1. INTRODUCTION

1.1 Overview

Keyword extraction (also known as keyword detection or keyword analysis) is a text analysis technique that automatically extracts the most used and most important words and expressions from a text. Text analysis (TA) is a machine learning technique used to automatically extract valuable insights from unstructured test data. Companies use text analysis tools to quickly digest online data and documents, and transform them into actionable insights. We can use text analysis to extract specific information, like keywords, names or company information from thousands of emails, or categorize survey responses by sentiment and topic. Text analysis delivers qualitative results. If a machine performs text analysis, it identifies important information within the text itself. Basically, the challenge in text analysis is decoding the ambiguity of human language. Keyword extraction uses machine learning, artificial intelligence (AI) with natural language processing (NLP) to break down human languages so that it can be understood and analyzed by machines. It's used to find keywords from all manner of text: regular documents and business reports, social media comments, online forums and reviews, news reports and more.

1.2 Purpose

The main purpose is to describe the content structure and functions of the messages contained in small texts and to analyze SMS messages with Watson Knowledge Studio (WKS) and Watson Language Understanding (NLU) capability to extract entities in the data.

2. LITERATURE SURVEY

2.1 Existing Problem

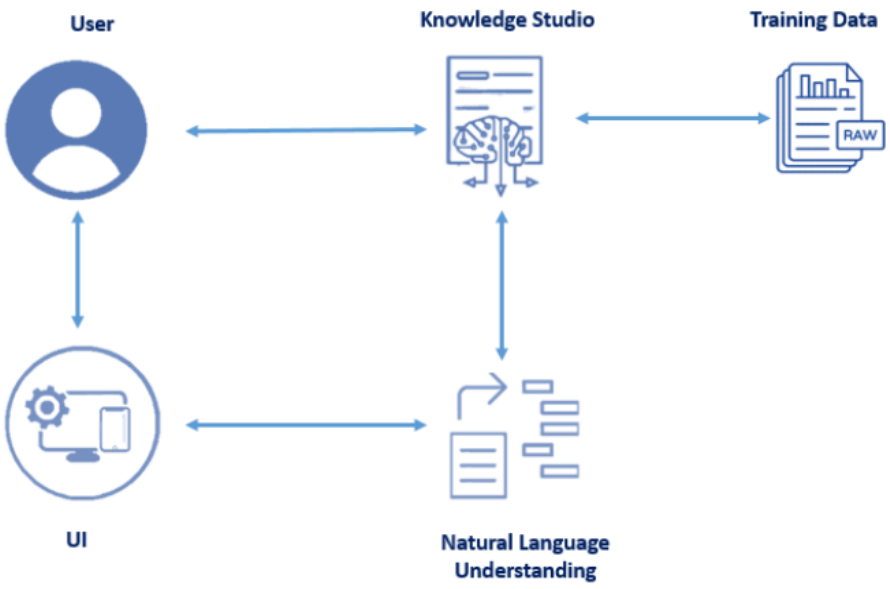
- Removing punctuations
- Transforming to lower case
- Grammatically tagging sentences and removing pre-identified stop phrases
- Removing generic words of English languages
- Stripping any excess white spaces

2.2 Proposed Solution

- We are making Web application that is based on the present mood of the user
- This application will provide a rapid, automated response to the customer, dramatically reducing their reliance on call center operators to solve problems.
- It can address both by analyzing large volumes of unstructured data, extracting opinions and their relations with brands and products.

3 THEORITICAL ANALYSIS

3.1 Block Diagram



3.2 Hardware and Software Designing

Hardware Requirements:

- Processor : Intel Core i3
- Processor Speed : 1.7 GHz
- RAM : 4 GB
- System Type : 64-Bit Operating System
- Monitor Resolution : 1280*800

Software Requirements:

- Software : Node-Red
- Operating system :windows 10
- Front end :HTML

4.EXPERIMENTAL INVESTIGATIONS:

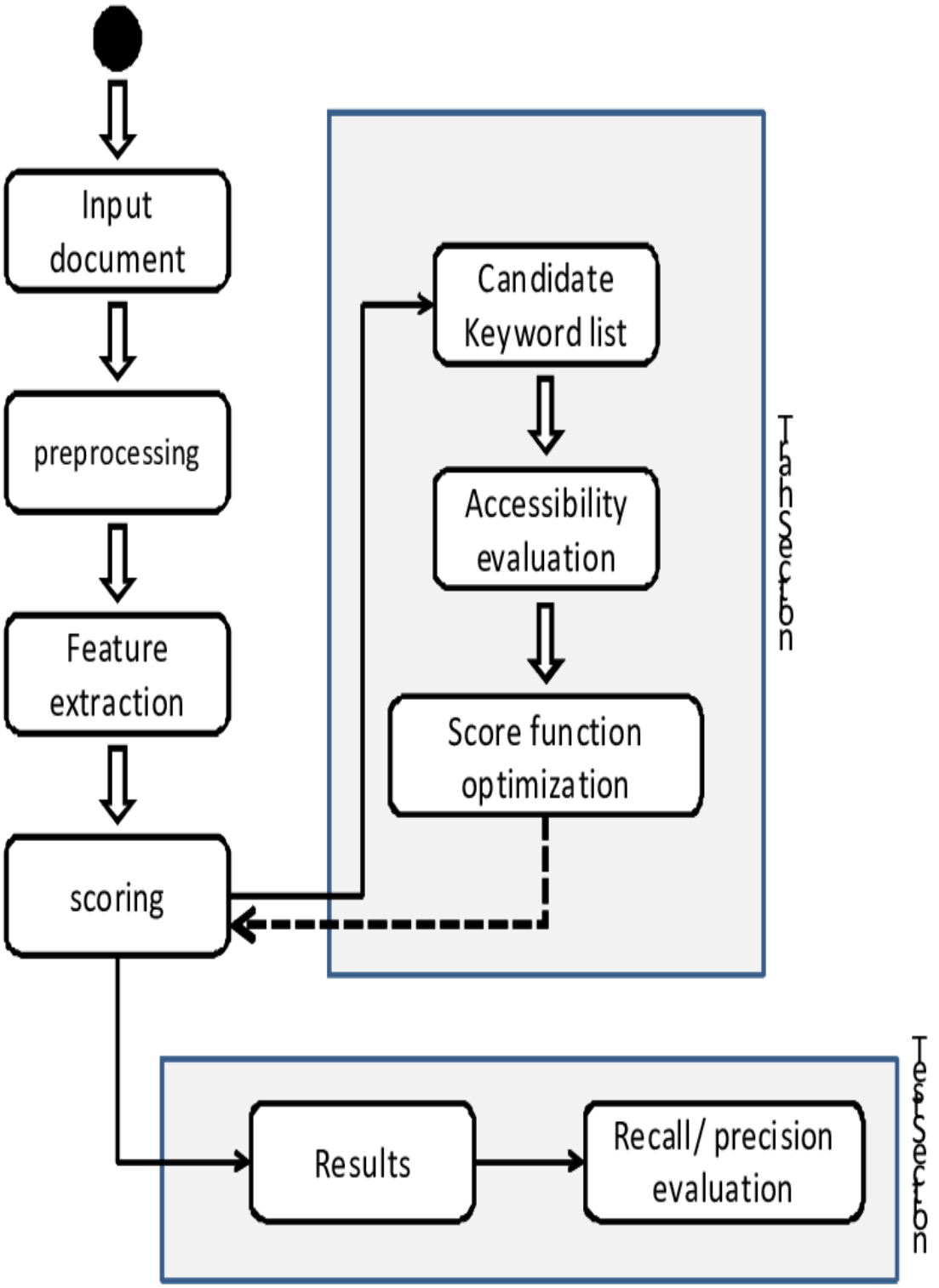
Teach IBM Watson the language of your domain with custom models that identify entities and relationships unique to your industry in unstructured text. Build your models in a collaborative environment designed for both developers and domain experts, without needing to write code. Use the models in IBM Watson Discovery, IBM Watson Natural Language Understanding and IBM Watson Explorer.

Natural language processing(NLP):

IBM Watson Natural Language Understanding (NLU) is an AI service on the IBM Cloud that enables user to analyze text to extract metadata from content such as concepts, entities, keywords, categories, sentiment, emotion, relations and semantic roles. Natural language processing (NLP) refers to the branch of computer science—and more specifically, the branch of artificial intelligence or AI—concerned with giving computers the ability to understand text and spoken words in much the same way human beings can. NLP combines computational linguistics—rule-based modeling of human language—with statistical, machine learning, and deep learning models. Together, these technologies enable computers to process human language in the form of text or voice data and to ‘understand’ its full meaning, complete with the speaker or writer’s intent and sentiment.

Node-Red: A flow-based programming tool for wiring together hardware devices, APIs, and online services. Create event-based apps with simple flow-based programming. Node-RED allows you to create functionality by wiring together flows of data between nodes using a browser. And it has gained tremendous popularity in the IoT space, by modeling bits of application functionality between IoT devices like sensors, cameras, and wireless routers.

5. FLOWCHART:


Input document
preprocessing
Feature extraction
scoring
Candidate Keyword list
Accessibility evaluation
Score function optimization
Results
Recall/ precision evaluation

6. RESULT:

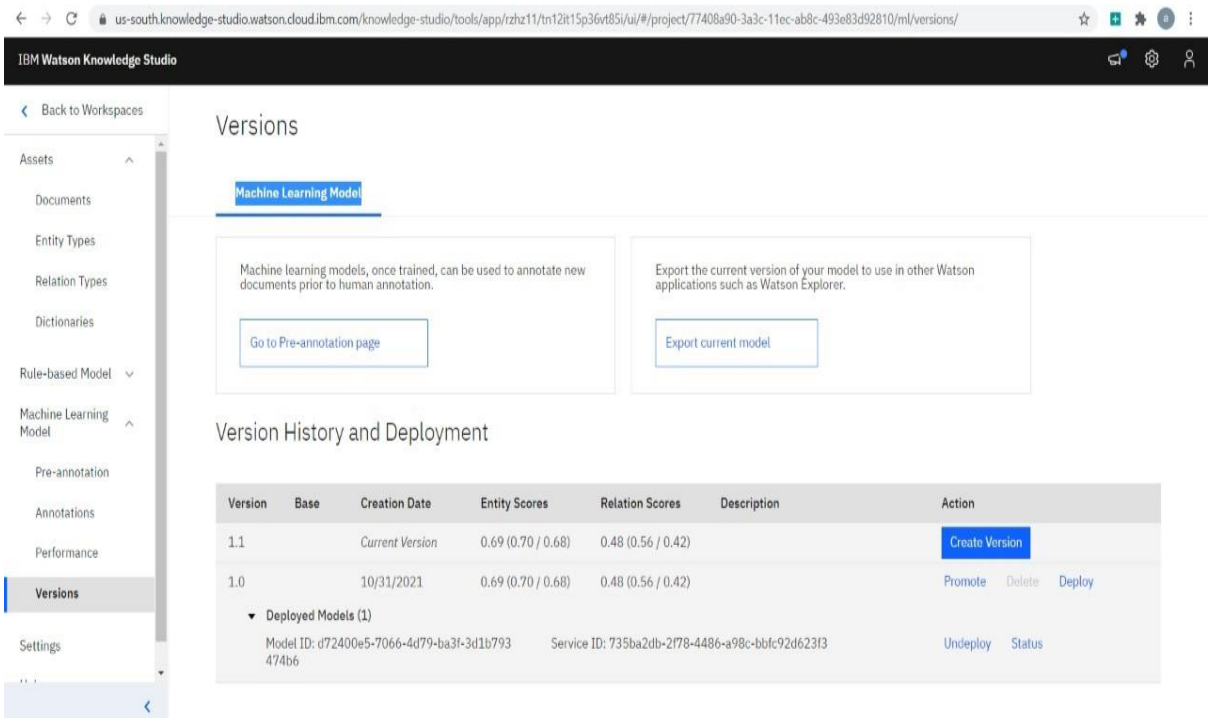


Fig 6.1: Deployed model ID

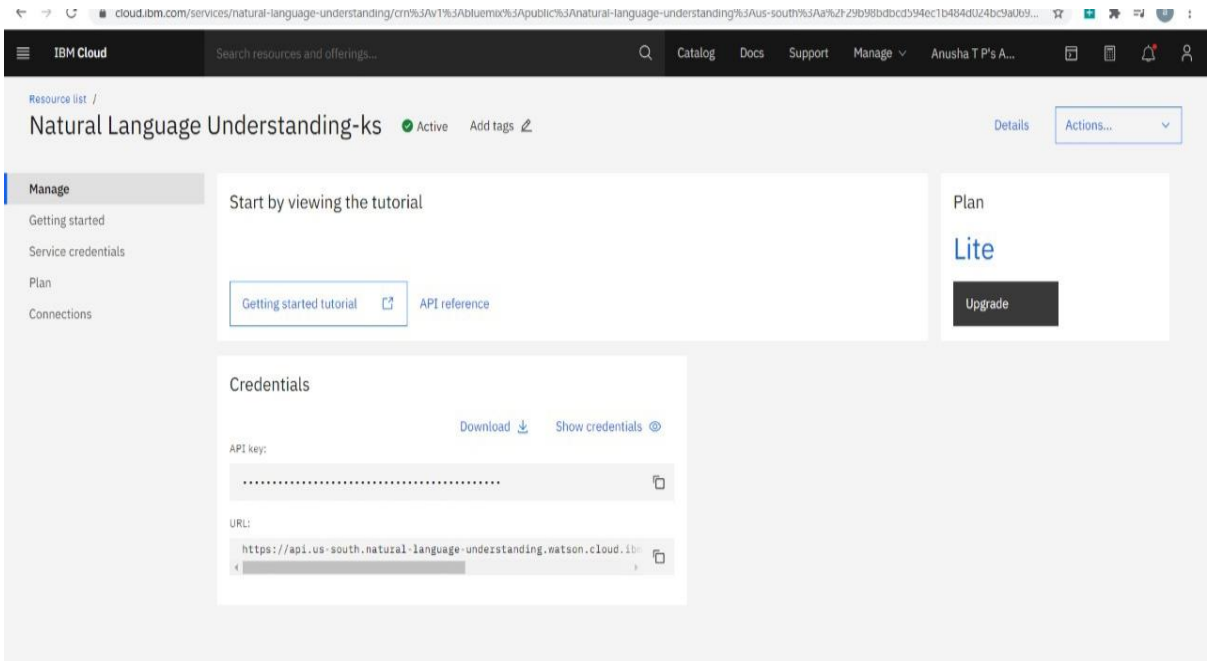


Fig 6.2: Natural language understanding service launch page

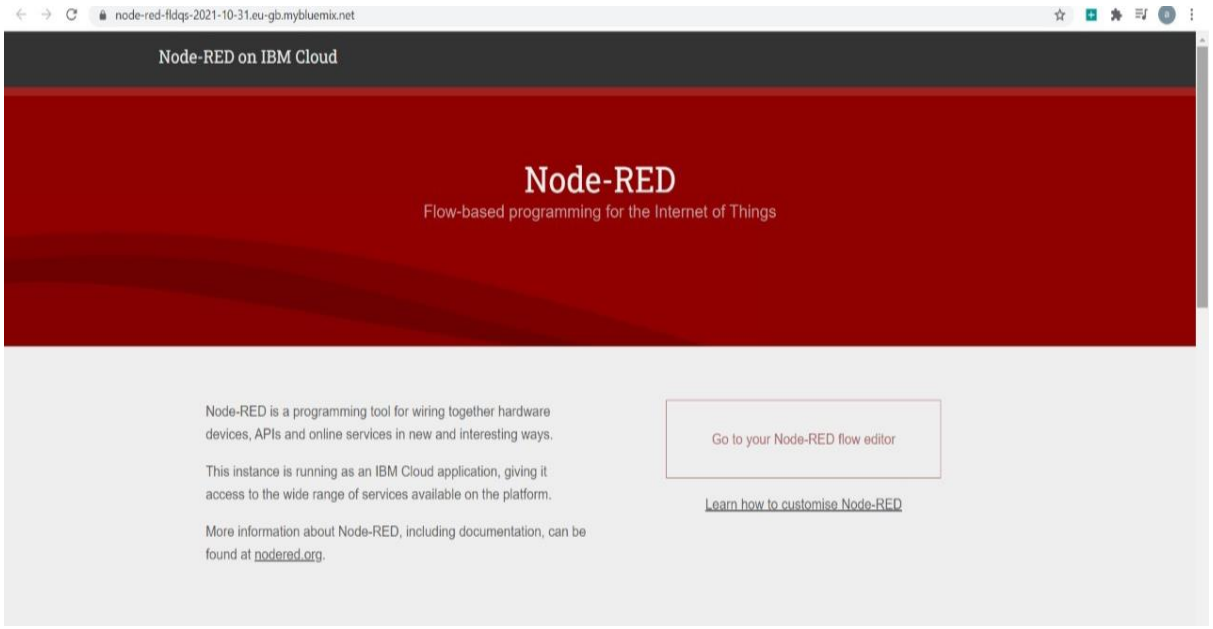


Fig 6.3: Node-Red software

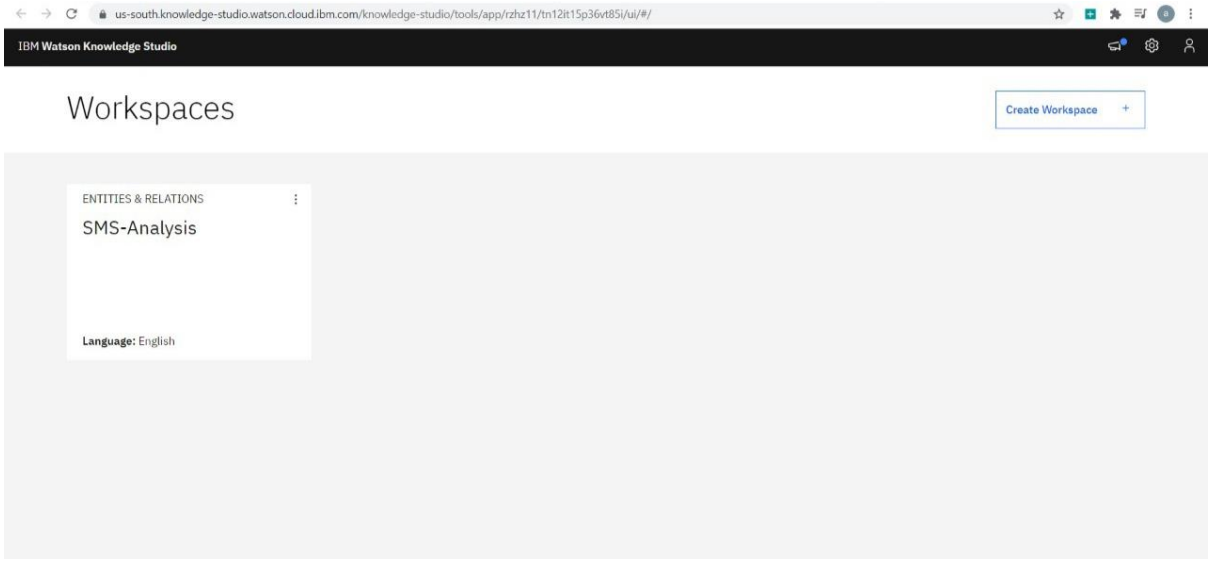


Fig 6.4: Work space of SMS analysis

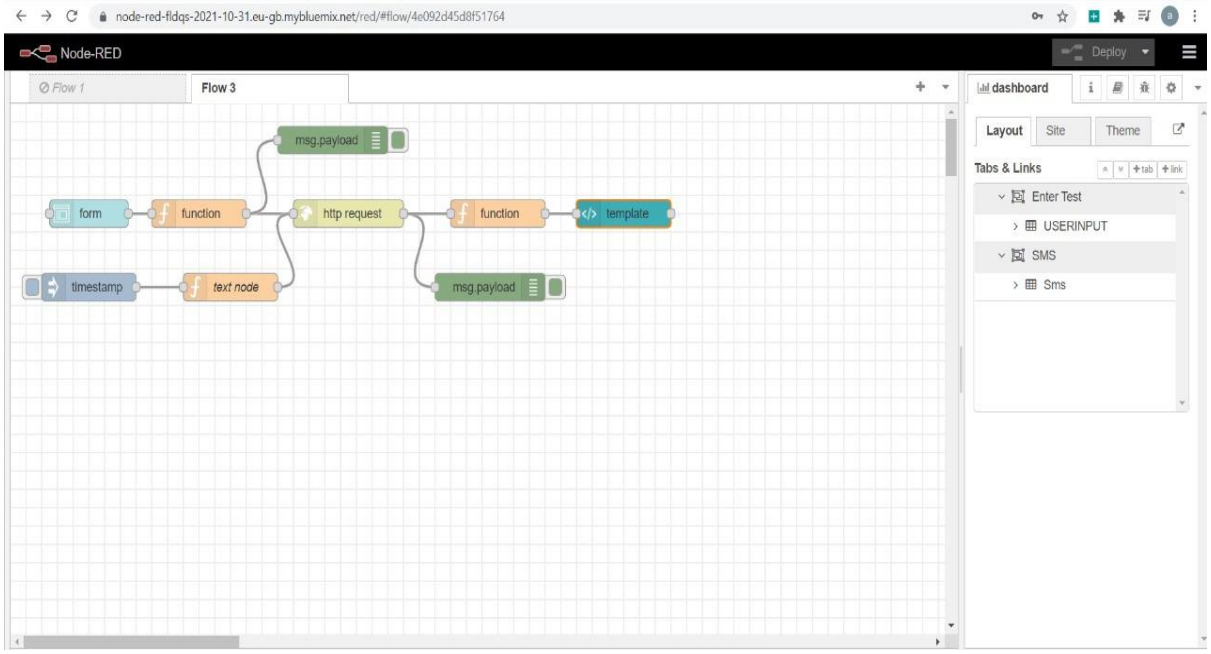


Fig 6.5: Node-Red Flows

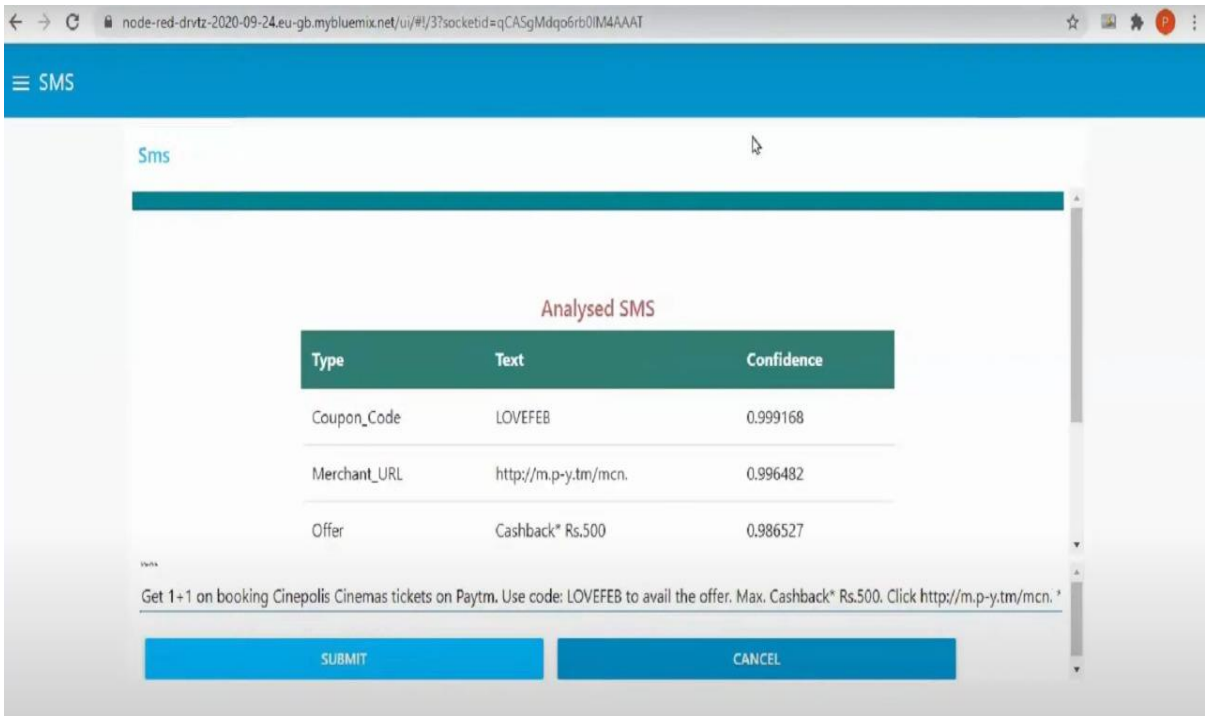


Fig 6.6: Output

7. Advantages and Disadvatages:

7.1 Advantages:

* With keyword extraction you can find the most important words and phrases in massive datasets in just seconds.

* these words and phrases can provide valuable insights into topics your customers are talking about.

* These insights can help you shape a data-driven business strategy by identifying what customers consider important, the aspects of your product that need to be improved, and what customers are saying about your competition, among others.

* In the academic world, keyword extraction may be the key to finding relevant keywords within massive sets of data (like new articles, papers, or journals) without having to actually read the entire content.

* Whatever your field of business, keyword extraction tools are the key to help you automatically index data, summarize a text, or generate tag clouds with the most representative keywords.

* Keyword extraction can automate workflows, like tagging incoming survey responses or responding to urgent customer queries, allowing you to save huge amounts of time. It also provides actionable, data-driven insights to help make better business decisions. But the best thing about keyword extraction models is that they are easy to set up and implement.

7.2 Disadvantages:

Considering that more than 80% of the data we generate every day is unstructured — meaning it's not organized in a predefined way, making it extremely difficult to analyze and process – businesses need automated keyword extraction to help them process and analyze customer data in a more efficient manner.

8.APPLICATIONS:

* Every day, internet users create 2.5 quintillion bytes of data. Social media comments, product reviews, emails, blog posts, search queries, chats, and so on. We have all sorts of unstructured text data at our disposal.

* Keyword extraction can help you obtain the most important keywords or key phrases from a given text without having to actually read a single line.

* Whether you are a product manager trying to analyze a pile of product reviews, a customer service manager analyzing customer interactions, or a researcher that has to go through hundreds of online papers about a specific topic, you can put keyword extraction to use to easily understand what a text is about.

* Thanks to keyword extraction, teams can be more efficient and take full advantage of the power of data. You can say goodbye to manual and repetitive tasks (saving countless human hours) and get access to interesting insights that will help you transform unstructured data into valuable knowledge.

Here are some common use cases and applications:

1. Social media monitoring
2. Brand monitoring
3. Customer service
4. Customer feedback
5. Business intelligence
6. Search engine optimization (SEO)
7. Product analytics
8. Knowledge management

9.CONCLUSION:

Keyword extraction simplifies the task of finding relevant words and phrases within unstructured text. This includes emails, social media posts, chat conversations, and any other types of data that are not organized in any predefined way. Keyword extraction can automate workflows, like tagging incoming survey responses or responding to urgent customer queries, allowing you to save huge amounts of time. It also provides actionable, data-driven insights to help make better business decisions. But the best thing about keyword extraction models is that they are easy to set up and implement.

10. FUTURE WORKS:

Key word extraction is an important technique for text mining, web retrieval, text classification, and so on.With the rapid development of the mobile Internet and the accelerated evolution of the Internet, the explosive growth of text data requires more advanced key word extraction techniques. The early key word extractionresearch focused on statistics, and the key words were sorted by calculating the cooccurrence frequency of words.Matsuo presented a key word extraction algorithm that uses the word cooccurrence statistical information from a single document; the main advantages of this method are that it is simple and does not require a corpus.

11.BIBILIOGRAPHY:

[1]Rada M, Paul T. TextRank: bringing order into texts. In: 2004 Conference on Empirical Methods in Natural
Language Processing; 25-26 July 2004; Barcelona, Spain. pp. 404-411.
[2] Page L, Brin S, Motwani R, Winograd T. The Pagerank Citation Ranking: Bringing Order to the Web. San
Francisco, USA: Stanford InfoLab Press, 1999.
[3] Mikolov T, Sutskever I, Chen K, Corrago G, Dean J. Distributed representations of words and phrases and their
compositionality. In: 27th Conference on Neural Information Processing Systems; 5-10 December 2013; Lake Tahoe,
Nevada, USA. pp. 1-9.

APPENDIX:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>SMS analysis</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
   <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
  <script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"></script>
  <style>
  .bg-light {
    background-color: #00838F!important;
}
h4 {
    color: #fff;
}
.custom-file {
margin-bottom: 14px;
}
.table .thead-dark th {
color: #fff;
background-color: #307d76;
border-color: #307d76;
}
h5{
    text-align: center;
    color: #bd6666;
}
.text-center {
  text-align: center;
}
.imgdiv
{
align:center;
}
</style>
</head>
<body>
<nav class="navbar navbar-expand-sm bg-light">
<div class="justify-content-center">
</div>
</nav>
<br><br>
    <div class="container">
<div class="row">
```

```
<div class="col-sm-2">
</div>
<div class="col-sm-8">
<br>
<!--<md-button ng-click="send({payload:action()})">
Predict
</md-button>-->
</div>
<div class="col-sm-2">
</div>
</div>
<div class="row">
<div class="col-sm-2">
</div>
<div class="col-sm-8">
<h5> Analysed SMS  </h5>
<table class="table">
<thead class="thead-dark">
  <tr>
<th>Type</th>
<th>Text</th>
<th>Confidence</th>
  </tr>
</thead>
<tbody id="scoretable">
</tbody>
  </table>

 </div>
<div class="col-sm-2">
</div>
</div>
</body>
</html>
<script>
var x="";
    function readURL(input) {
        if (input.files && input.files[0]) {
            var reader = new FileReader();

            reader.onload = function (e) {
                $('#blah')
                    .attr('src', e.target.result)
                    .width(150)
                    .height(200);
            };

            reader.readAsDataURL(input.files[0]);
            x= input.files[0]
```

```
            }
    }
    function getdata(data)
{
 var html = '';
            if(data != 0)
{
$.each(data, function(i){
var row = data[i];
console.log(row);
html += '<tr>';
html += '<td>';
html +=  row.type;
html += '<td>';
html +=  row.text;
html += '</td>';
html += '<td>';
html +=  row.confidence;
html += '</td>';
html += '</tr>';
});
}
else
html+="<div>No Data</div>";
$('#scoretable').html(html);
}
    (function(scope) {
        scope.$watch('msg.payload', function(data) {
            console.log('Position 2');
            console.dir(data);
            getdata(data);

        });
    })(scope);
 // or overwrite value in your callback function ...
this.scope.action = function() { return x; }

</script>
```