

System Integration Final Project Documentation

Name: Chiranjeevi Venkata Siva kumar Gorantla

Id: 801259516

API: Language Service - Entity Recognition

1. Named Entity Recognition:

Named Entity Recognition (NER) is one of the aspects of Azure Cognitive Service for Language for building intelligent apps that use written language. The NER feature is used to recognize and categorize items in unstructured text. For example, People, locations, organizations, and amounts.

The API can be found here - [Named Entity Recognition](#)

- Swagger Link:-

API Endpoint: /entity-recog

Method: POST

Body Param: JSON Format

```
{  
    "inputText": [  
        "Microsoft was founded by Bill Gates and Paul Allen on April 4, 1975, to develop and  
        sell BASIC interpreters for the Altair 8800",  
        "La sede principal de Microsoft se encuentra en la ciudad de Redmond, a 21  
        kilómetros de Seattle.",  
        "Hello world I am Chitti the robot",  
        "Apple headquarters is in",  
        "Charlotte is a queens city"  
    ]  
}
```

Request:

Named Entity Recognition

POST /entity-recog

The NER feature can identify and categorize entities in unstructured text.

Parameters

Name	Description
body * required	Input (body) Edit Value Model

```
{
  "inputText": [
    "Microsoft was founded by Bill Gates and Paul Allen on April 4, 1975, to develop and sell BASIC interpreters for the Altair 8800",
    "La sede principal de Microsoft se encuentra en la ciudad de Redmond, a 21 kilómetros de Seattle.",
    "Hello world I am Chitti the robot",
    "Apple headquarters is in",
    "Charlotte is a queens city"
  ]
}
```

Cancel

Parameter content type application/json

Response:

Responses Response content type application/json

Curl

```
curl -X 'POST' \
'http://localhost:3000/entity-recog' \
-H 'accept: application/json' \
-H 'Content-Type: application/json' \
-d '{
  "inputText": [
    "Microsoft was founded by Bill Gates and Paul Allen on April 4, 1975, to develop and sell BASIC interpreters for the Altair 8800",
    "La sede principal de Microsoft se encuentra en la ciudad de Redmond, a 21 kilómetros de Seattle.",
    "Hello world I am Chitti the robot",
    "Apple headquarters is in",
    "Charlotte is a queens city"
  ]
}'
```

Request URL

http://localhost:3000/entity-recog

Server response

Code	Details
200	Response body

```
[
  {
    "id": "0",
    "warnings": [],
    "entities": [
      {
        "text": "Microsoft",
        "category": "Organization",
        "offset": 0,
        "length": 9,
        "confidenceScore": 0.99
      },
      {
        "text": "Bill Gates",
        "category": "Person",
        "offset": 25,
        "length": 10,
        "confidenceScore": 1
      },
      {
        "text": "Paul Allen",
        "category": "Person",
        "offset": 40,
        "length": 10,
        "confidenceScore": 1
      }
    ]
  }
]
```

Download

2. Language Detection: Detects the language a document is written in, and returns a language code for a wide range of languages.

API Endpoint: /language-detect

Method: POST

Body Param: JSON Format

```
{  
  "langText": [  
    "Ce document est rédigé en Français."  
  ]  
}
```

Request:

Language Detection

POST /language-detect

This can detect the language a document is written in, and returns a language code for a wide range of languages.

Parameters

Name Description

body • required
array[string] Input
(body) Example Value | Model

```
{  
  "langText": [  
    "Ce document est rédigé en Français."  
  ]  
}
```

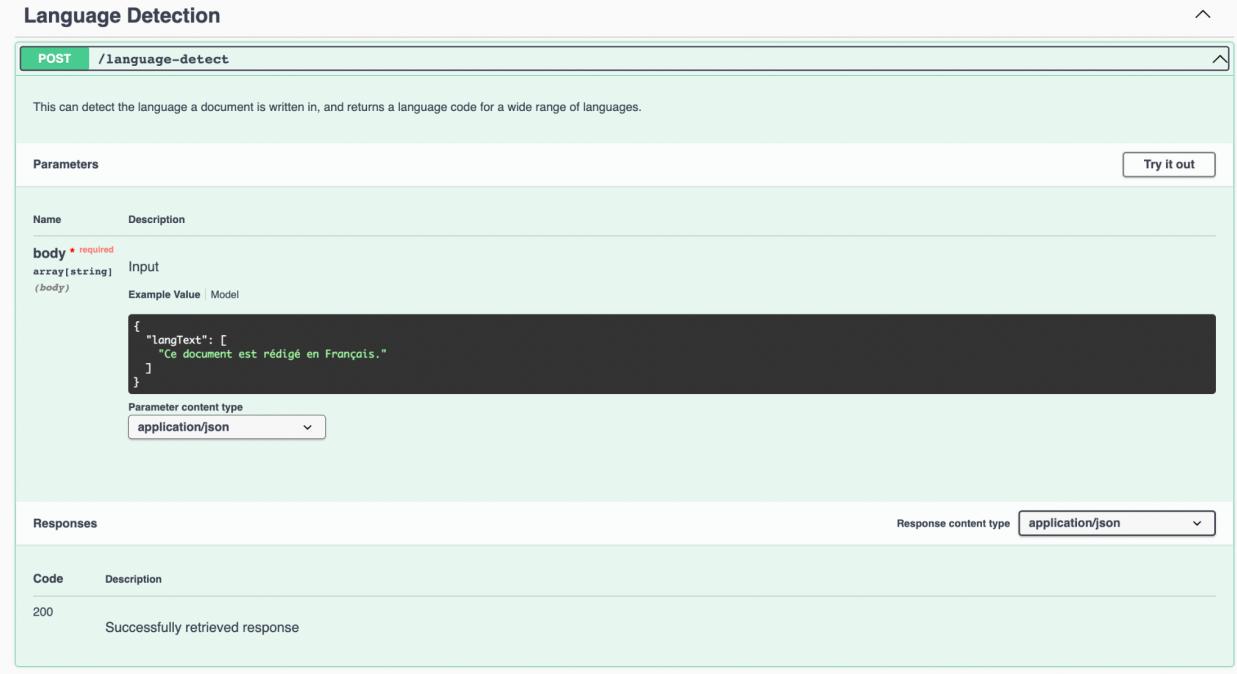
Parameter content type application/json

Responses

Code Description

200 Successfully retrieved response

Response content type application/json



Response:

Responses

Response content type application/json

Curl

```
curl -X 'POST' \
'http://localhost:3000/language-detect' \
-H 'accept: application/json' \
-H 'Content-Type: application/json' \
-d '{
  "langText": [
    "Ce document est rédigé en Français."
  ]
}'
```

Request URL

<http://localhost:3000/language-detect>

Server response

Code	Details
200	<p>Response body</p> <pre>[{ "id": "0", "warnings": [], "primaryLanguage": { "name": "French", "iso6391Name": "fr", "confidenceScore": 1 } }]</pre> <p>Download</p> <p>Response headers</p> <pre>access-control-allow-origin: * connection: keep-alive content-length: 101 content-type: application/json; charset=utf-8 date: Sun, 01 May 2012 22:35:13 GMT etag: W/"65-2Je8P+7a+d81sz50t+QGDzsvlc8" keep-alive: timeout=5 x-powered-by: Express</pre>

Responses

Code	Description
200	Successfully retrieved response

3. Key Phrase Extraction: Use key phrase extraction to quickly identify the main concepts in text.

API Endpoint: /key-phrase

Method: POST

Body Param: JSON Format

```
{
  "phraseText": [
    "My cat might need to see a veterinarian."
  ]
}
```

Request:

Key Phrase Extraction

POST /key-phrase

Use key phrase extraction to quickly identify the main concepts in text.

Parameters

Name	Description
body <small>required</small>	Input (body)

Example Value Model

```
{
  "phraseText": [
    "My cat might need to see a veterinarian."
  ]
}
```

Parameter content type application/json

Try it out

Response:

Responses

Curl

```
curl -X 'POST' \
'http://localhost:3000/key-phrase' \
-H 'accept: application/json' \
-H 'Content-Type: application/json' \
-d '{
  "phraseText": [
    "My cat might need to see a veterinarian."
  ]
}'
```

Request URL

http://localhost:3000/key-phrase

Server response

Code Details

200 Response body

```
[
  {
    "id": "0",
    "warnings": [],
    "keyPhrases": [
      "cat",
      "veterinarian"
    ]
  }
]
```

Response content type application/json

Download

4. Personally Identifiable Information (PII) detection: The PII detection feature can identify, categorize, and redact sensitive information in unstructured text.

API Endpoint: /Per-Iden

Method: POST

Body Param: JSON Format

```
{
  "perText": [
    "The employee's phone number is (555) 555-5555."
  ]
}
```

Request:

Personally Identifiable Information (PII) detection

POST /Per-Iden

The PII detection feature can identify, categorize, and redact sensitive information in unstructured text.

Parameters

Name Description

body • required
array(string) Input
(body)
Example Value | Model

```
{  
  "perText": [  
    "The employee's phone number is (555) 555-5555."  
  ]  
}
```

Parameter content type application/json

Try it out

Response:

Responses

curl -X 'POST' \
 'http://localhost:3000/Per-Iden' \
 -H 'accept: application/json' \
 -H 'Content-Type: application/json' \
 -d '{
 "perText": [
 "The employee``s phone number is (555) 555-5555."
]
}'

Request URL
http://localhost:3000/Per-Iden

Server response

Code Details

200 Response body

```
[  
  {  
    "id": "0",  
    "warnings": [],  
    "entities": [  
      {  
        "text": "employee",  
        "category": "PersonType",  
        "offset": 4,  
        "length": 8,  
        "confidenceScore": 0.96  
      },  
      {  
        "text": "(555) 555-5555",  
        "category": "PhoneNumber",  
        "offset": 31,  
        "length": 14,  
        "confidenceScore": 0.8  
      }  
    ],  
    "redactedText": "The *****`s phone number is *****."  
  }]
```

Download

Tools, Languages and Frameworks Used:

Node js - To implement the API

Digital Ocean - To deploy the project

Npm - Dependencies in node js application

Swagger - API documentation and testing

Git - Code evaluation

References:

Azure Documentation -

<https://azure.microsoft.com/en-us/services/cognitive-services/language-service/>

Source Code - <https://github.com/chiranjeevi-gorantla/SystemIntegration-Final-Project>