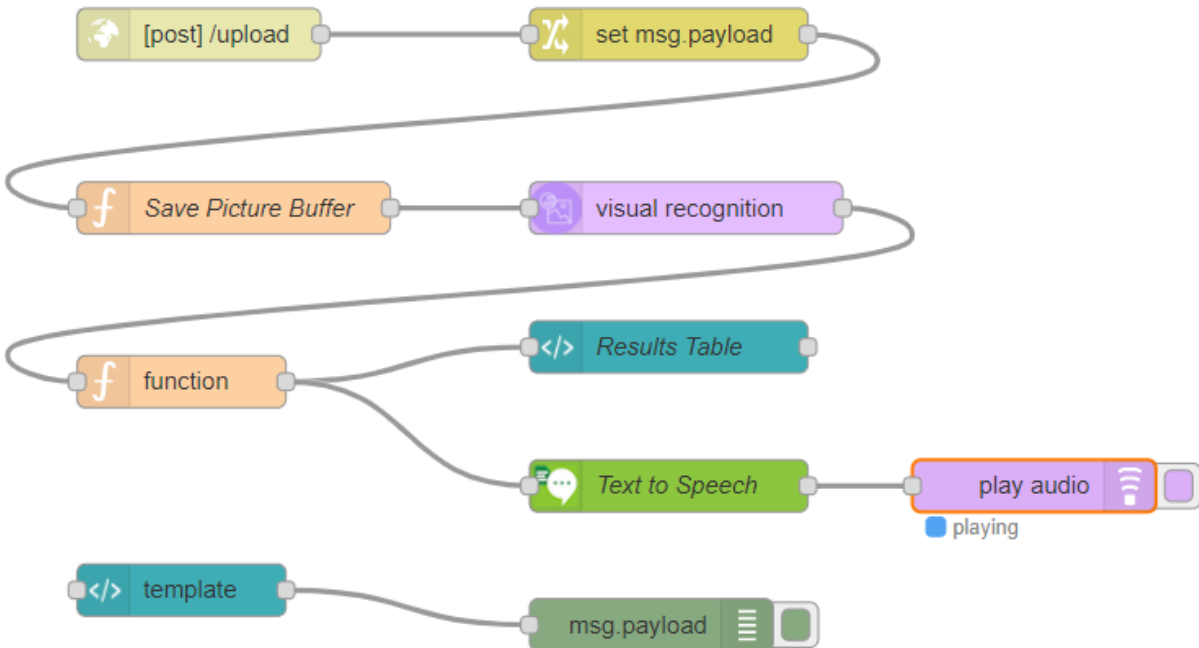
4. EXPERIMENTAL INVESTIGATION

The following are the steps followed to create the web based Application.

1. The template node is used to create the web based UI, where the user have to browse for an image, and upload one to generate caption for.
2. This image then saved as image buffer and then transferred to the visual recognition node.
3. The output of Visual Recognition node is wired to a function node, that generates the caption in a required format.
4. The function node is then wired to the Text to Speech node, which converts the caption generated into speech form and plays the audio through audio node.
5. The function node is also wired to another template node that processess the result into table format showing the classes and confidence score.

5. FLOWCHART

The following node red flow is used for the project.




6. RESULT

The following is the result of deploying the node red flow.

Choose Image

Upload Image



Analyze File

Results

This picture contains a ash grey color Siberian husky dog.

Class	Confidence
Siberian husky dog	0.729
dog	0.966
domestic animal	0.967
animal	0.968
Eskimo dog	0.618
malamute dog	0.598
ash grey color	0.898
gray color	0.872

7. ADVANTAGES AND DISADVANTAGES

7.1 Advantages


1. Intelligent monitoring enables the machine to identify and determine the behaviour of people or vehicles in the captured scene and generate alarms under appropriate conditions to prompt the user to react to emergencies and prevent unnecessary accidents.
2. With the advancements of science and technology and the need for the development of human life, robots have been used in more and more industries. Auto-pilot robots can intelligently avoid obstacles, change lanes and pedestrians based on the road conditions according to the surrounding driving environment they observe.
3. Image and video annotation can also help visually impaired people to understand a large number of videos and pictures on the Internet. The image description is generated automatically.

7.2 Disadvantages

1. It may be difficult to classify the images accurately for the sensitive changes in the image which would not fool a human observer.
2. Datasets are not trained to the Visual recognition model, so might be difficult to caption complicated images having less clarity.
3. As we can see the following example, the flow generates a wrong caption.

Choose Image

Upload Image



Analyze File

Results

This picture contains a jade green color toothbrush.

Class	Confidence
toothbrush	0.665
toiletry	0.665
domestic cat	0.502
cat	0.55
feline	0.559
carnivore	0.559
mammal	0.56
animal	0.562
knocker	0.5
jade green color	0.859

8. APPLICATIONS

1. **Self-driving cars** - Automatic driving is one of the biggest challenges and if we can properly caption the scene around the car, it can give a boost to the self-driving system.

2. **Aid to the blind** - We can create a product for the blind which will guide them travelling on the roads without the support of anyone else. We can do this by first converting the scene into text and then the text to voice. Both are now famous applications of Deep Learning.

3. CCTV cameras are everywhere today, but along with viewing the world, if we can also generate relevant captions, then we can raise alarms as soon as there is some malicious activity going on somewhere. This could probably help reduce some crime and/or accidents.

4. **Media and Publishing Houses**- The media and public relations industry circulate tens of thousands of visual data across borders in the form of newsletters, emails, etc. The image captioning model accelerates subtitle creation and enables executives to focus on more important tasks.

5. **Social Media Posts** - For social media, artificial intelligence is moving from discussion rooms to underlying mechanisms for identifying and describing terabytes of media files. It enables community administrators to monitor interactions and analysts to formulate business strategies.

9. CONCLUSION

Caption Generator for Images powered by Watson Visual Recognition is a basic application, which uses Node-RED services to create user interface and to generate the caption for the image given as input using . The image is classified using IBM Watson Visual Recognition service, the result table with various classes and their confidence is displayed along with the suitable caption for the image and the generated caption is then converted to speech using IBM Watson text to speech service.

10. FUTURE SCOPE

The future scope of this project might be to create an application to generate captions for multiple images provided as input simultaneously. The application might also be able to compare images and specify the differences between them.

11. BIBLIOGRAPHY

1. <https://artificialintelligence.oodles.io/blogs/ai-powered-image-caption-generator/>
2. <https://en.wikipedia.org/wiki/Node-RED>
3. <https://www.hindawi.com/journals/cin/2020/3062706/>
4. <https://www.ibm.com/blogs/research/2019/06/image-captioning/>
5. <https://node-red.gitbook.io/node-red-twitter/more-node-red-flows/twitter-image-analysis>
6. https://github.com/watson-developer-cloud/node-red-labs/tree/master/basic_examples/visual_recognition
7. https://www.matec-conferences.org/articles/matecconf/pdf/2018/91/matecconf_eitce2018_01052.pdf

12. APPENDIX

12.1 Source Code

```
1 [{"id":"7d8881d0.b45c7","type":"tab","label":"Project","disabled":false,"info":""},{
  "id":"b7c5c004.2b7ca","type":"function","z":
  "7d8881d0.b45c7","name":"","func":
  "if (typeof msg.result == 'undefined') {\n
  return null;\n}\n\nif (typeof msg.result.error != 'undefined') {\n
  msg.template = msg.result.error.message;\n
  return msg;\n}\n\n// Text Extraction\nif (typeof msg.result.images[0].text != 'undefined')\n{\n
  var image_text = msg.result.images[0].text;\n
  msg.payload = image_text;\n
  msg.template = image_text;\n
  if( image_text.length >0 ) {\n
  msg.template= \"Watson found the words: \"+image_text;\n
  }\n
  return msg;\n}\n\nvar bestcolor = -1;\nvar colorscore = 0;\nvar c_id = 0;\nvar say = \"\";\nvar item;\n\nfor ( c_id=0; c_id < (msg.result.images[0].classifiers.length); c_id++) {\n
  // find the best color, if any\n
  for( i =0; i<(msg.result.images[0].classifiers[c_id].classes.length); i++) {\n
    var object = msg.result.images[0].classifiers[c_id].classes[i].class;\n
    if ( object.includes(\"color\") ) {\n
      if( msg.result.images[0].classifiers[c_id].classes[i].score > colorscore){\n
        bestcolor = i;\n
        colorscore = msg.result.images[0].classifiers[c_id].classes[i].score;\n
      }\n
    }\n
  }\n\n  var bestItem = 0;\n  var itemScore = 0;\n\n  for( i =0; i<(msg.result.images[0].classifiers[c_id].classes.length); i++) {\n
    object = msg.result.images[0].classifiers[c_id].classes[i].class;\n
    if ( !object.includes(\"color\") ) {\n
      if( msg.result.images[0].classifiers[c_id].classes[i].score > itemScore){\n
        //bestItem = i;\n
        bestItem =
```

```

0;\n                itemScore =
msg.result.images[0].classifiers[c_id].classes[i].score;\n
}\n        }\n        }\n        \nif( bestcolor != \"-1\)") {\n                //
found a color\n                item =
msg.result.images[0].classifiers[c_id].classes[bestcolor].class
+ \ " \ " +
msg.result.images[0].classifiers[c_id].classes[bestItem].class;
\n                bestcolor = -1;\n        } else {\n                item =
msg.result.images[0].classifiers[c_id].classes[bestItem].class;
\n        }\n        say = say + \ " This picture contains a \ " + item
+ \ ". \ ";\n}\nmsg.payload = say;\n\nvar picInfo =
msg.result.images[0].classifiers[0].classes;\nvar arrayLength =
picInfo.length;\n\nmsg.template=\ "<style>\n";\nmsg.template=msg.
template+\ "table { width: 440px; margin-top: 10px;
}\n";\nmsg.template=msg.template+\ "tr:nth-child(even){background
-color: #f2f2f2;}\n";\nmsg.template=msg.template+\ "th, td {
padding: 8px; text-align: left; border-bottom: 1px solid #ddd;
width:
10%;}\n";\nmsg.template=msg.template+\ "</style>\n";\n\nmsg.templa
te=msg.template+\ "<h2>\n"+say+\ "</h2><table
span=100%><tr><th>Class</th><th>Confidence</th></tr>\n";\nfor
(var i = 0; i < arrayLength; i++) {\n msg.template =
msg.template + \ "<tr><td>\n" + picInfo[i].class + \ "</td><td>\n"
+ picInfo[i].score + \ "</td></tr>\n";\n}\n\nmsg.template =
msg.template + \ "</table>\n";\n\nreturn
msg;" ,"outputs":1,"noerr":0,"initialize":"","finalize":"","x":1
60,"y":320,"wires":[["259462eb.4f5c7e","af0b9fc9.a35d1"]]},{"id
":"259462eb.4f5c7e","type":"ui_template","z":"7d8881d0.b45c7","
group":"4596c977.ce05e8","name":"Results
Table","order":1,"width":9,"height":10,"format":"","storeOut
Messages":true,"fwdInMessages":true,"resendOnRefresh":false,"t
emplateScope":"local","x":440,"y":300,"wires":[[]]},{"id":"ba0c
36bb.47f618","type":"debug","z":"7d8881d0.b45c7","name":"","act
ive":true,"tosidebar":true,"console":false,"tostatus":false,"co
mplete":"payload","targetType":"msg","statusVal":"","statusType
":"auto","x":430,"y":460,"wires":[[]]},{"id":"8797c69a.181f38","t
ype":"ui_template","z":"7d8881d0.b45c7","group":"8247190c.62ded

```

```

8","name":"","order":0,"width":"9","height":"10","format":"<html>\n
  <body>\n
    <form action="/upload"
method="POST" enctype="multipart/form-data">\n
<p><input type="file" accept="image/*" name="image"
id="file_id" onchange="loadFile(event)" style="display:
none;"></p>\n
    <p><label for="file_id" style="cursor:
pointer;">Upload Image</label></p>\n
    <p><img
id="output" width="400" /></p>\n
    <input
type="submit" value="Analyze File">\n
    </form>\n
\n
    <script>\n
      var loadFile = function(event) \n
{\n\t
  var image = document.getElementById('output');\n\t
  image.src = URL.createObjectURL(event.target.files[0]);\n\t
\n\t
  };\n
    </script>\n
\n
</body>\n</html>","storeOutMessages":true,"fwdInMessages":true,
"resendOnRefresh":true,"templateScope":"local","x":160,"y":440,
"wires":[["ba0c36bb.47f618"]],{"id":"f4f3e32c.84d58","type":"v
isual-recognition-v3","z":"7d8881d0.b45c7","name":"","vr-servic
e-endpoint":"https://api.us-south.visual-recognition.watson.clo
ud.ibm.com","image-feature":"classifyImage","lang":"en","x":450
,"y":220,"wires":[["b7c5c004.2b7ca"]],{"id":"4ad44dfd.b328e4",
"type":"http
in","z":"7d8881d0.b45c7","name":"","url":"/upload","method":"po
st","upload":true,"swaggerDoc":"","x":170,"y":120,"wires":[["37
2c6645.5b1dfa"]],{"id":"372c6645.5b1dfa","type":"change","z":"
7d8881d0.b45c7","name":"","rules":[{"t":"set","p":"payload","pt
":"msg","to":"req.files[0].buffer","tot":"msg"}],"action":"","p
roperty":"","from":"","to":"","reg":false,"x":440,"y":120,"wire
s":[["a755502f.3d181"]],{"id":"a755502f.3d181","type":"functio
n","z":"7d8881d0.b45c7","name":"Save Picture Buffer","func":"if
(msg.req.files[0].mimetype.includes('image')) {\n  msg.mypic
= ``; \n} else {\n  msg.payload =
msg.payload.toString(); \n} \n\nreturn
msg;","outputs":1,"noerr":0,"initialize":"","finalize":"","x":1
90,"y":220,"wires":[["f4f3e32c.84d58"]],{"id":"af0b9fc9.a35d1"
,"type":"watson-text-to-speech","z":"7d8881d0.b45c7","name":"Te

```

xt to

```
Speech","lang":"en-US","langhidden":"en-US","langcustom":"NoCustomisationSetting","langcustomhidden":"","voice":"en-US_OliviaV3Voice","voicehidden":"en-US_OliviaV3Voice","format":"audio/mp3","password":"Kunal23/7/98","apikey":"CqZu6xQ6SLD5hEdsYfkdUz4NIgLvRFFf7qdEYGNzmVWh","payload-response":true,"service-endpoint":"https://api.us-south.text-to-speech.watson.cloud.ibm.com","x":440,"y":380,"wires":[["97c7e1ab.f5b61"]]},{"id":"97c7e1ab.f5b61","type":"playaudio","z":"7d8881d0.b45c7","name":"","voice":"0","x":650,"y":380,"wires":[]},{"id":"4596c977.ce05e8","type":"ui_group","z":"","name":"Results","tab":"1894617.6c6ab9f","order":3,"disp":true,"width":"9","collapse":false},{"id":"8247190c.62ded8","type":"ui_group","z":"","name":"ChooseImage","tab":"1894617.6c6ab9f","order":1,"disp":true,"width":"9","collapse":true},{"id":"1894617.6c6ab9f","type":"ui_tab","z":"","name":"ImageAnalysis","icon":"dashboard","disabled":false,"hidden":false}]
```