PROJECT REPORT

By Shaharyaar Kutchi

Table Of Contents

1. INTRODUCTION

1.1 Overview

1.1.1 Project Summary

The web is home to massive amounts of data, with more being created every day. Organizations can harness this constant stream of information to gain understanding, plan strategies, and find opportunities. Enriched news data can help your application make dynamic connections across current events faster.

In this project, we built a NEWS mining web application starting with the basics, using Node-RED / Python Web App and the IBM Watson Discovery Service.

To do this we start by building a Server-Side Application using Node-RED then we use the pre-built Watson Discovery News collection and access the Watson Discovery Service through the Discovery API. Optionally, we can choose to use a Slack interface to query the data, push news alerts out to web notification and deploy the app on IBM Cloud.

1.1.2 Purpose

The purpose of the report is to improve news fetching methodologies. Today, businesses want to know what buyers say about their brand and how they feel about their products. However, with all of the "noise" filling our email, social and other communication channels, listening to customers has become a difficult task. In this guide to sentiment analysis, we'll learn how a bot uses artificial intelligence-based approach can provide customer insight on a massive scale and ensure that you don't miss a single conversation.

2. LITERATURE SURVEY

2.1 Existing problem

a. *Too much information* - It goes without saying that the distributed content production of individuals is an amazing and powerful phenomenon, but relying on search engines to index, sort and rank externally-produced content that isn't standardized for format doesn't work for the newsreader. There's still too much information and the organization is distorted by advertising dollars.

b. *News is time-sensitive information* - There should be constant organizing externally-produced content and figure out that new content was produced and, then organize all of the content that comes in many different structural formats and it has to passively (algorithmically) distinguish between news information (time-sensitive) and historical information that isn't time-sensitive.

c. *News is geographically sensitive information* - If you live in Dallas, TX do you want to have to sort through local content from Boston, MA or New York, NY or vice versa to find local news? No, you want to see time-sensitive information about what's happening in your community and NOW! While Geo RSS is out there, it's still nowhere near being adopted as a widely-used content standard.

d. *Distilling quality news* - It is difficult as link-backs aren't necessarily reflective of quality or demand. One has to filter the glut of information hitting the internet 24/7 while distilling the time-sensitive and geographically-sensitive content a newsreader wants.

2.2 Proposed solution

IBM has upgraded its Watson AI platform to boost the ability to understand business lingo that could be used to analyze the English language text and conversations.

The natural language processing capability represents the first commercialization of IBM Research's Project Debater, billed as an AI system capable of debating humans.

The tool is designed to help develop arguments that could ultimately be used to improve machine understanding and help analyze complex human speech. The integration of Project Debater technology with Watson includes a sentiment analysis and "summarization" tools along with the ability to cluster data by topic.

The sentiment analyzer being released this week can detect sentiment shifters because a negative modifier can reverse a speaker's meaning using a phase like "hardly helpful." The analysis tool would also enable the Watson API to identify common idioms like "calling it a day."

The briefing tool extracts text data from different sources to provide a summary of speech and text on a specific topic. The clustering feature allows users to organize data and generate topics.

IBM's <u>Watson Discovery</u> enterprise search platform, the data organizer is aimed at subject experts in specific industries like
  a. Health care
  b. Insurance
  c. Manufacturing

3. THEORITICAL ANALYSIS

3.1 Block diagram
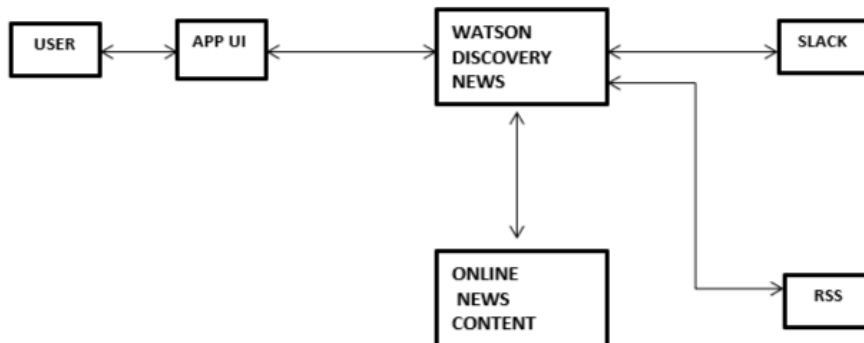
3.2 Hardware / Software designing

1. IBM Cloud
2. Node Red
3. IBM Watson Discovery
4. Node.js
5. GitHub
6. Zoho Writer
7. Slack
8. Html
9. RSS

4. EXPERIMENTAL INVESTIGATIONS

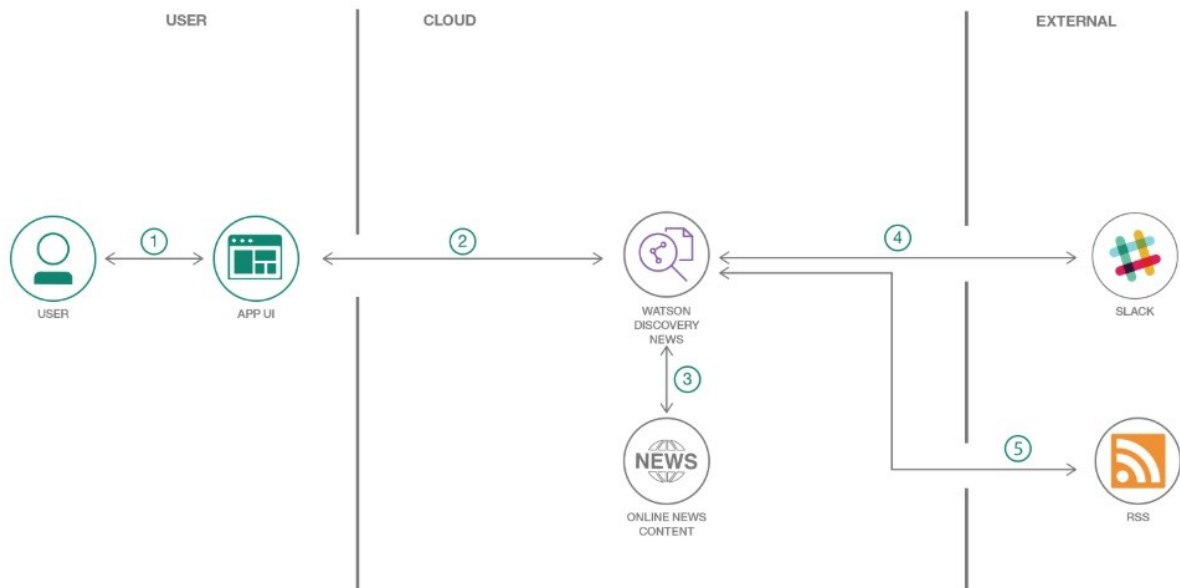This project used is an experiment to the use of Watson Discovery to query news and display it to the user via UI and a slack bot. It was taken into consideration after examining differences in credibility assigned to news read in paper form and the same news read on a web site.

This project includes a bot it is the same as a regular app it can access the same range of APIs and do all of the magical things that a Slack App can do. A bot is a type of Slack App designed to interact with users via conversation. But when you build a bot for your Slack App, you're giving that app a face, a name, and a personality and encouraging users to talk to it.

The bot can send DMs, it can be mentioned by users, it can post messages or upload files, and it can be invited to channels or kicked out. Since bot is capable of doing everything that a Slack App can do, we're going to limit our focus to a common use-case for bots.

5. FLOWCHART

Flow

USER | CLOUD | EXTERNAL

USER — ① → APP UI — ② → WATSON DISCOVERY NEWS — ④ → SLACK

WATSON DISCOVERY NEWS — ③ → ONLINE NEWS CONTENT

⑤ → RSS

6. RESULT

The final application has

a. A user-interface is used to request the extraction of required news content.

b. The app generates the latest news regarding request generated by user using the Watson Discovery News.

c. It generates the URL it used to extract the news from.

d. It displays the author who had submitted the news report.

e. It provides with sentimental analysis that separate positive or negative opinion from good or bad news.

URL Link of Project:

https://node-red-izkrz.eu-gb.mybluemix.net/ui/

GitHubLink:

https://github.com/Shaharyaar7/IISPS-INT-145-AI-Powered-News-Search-App-Level-1-

Project Presentation Video Link:

https://drive.google.com/file/d/1BtW3VOI4pzPyGHWtHnK_gwrsZDPOfxj_/view

Feedback Video Link:

https://drive.google.com/file/d/1LsLh7rvNtJeXd5GTjVfL3eTD-J4Hh-ch/view

7. ADVANTAGES & DISADVANTAGES

7.1 ADVANTAGES

By querying and manipulating enriched data, we can build a more insightful search interface. This code pattern provides a Node.js app built on the Watson Discovery Service. The pattern demonstrates how you can use individual out-of-the-box UI components to extract and visualize the enriched data provided by the Discovery analytics engine.

The main benefit of using the Watson Discovery Service is its powerful analytics engine that provides cognitive enrichments and insights into our data. The app in this code pattern provides examples of how to showcase these enrichments through the use of filters, lists, and graphs. The key enrichments are:

a. *Entities:* People, companies, organizations, cities, and more
b. *Categories:* Classification of the data into a hierarchy of categories up to 5 levels deep
c. *Concepts:* Identified general concepts that aren't necessarily referenced in the data
d. *Keywords:* Important topics typically used to index or search the data
e. *Sentiment:* The overall positive or negative sentiment of each document

The app uses standard search UI components such as filter lists, tag clouds, and sentiment graphs, but also more complex Discovery options such as the passages and highlight features.

With these two features, the app identifies the most relevant snippets in your data based on your query and is more likely to return the data that you're searching for.

7.2 DISADVANTAGES

Application Programming Interface (API)

a. One of the key problems that most of the developer's encounter is the Application Programming Interface (API) keeps on changing at frequent intervals and does not remain stable.

b. At times, a new API appears having a number of backwards-incompatible changes. As a result, the developers are forced to make changes in the accessible code bases to match the compatibility with the latest version of the Node.js API.

Chatbot Disadvantages:

a. *Complex Interface* – Chatbots are often seen to be complicated and require a lot of time to understand user's requirement. It is also the poor processing which is not able to filter results in time that can annoy people.

b. *Inability to Understand* – Due to fixed programs, chatbots can be stuck if an unsaved query is presented in front of them. This can lead to customer dissatisfaction and result in loss. It is also the multiple messaging that can be taxing for users and deteriorate the overall experience on the website.

c. *Time-Consuming* – Chatbots are installed with the motive to speed-up the response and improve customer interaction. However, due to limited data-availability and time required for self-updating, this process appears more time-taking and expensive. Therefore, in place of attending several customers at a time, chatbots appear confused about how to communicate with people.

d. *Increased Installation Cost* – Chatbots are useful programs that help you save a lot of manpower by ensuring the all-time availability and serving to several clients at once. But unlike humans, every chatbot needs to be programmed differently for a new business which increases the initial installation cost. This also increases the time needed to prepare for the program and plan everything effectively.

e. *Zero decision-making* – Chatbots are known for being infamous because of their inability to make decisions. A similar situation has landed big companies like Microsoft etc. in trouble when their chatbot went on making a racist

rant. Therefore, it is critical to ensure proper programming of your chatbot to prevent any such incident which can hamper your brand.

f. *Poor Memory* – Chatbots are not able to memorize the past conversation which forces the user to type the same thing again & again. This can be cumbersome for the customer and annoy them because of the effort required. Thus, it is important to be careful while designing chatbots and make sure that the program is able to comprehend user queries and respond accordingly. Thus, chatbots are a thing of the future which is yet to uncover its potential but with its rising popularity and craze among companies, they are bound to stay here for long.

Sentiment Analysis:
Sentiment analysis tools can identify and analyze many pieces of text automatically and quickly.
But computer programs have problems recognizing things like sarcasm and irony, negations, jokes, and exaggerations.

The sorts of things a person would have little trouble identifying. And failing to recognize these can skew the results.

a. 'Disappointed' may be classified as a negative word for the purposes of sentiment analysis, but within the phrase "I wasn't disappointed", it should be classified as positive.
b. We would find it easy to recognize as sarcasm the statement "I'm really loving the enormous pool at my hotel!", if this statement is accompanied by a photo of a tiny swimming pool; whereas an automated sentiment analysis tool probably would not, and would most likely classify it as an example of positive sentiment.
c. With short sentences and pieces of text, for example like those you find on Twitter especially, and sometimes on Facebook, there might not be enough context for a reliable sentiment analysis.
d. Languages – Many services only operate in English, ignoring the other 6999 languages (yes – that's how many there are) spoken in the world.
e. Geographical variations – Different countries and regions us different expressions and slang, even within the same language.

So, automated sentiment analysis tools do a really great job of analyzing text for opinion and attitude, but they're not perfect.

8. APPLICATIONS

A Chatbot is a program that can have a conversation with a person using rules and Artificial Intelligence (AI) in a way that mimics human like conversations and interactions.

Chatbots have become popular in the past few years as businesses discover innovative ways to put them to use. Having a Chatbot today has numerous benefits for businesses – they make life easier for customers, are available 24/7, save time and they are easy to use.

These benefits have led to increased adoption of Chatbots by both businesses and consumers. This is great news for businesses and entrepreneurs as they can develop Chatbots to augment or improve the efficiency of any aspect of their business operations, improve customer engagement and enhance User Experience.

The following examples highlight interesting applications of Chatbots in businesses across various industries –

a. Content delivery

Media Publishers have realized that chatbots are a powerful way to engage with their audiences and monitor engagement to gain valuable insights on reader interests. Chat with the CNN and Wall Street Journal Chatbots on Facebook Messenger and receive the latest news directly in Messenger, without having to visit their websites.

b. Order Food

Various fast food giants like KFC and Pizza Hut have invested in Chatbots that enable customers to place their orders through conversations. Taco Bell went a step further to improve the conversational experience by giving their Chatbot named Taco Bell some personality. It cracks jokes, uses emojis, answers trivia questions, and will even add a cup of water to an order if the customer mentions being hungover.

c. Book Flights

Icelandair's chatbot gives their customers the ability to search for and book flights in a text-based conversational manner. Instead of drop-down menus, customers enter the information themselves. These features give customers more control over how the flight is booked and it also keeps the entire conversation in one thread so that the purchase information can be reviewed and called up with ease.

d. Companionship

Russian technology company Endurance developed its companion chatbot for Senior People and Patients with Alzheimer's Disease. The primary function of the chatbot is to be a virtual companion – To speak with senior people on general topics like the weather, nature, hobbies, movies, music, news, etc. The chatbot asks questions, reacts to the answers, is able to speak on various topics, and share

interesting news and facts from Google.

e. Market Research

The way market researchers get in touch with respondents is changing as many respondents may prefer to be contacted on WhatsApp or the Facebook messenger. Instead of having a dedicated survey app, many research firms are developing Chatbots to have personalized, engaging conversational surveys with respondents to improve the experience and increase completion rates.

f. Transportation

Uber in partnership with Facebook has enabled users to sign up for Uber and request a ride, without having to leave Messenger or download the Uber app. Ride status updates and ride receipts are delivered to a private conversation between the customers and Uber on Messenger, making it easy to track Uber rides and payment history.

g. Health Care

Chatbots have also made their way into health care by easing the burden on medical professionals by facilitating faster medical diagnosis, answering health-related questions, booking appointments and lots more. A Chatbot like Super Izzy can track menstrual cycles, dates and fertile windows.

h. E-commerce

The e-commerce industry is also improving shopping experience with Chatbots. Customers can now search and shop more conveniently with the help of chatbots. eBay has invested in Chatbot technology with their Shop Bot, a virtual shopping assistant that helps customers find the items they want at the desired price range.

i. Hospitality

Chatbots have been put to use in the hospitality industry in various ways. Radisson Blu Edwardian hotels use a virtual host called Edward to help guests with a variety of things from; online check-in, to handling customer inquiries, letting them know their rooms are ready and more.

j. Legal

Chatbots have also shown their practicality in the Legal industry as well, they are a great way of optimizing work processes to save both lawyers and their clients time and money. Montreal-based law firm EXEO launched a smart virtual assistant called IVA to help future immigrants. IVA (Immigration Virtual Assistant) makes the process of immigration much easier and less stressful.

The following examples highlight interesting applications of Sentiment Analysis in businesses across various industries –

a. Sentiment analysis has many applications and benefits to

your business and organization. It can be used to give your business valuable insights into how people feel about your product brand or service.

b. When applied to social media channels, it can be used to identify spikes in sentiment, thereby allowing you to identify potential product advocates or social media influencers.

c. It can be used to identify when potential negative threads are emerging online regarding your business, thereby allowing you to be proactive in dealing with it more quickly.

9. CONCLUSION

This application uses individual out-of-the-box UI components to extract and visualize the enriched data provided by the Discovery analytics engine. The code pattern includes Watson Discovery series. It gets the customer sentiment insights from product reviews.

10. FUTURE SCOPE

Improving result relevance with the tooling:

a. When we provide a Discovery instance with training data, the service uses machine-learning Watson techniques to find signals in your content and questions. As you add more training data, the service instance becomes more accurate and sophisticated in the ordering of results it returns.

To train Watson, you must provide the following:

a. Example queries that are representative of the queries your users enter
b. Ratings that indicate which results for each query are relevant and not relevant

To use Continuous Relevancy Training:

a. Once trained, Continuous Relevancy Training is used to influence the results of a natural language query when using an environment-level query.
b. Continuous Relevancy Training can be used at query time by running a multi-collection natural language query across all collections in your environment.

Usage monitoring:

a. You can monitor and track usage of your Discovery instance and use this data to help you understand and improve your applications. The Events API can be used to create log entries that are associated with specific natural language queries and actions. For example, you can record which documents in a results set were "clicked" by a user, and when that click occurred.

Making more complex bots:

a. In the steps above, we made a lot of assumptions of simplicity. For example, we expected that users would respond with a very specific spelling, we assumed a test environment where there were no other conversations happening, and so on.

b. For a real bot in production, some of these assumptions would break the behavior of the bot. So, let's cover some situations that you should address for your own bots - think of these as best practices rather than specific instructions to follow.

Tracking conversations:

a. In our example bot, we've used a mention as the triggering point for a specific conversation, but you'll notice that your bot will still respond if you skip some of the steps - for example if you type Who's there?, your bot will respond to this message with A bot user, even if you didn't mention the bot or start at the beginning of the conversation.

b. A solution to this might involve tracking the beginning of a conversation, the participants involved, and the progress through the flow. For example, when the user first mentions the bot, a database entry is created that identifies that user and the open workflow with them.

c. As the user progresses through the flow, the database records this, and the user is unable to repeat earlier steps in the conversation. The workflow is completed, the database entry is also marked as complete, and the bot waits for another mention before starting anew.

Threaded messages:

a. Be aware that a user might choose to reply to your bot's messages in a thread rather than at the channel-level. Your bot will still receive message events for these threaded replies, but you will have to add some extra logic to ensure that your bot responds to the user in the relevant location.

b. Check out the Threading Messages docs for more information on how to spot the difference between messages and threaded messages.

Variations in phrasing:

a. Because your bot will be interacting with humans, it's unlikely that you can expect consistent spelling and phrasing across messages from different people that might be trying to invoke the same thing. For example, our simple example bot used the phrase tell me a joke to trigger the start of the workflow, but at a very basic level a user might also try typing what's a good joke? or make me laugh.

b. Your bot can get more complex by broadening its understanding of natural language queries to capture a wider range of potential trigger phrases. Alternatively, you can be more prescriptive about the exact phrasing to use, and provide user education to train correct usage.

Integrating with other services:

a. The real magic of a bot comes when it is connected with external services, providing a seamless conversational interface for them from within Slack.

11. BIBLIOGRAPHY

a. https://www.youtube.com/watch?v=LOCkV-mENq8&feature=youtu.be
b. https://github.com/
c. https://www.youtube.com/watch?v=w3jLJU7DT5E&feature=youtu.be
d. https://www.youtube.com/watch?v=7YUTc4Cigc8&feature=youtu.be
e. https://www.ibm.com/cloud/get-started
f. https://developer.ibm.com/tutorials/how-to-create-a-node-red-starter-application/
g. https://www.w3schools.com/howto/howto_make_a_website.asp
h. https://www.youtube.com/watch?v=W3iPbFTAAds&feature=youtu.be
i. https://www.youtube.com/watch?v=W3iPbFTAAds&feature=youtu.be
j. https://discovery-news-demo.ng.bluemix.net/

APPENDIX

A. Source Code

```json
[
  {
    "id": "8522401.ff813c",
    "type": "tab",
    "label": "Flow 2",
    "disabled": false,
    "info": ""
  },
  {
    "id": "54ee22e6.be9d8c",
    "type": "ui_form",
    "z": "8522401.ff813c",
    "name": "",
    "label": "",
    "group": "dd4ec0bb.0d939",
    "order": 3,
    "width": "14",
    "height": "4",
    "options": [
      {
        "label": "Enter the input",
        "value": "Input",
        "type": "text",
        "required": true,
        "rows": null
      }
    ],
    "formValue": {
      "Input": ""
    },
    "payload": "",
    "submit": "submit",
```

```
      "cancel": "cancel",
      "topic": "",
      "x": 100,
      "y": 300,
      "wires": [
         [
            "e7e0b309.b3847"
         ]
      ]
   },
   {
      "id": "e7e0b309.b3847",
      "type": "function",
      "z": "8522401.ff813c",
      "name": "",
      "func": "msg={\n   discoveryparams:\n{\n   \"environment_id\":\"system\",\n
\"query\":msg.payload.Input\n}\n}\nreturn msg;",
      "outputs": 1,
      "noerr": 0,
      "x": 245,
      "y": 320,
      "wires": [
         [
            "5fdc6f20.9f714"
         ]
      ]
   },
   {
      "id": "5fdc6f20.9f714",
      "type": "watson-discovery-v1",
      "z": "8522401.ff813c",
      "name": "discovery",
      "environmentname": "",
      "environmentId": "system",
      "collectionId": "news-en",
      "configurationname": "",
      "configurationId": "",
```

        "language_code": "en",
        "collection_name": "",
        "count": "1",
        "passages": true,
        "nlp_query": true,
        "query": "",
        "filter": "",
        "aggregation": "",
        "return": "",
        "description": "",
        "size": "LT",
        "discovery-method": "query",
        "service-endpoint":
"https://api.eu-gb.discovery.watson.cloud.ibm.com/instances/260463e6-9432-44ba-80
e0-ef1a24051890",
        "x": 380,
        "y": 280,
        "wires": [
            [
                "42104dc3.99cbc4",
                "fc28bd4f.aa882",
                "a132def6.859ab",
                "ff28ef44.661bc",
                "b119a7de.565908",
                "b5eeabf8.099a08",
                "4f27e62c.2a0b08",
                "7df2a091.dcc7d"
            ]
        ]
    },
    {
        "id": "42104dc3.99cbc4",
        "type": "debug",
        "z": "8522401.ff813c",
        "name": "",
        "active": true,
        "tosidebar": true,

```
        "console": false,
        "tostatus": false,
        "complete": "payload",
        "targetType": "msg",
        "x": 320,
        "y": 160,
        "wires": []
    },
    {
        "id": "774faadb.7d0a24",
        "type": "debug",
        "z": "8522401.ff813c",
        "name": "",
        "active": true,
        "tosidebar": true,
        "console": false,
        "tostatus": false,
        "complete": "true",
        "targetType": "full",
        "x": 410,
        "y": 420,
        "wires": []
    },
    {
        "id": "7df2a091.dcc7d",
        "type": "function",
        "z": "8522401.ff813c",
        "name": "text",
        "func": "msg.payload=msg.search_results.results[0].text\nreturn msg;",
        "outputs": 1,
        "noerr": 0,
        "x": 560,
        "y": 40,
        "wires": [
            [
                "98ee8d88.a789e",
                "35e328df.3f4358",
```

```
            "4f27e62c.2a0b08"
        ]
    ]
},
{
    "id": "fc28bd4f.aa882",
    "type": "function",
    "z": "8522401.ff813c",
    "name": "author",
    "func": "msg.payload=msg.search_results.results[0].author\nreturn msg;",
    "outputs": 1,
    "noerr": 0,
    "x": 560,
    "y": 120,
    "wires": [
        [
            "b022b95d.b5aa78"
        ]
    ]
},
{
    "id": "a132def6.859ab",
    "type": "function",
    "z": "8522401.ff813c",
    "name": "main page url",
    "func": "msg.payload=msg.search_results.results[0].main_image_url\nreturn msg;",
    "outputs": 1,
    "noerr": 0,
    "x": 590,
    "y": 180,
    "wires": [
        [
            "1f8d1dd8.b11bc2"
        ]
    ]
},
{
```

```json
    "id": "ff28ef44.661bc",
    "type": "function",
    "z": "8522401.ff813c",
    "name": "url",
    "func": "msg.payload=msg.search_results.results[0].url\nreturn msg;",
    "outputs": 1,
    "noerr": 0,
    "x": 560,
    "y": 220,
    "wires": [
      [
        "64ac5c7f.6920c4"
      ]
    ]
  },
  {
    "id": "b119a7de.565908",
    "type": "function",
    "z": "8522401.ff813c",
    "name": "score",
    "func":
"msg.payload=msg.search_results.results[0].enriched_text.sentiment.document.score\n
return msg;",
    "outputs": 1,
    "noerr": 0,
    "x": 560,
    "y": 280,
    "wires": [
      [
        "ccd44d70.1d8e4",
        "ef5e1413.8fe598"
      ]
    ]
  },
  {
    "id": "b5eeabf8.099a08",
    "type": "function",
```

"z": "8522401.ff813c",

"name": "label",

"func":

"msg.payload=msg.search_results.results[0].enriched_text.sentiment.document.label\nr

eturn msg;",

"outputs": 1,

"noerr": 0,

"x": 570,

"y": 360,

"wires": [

[

"3f925a55.2b7016"

]

]

},

{

"id": "4f27e62c.2a0b08",

"type": "sentiment",

"z": "8522401.ff813c",

"name": "",

"property": "payload",

"x": 240,

"y": 500,

"wires": [

[

"774faadb.7d0a24",

"e6fcdb1.89e9e28",

"4874c1be.f5e2d",

"7ab47479.e4025c"

]

]

},

{

"id": "e6fcdb1.89e9e28",

"type": "function",

"z": "8522401.ff813c",

"name": "sentiment",

```
      "func": "msg.payload=msg.sentiment.score;\nreturn msg;",
      "outputs": 1,
      "noerr": 0,
      "x": 582,
      "y": 440,
      "wires": [
        [
            "5e8632ce.796eec"
        ]
      ]
  },
  {
      "id": "4874c1be.f5e2d",
      "type": "function",
      "z": "8522401.ff813c",
      "name": "positive",
      "func": "msg.payload=msg.sentiment.positive.length;\nreturn msg;",
      "outputs": 1,
      "noerr": 0,
      "x": 580,
      "y": 500,
      "wires": [
        [
            "47837ed5.d7609"
        ]
      ]
  },
  {
      "id": "7ab47479.e4025c",
      "type": "function",
      "z": "8522401.ff813c",
      "name": "negative",
      "func": "msg.payload=msg.sentiment.negative.length;\nreturn msg;",
      "outputs": 1,
      "noerr": 0,
      "x": 580,
      "y": 560,
```

```
    "wires": [
        [
            "7e2e056f.6b062c"
        ]
    ]
},
{
    "id": "5e8632ce.796eec",
    "type": "ui_chart",
    "z": "8522401.ff813c",
    "name": "",
    "group": "dd4ec0bb.0d939",
    "order": 11,
    "width": "3",
    "height": "5",
    "label": "sentiment score",
    "chartType": "bar",
    "legend": "false",
    "xformat": "HH:mm:ss",
    "interpolate": "linear",
    "nodata": "",
    "dot": false,
    "ymin": "",
    "ymax": "",
    "removeOlder": 1,
    "removeOlderPoints": "",
    "removeOlderUnit": "3600",
    "cutout": 0,
    "useOneColor": true,
    "useUTC": false,
    "colors": [
        "#ff0000",
        "#aec7e8",
        "#ff7f0e",
        "#2ca02c",
        "#98df8a",
        "#d62728",
```

```
            "#ff9896",
            "#9467bd",
            "#c5b0d5"
        ],
        "useOldStyle": false,
        "outputs": 1,
        "x": 780,
        "y": 440,
        "wires": [
            []
        ]
    },
    {
        "id": "47837ed5.d7609",
        "type": "ui_chart",
        "z": "8522401.ff813c",
        "name": "",
        "group": "dd4ec0bb.0d939",
        "order": 12,
        "width": "3",
        "height": "5",
        "label": "positive chart",
        "chartType": "bar",
        "legend": "false",
        "xformat": "HH:mm:ss",
        "interpolate": "linear",
        "nodata": "",
        "dot": false,
        "ymin": "",
        "ymax": "",
        "removeOlder": 1,
        "removeOlderPoints": "",
        "removeOlderUnit": "3600",
        "cutout": 0,
        "useOneColor": true,
        "useUTC": false,
        "colors": [
```

```
            "#1f77b4",
            "#aec7e8",
            "#ff7f0e",
            "#2ca02c",
            "#98df8a",
            "#d62728",
            "#ff9896",
            "#9467bd",
            "#c5b0d5"
        ],
        "useOldStyle": false,
        "outputs": 1,
        "x": 770,
        "y": 500,
        "wires": [
            []
        ]
    },
    {
        "id": "7e2e056f.6b062c",
        "type": "ui_chart",
        "z": "8522401.ff813c",
        "name": "",
        "group": "dd4ec0bb.0d939",
        "order": 13,
        "width": "3",
        "height": "5",
        "label": "negative chart",
        "chartType": "bar",
        "legend": "false",
        "xformat": "HH:mm:ss",
        "interpolate": "linear",
        "nodata": "",
        "dot": false,
        "ymin": "",
        "ymax": "",
        "removeOlder": 1,
```

        "removeOlderPoints": "",
        "removeOlderUnit": "3600",
        "cutout": 0,
        "useOneColor": true,
        "useUTC": false,
        "colors": [
            "#1f77b4",
            "#aec7e8",
            "#ff7f0e",
            "#2ca02c",
            "#98df8a",
            "#d62728",
            "#ff9896",
            "#9467bd",
            "#c5b0d5"
        ],
        "useOldStyle": false,
        "outputs": 1,
        "x": 780,
        "y": 560,
        "wires": [
            []
        ]
    },
    {
        "id": "98ee8d88.a789e",
        "type": "ui_text",
        "z": "8522401.ff813c",
        "group": "dd4ec0bb.0d939",
        "order": 4,
        "width": "14",
        "height": "2",
        "name": "",
        "label": "search result",
        "format": "{{msg.payload}}",
        "layout": "row-spread",
        "x": 740,

```
            "y": 40,
            "wires": []
        },
        {
            "id": "b022b95d.b5aa78",
            "type": "ui_text",
            "z": "8522401.ff813c",
            "group": "dd4ec0bb.0d939",
            "order": 5,
            "width": "14",
            "height": "2",
            "name": "",
            "label": "author",
            "format": "{{msg.payload}}",
            "layout": "row-spread",
            "x": 730,
            "y": 120,
            "wires": []
        },
        {
            "id": "1f8d1dd8.b11bc2",
            "type": "ui_text",
            "z": "8522401.ff813c",
            "group": "dd4ec0bb.0d939",
            "order": 6,
            "width": "14",
            "height": "2",
            "name": "",
            "label": "main page url",
            "format": "{{msg.payload}}",
            "layout": "row-spread",
            "x": 780,
            "y": 180,
            "wires": []
        },
        {
            "id": "64ac5c7f.6920c4",
```

```json
        "type": "ui_text",
        "z": "8522401.ff813c",
        "group": "dd4ec0bb.0d939",
        "order": 7,
        "width": "14",
        "height": "2",
        "name": "",
        "label": "url",
        "format": "{{msg.payload}}",
        "layout": "row-spread",
        "x": 730,
        "y": 220,
        "wires": []
    },
    {
        "id": "ccd44d70.1d8e4",
        "type": "ui_text",
        "z": "8522401.ff813c",
        "group": "dd4ec0bb.0d939",
        "order": 8,
        "width": "14",
        "height": "2",
        "name": "",
        "label": "score",
        "format": "{{msg.payload}}",
        "layout": "row-spread",
        "x": 730,
        "y": 280,
        "wires": []
    },
    {
        "id": "3f925a55.2b7016",
        "type": "ui_text",
        "z": "8522401.ff813c",
        "group": "dd4ec0bb.0d939",
        "order": 10,
        "width": "14",
```

```json
        "height": "2",
        "name": "",
        "label": "label",
        "format": "{{msg.payload}}",
        "layout": "row-spread",
        "x": 730,
        "y": 360,
        "wires": []
    },
    {
        "id": "ef5e1413.8fe598",
        "type": "ui_gauge",
        "z": "8522401.ff813c",
        "name": "",
        "group": "dd4ec0bb.0d939",
        "order": 9,
        "width": 0,
        "height": 0,
        "gtype": "gage",
        "title": "sentiment score",
        "label": "units",
        "format": "{{value}}",
        "min": "-2",
        "max": "2",
        "colors": [
            "#00b500",
            "#ffff00",
            "#ff0000"
        ],
        "seg1": "",
        "seg2": "",
        "x": 890,
        "y": 320,
        "wires": []
    },
    {
        "id": "35e328df.3f4358",
```

        "type": "debug",
        "z": "8522401.ff813c",
        "name": "",
        "active": true,
        "tosidebar": true,
        "console": false,
        "tostatus": false,
        "complete": "true",
        "targetType": "full",
        "x": 860,
        "y": 80,
        "wires": []
    },
    {
        "id": "dd4ec0bb.0d939",
        "type": "ui_group",
        "z": "",
        "name": "IBM",
        "tab": "368ea61e.342a8a",
        "order": 1,
        "disp": true,
        "width": "14",
        "collapse": false
    },
    {
        "id": "368ea61e.342a8a",
        "type": "ui_tab",
        "z": "",
        "name": "IBM",
        "icon": "dashboard",
        "disabled": false,
        "hidden": false
    }
]

--------------------------------------xxx-------------------------------------