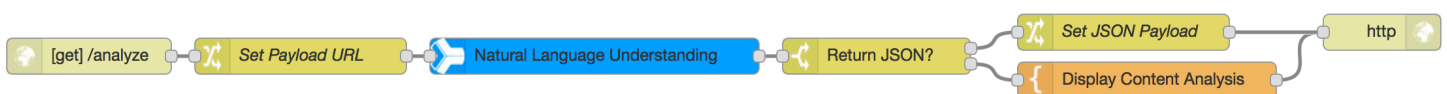


Natural Language Understanding in Node-RED

Hands-On Lab

JeanCarl Bisson | jbisson@us.ibm.com | [@dothewww](#)



Extract keywords, entities, concepts, sentiment and more from a news article
(see *Analyze a News Article in Node-RED*)



A digital copy of this lab and code snippets can be found at:
<http://ibm.biz/node-red-natural-language-understanding>



Add Natural Language Understanding Service in IBM Bluemix

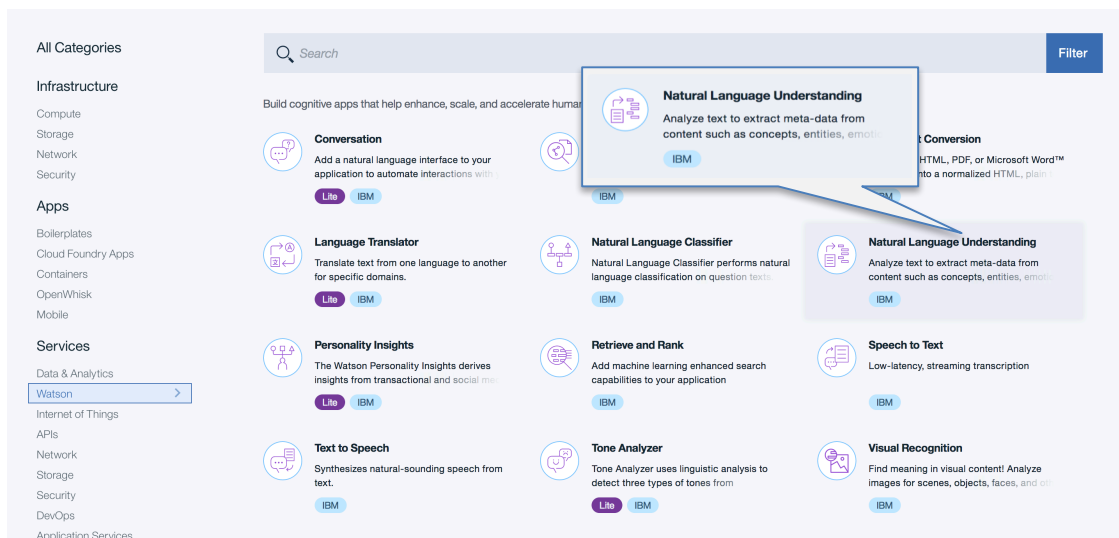
The Natural Language Understanding service analyzes text to extract meta-data from content such as concepts, entities, keywords, categories, sentiment, emotion, relations, semantic roles, using natural language understanding. This tutorial uses the Node-RED boilerplate in IBM Bluemix with the Natural Language Understanding service found under the **Boilerplates** section of the IBM Bluemix catalog.

To get started using the Natural Language Understanding service, you'll need to create service credentials.

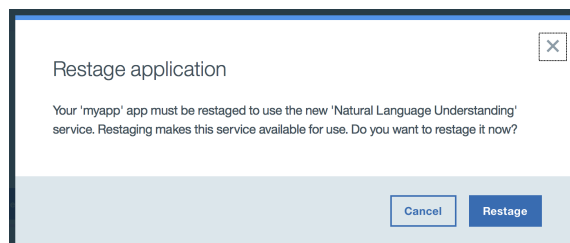
1. Go to the **Connections** tab in the application overview for your Node-RED application in the IBM Bluemix dashboard. Click on **Connect New**.



2. Click on the **Natural Language Understanding** tile under the Watson section. Click on **Create**.



3. IBM Bluemix will prompt to restage the application. Click on **Restage**. The application will restart and include the new service credentials in the environment.

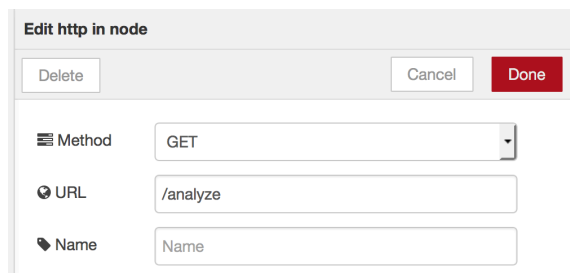


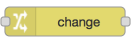
4. When the application has finished restaging, open the Node-RED Flow Editor. If you already have Node-RED open, refresh the page.

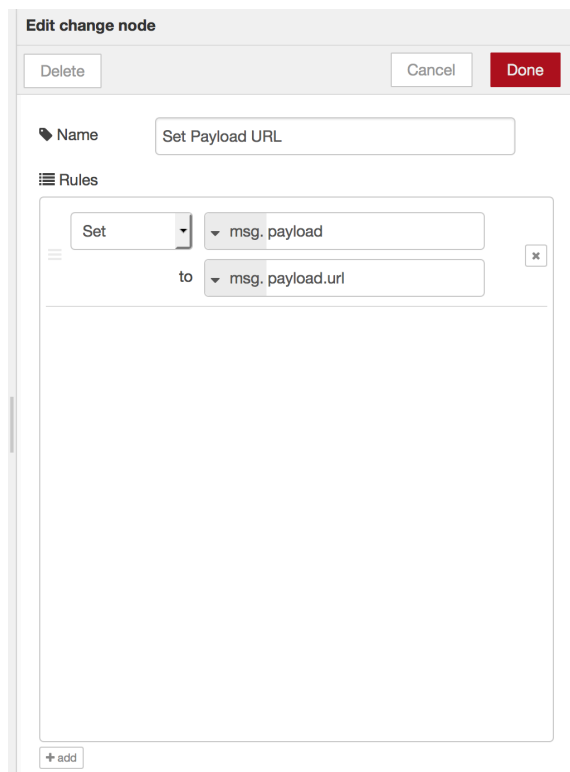
Analyze a News Article in Node-RED


The Watson Natural Language Understanding service takes either a body of text or a publicly-accessible URL to content which the service can analyze. In this section, we will analyze a news article that is accessible via an URL. You can also choose to analyze other URLs that contain a body of text. Please refer to the **Add Natural Language Understanding Service in IBM Bluemix** section to create and bind the Natural Language Understanding service to your Node-RED application.

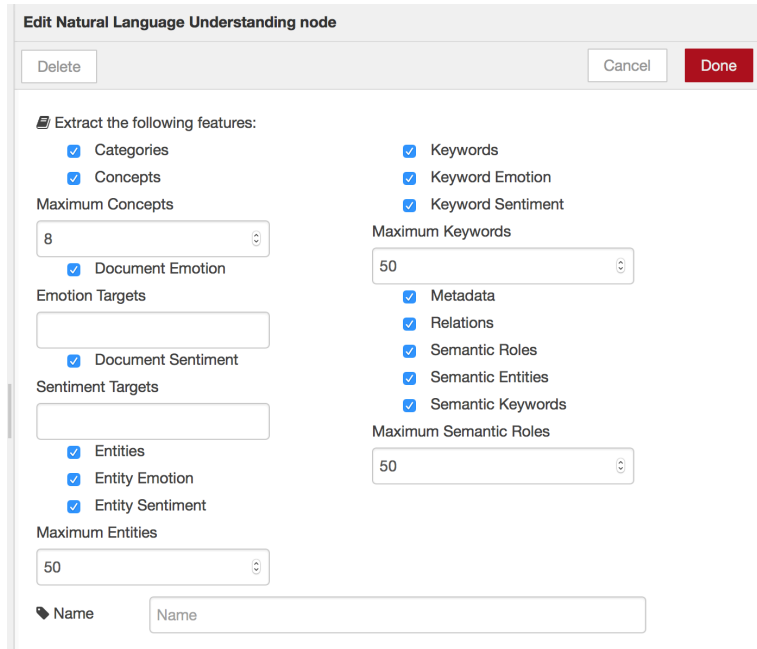
1. Add a  node as shown below.

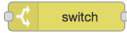


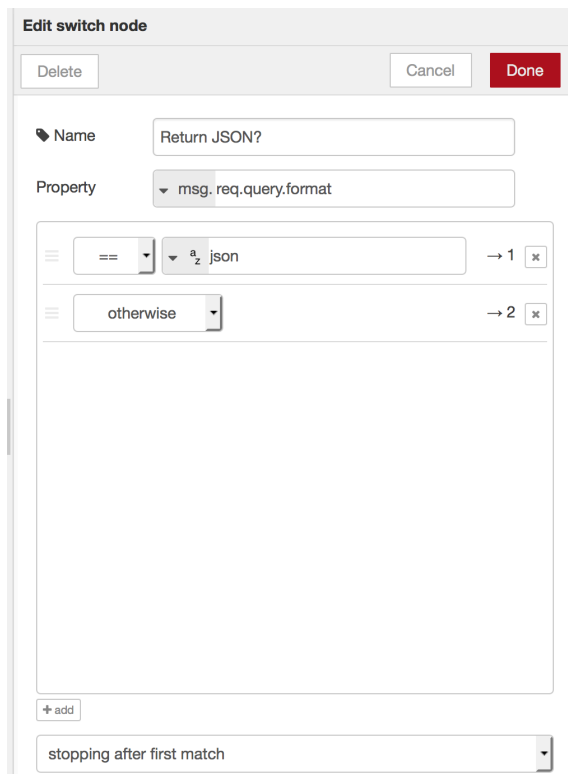
2. Add a  node as shown below. This will take the *url* query parameter and place it in the message *payload* to be passed to the Natural Language Understanding node in the next step.



3. Add a  node. Check the options that you want to be included in the response. You can narrow the results by changing the maximum values and/or by unchecking checkboxes next to each feature. We have enabled all the options to showcase many of the insights returned by the Watson service.



4. To make this application versatile, we'll split the flow so our application will return the results in two formats. The first option will be where the results are returned in JSON format, great for use in applications that can call the web endpoint and consume the JSON. The second option will be a webpage showing the data in a human readable report. Add a  node as shown below.



- For the flow where the JSON should be returned, we can simply return the contents of *msg.features*. Add a node as shown below. This will move the results from the *features* property to the message *payload*.

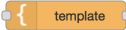
Edit change node

Delete Cancel Done

Name Set JSON Payload

Rules

Set msg. payload to msg. features

- For the flow where a webpage should be returned, add a  node with the HTML from the file at ibm.biz/BdiBpU

Edit template node

Delete Cancel Done

Name Display Content Analysis

Set property msg. payload

Template Syntax Highlight: mustache

```

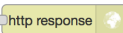
1 <h1>{{features.metadata.title}}</h1>
2 <table>
3   <tr>
4     <td>Author</td>
5     <td>{{#features.metadata.authors}}{{name}} {{/features.metadata.authors}}</td>
6   </tr>
7   <tr>
8     <td>Document Sentiment</td>
9     <td>{{features.sentiment.document.label}} (Score: {{features.sentiment.document.score}})</td>
10  </tr>
11  <tr>
12    <td>URL</td>
13    <td><a href="{{req.query.url}}" target="_blank">{{req.query.url}}</a></td>
14  </tr>
15  <tr>
16    <td>Published</td>
17    <td>{{features.metadata.publication_date}}</td>
18  </tr>
19  <tr>
20    <td></td>
21    <td><a href="/story?format=json&url={{req.query.url}}">View JSON</a></td>
22  </tr>
23 </table>
24
25 <h2>Entities</h2>
26 <table border="1">
27   <tr>
28     <th>Entity</th>

```


Format Mustache template



Get the code:
ibm.biz/BdiBpU

- Add a  node. Connect the nodes together as shown below.



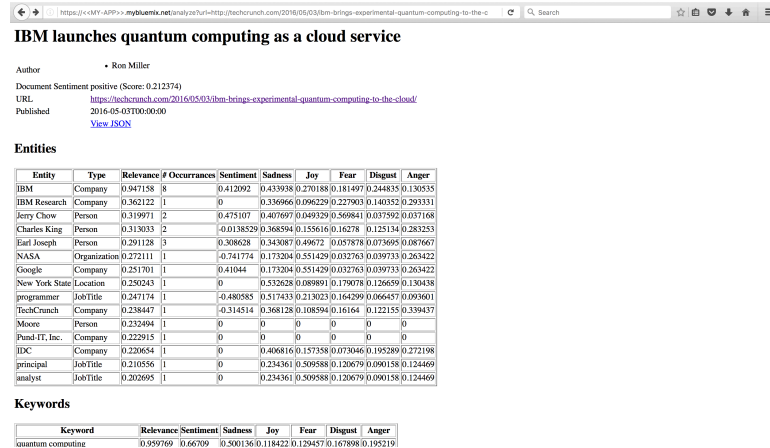
- Click on the red  button in the top-right corner of the screen to save and deploy your changes.
- Visit a news website and copy the URL of a publicly accessible news article. Make sure the content isn't behind an authentication wall where you need to sign in to access the content.

10. Open a browser tab and visit your application's endpoint, passing in the URL to the content:

`http://<<MY-APP>>.mybluemix.net/analyze?url=<<URL-TO-STORY>>`

- Replace <<MY-APP>> with the host of the Node-RED application you chose.
- Replace <<URL-TO-STORY>> with the URL of the content.

11. Depending on the content located at the URL, you may see a list of attributes including concepts, entities, keywords, categories, sentiment, emotion, relations, semantic roles and more mentioned within the text.



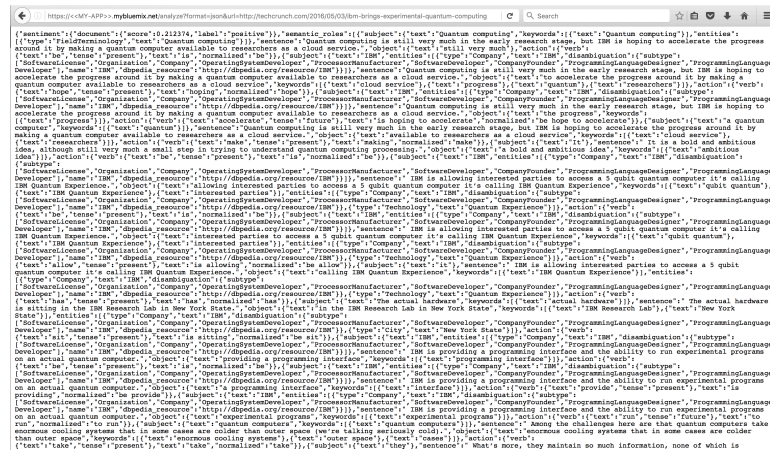
The screenshot shows a web browser displaying the article "IBM launches quantum computing as a cloud service" by Ron Miller. Below the article text, there is a section titled "Entities" which contains a table with sentiment analysis results. The table has columns for Entity, Type, Relevance, # Occurrences, Sentiment, Sadness, Joy, Fear, Disgust, and Anger. The entities listed include IBM, IBM Research, Jerry Chow, Charles King, Earl Joseph, NASA, Google, New York State, programmer, TechCrunch, Moore, Pund-It, Inc., IDC, principal, and analyst, each with associated sentiment scores.

Entity	Type	Relevance	# Occurrences	Sentiment	Sadness	Joy	Fear	Disgust	Anger
IBM	Company	0.947158	8	0.412092	0.433938	0.270188	0.181497	0.244835	0.130535
IBM Research	Company	0.362122	1	0	0.330966	0.096229	0.227903	0.140352	0.203331
Jerry Chow	Person	0.319971	2	0.474107	0.407949	0.040320	0.569841	0.037592	0.037168
Charles King	Person	0.313033	2	0.0138529	0.368594	0.155616	0.16278	0.125134	0.28355
Earl Joseph	Person	0.291128	3	0.308628	0.343087	0.49672	0.057878	0.073695	0.087667
NASA	Organization	0.272111	1	-0.341774	0.173204	0.551429	0.032763	0.039753	0.263422
Google	Company	0.251701	1	0.410444	0.173204	0.551429	0.032763	0.039753	0.263422
New York State	Location	0.250243	1	0	0.532628	0.089891	0.179078	0.126659	0.130438
programmer	JobTitle	0.247174	1	-0.480585	0.517433	0.213023	0.164299	0.066457	0.093601
TechCrunch	Company	0.234514	1	0.314514	0.368126	0.108594	0.16164	0.122153	0.339437
Moore	Person	0.233494	1	0	0	0	0	0	0
Pund-It, Inc.	Company	0.222915	1	0	0	0	0	0	0
IDC	Company	0.220654	1	0	0.406816	0.157358	0.073046	0.195289	0.272198
principal	JobTitle	0.210556	1	0	0.234361	0.309588	0.120679	0.090158	0.124469
analyst	JobTitle	0.202695	1	0	0.234361	0.309588	0.120679	0.090158	0.124469

Keyword	Relevance	Sentiment	Sadness	Joy	Fear	Disgust	Anger
quantum computing	0.959769	0.66709	0.500136	0.118422	0.129457	0.167898	0.195219

12. To see the JSON representation of the content insert *format=json* in the URL query string:

`http://<<MY-APP>>.mybluemix.net/analyze?format=json&url=<<URL-TO-STORY>>`



The screenshot shows a web browser displaying the JSON representation of the sentiment analysis results. The JSON object contains a 'sentiment' field with a 'score' of 0.66709 and a 'label' of 'positive'. It also includes a 'semantic_roles' field with a list of entities and their roles, and a 'keywords' field with a list of keywords. The JSON is formatted as a single line of text.

13. Return to Step #3 and experiment by disabling some of the features to see how the results change. Try analyzing other URLs and see what results are returned.