

# PROYECTO INTEGRADOR FINAL IA + SE CHATBOT EL SABROSITO

Universidad Politécnica Salesiana  
Jorge Arévalo, Gustavo Gualpa

[jarevalop1@est.ups.edu.ec](mailto:jarevalop1@est.ups.edu.ec)

[fgualpa@est.ups.edu.ec](mailto:fgualpa@est.ups.edu.ec)



## RESUMEN:

*Los métodos de búsqueda son muy importantes en nuestro mundo actúa ya que ellos nos permiten en base a ciertos criterios poder recomendar soluciones acordes a las necesidades del usuario mediante el desarrollo del presente documento se hablará y se implementará un **chatbot** como un sistema informático que permite mantener un diálogo con un humano.*

**PALABRAS CLAVE:** *Inteligencia Artificial, Chatbot, Neo4j, Algoritmos, Grafos, Python, etc.*

## ABSTRACT:

Search methods are very important in our world they operate since they allow us, based on certain criteria, to be able to recommend solutions according to the user's needs. Through the development of this document, a chatbot is spoken and implemented as a computer system that allows us to maintain A dialogue with a human.

**KEY WORDS:** *Artificial Intelligence, Neo4j, Java, Chatbot, Algorithms, Graphs, Python, etc.*

## 1 INTRODUCCION

Para el desarrollo de nuestro sistema de recomendación partimos desde la base que es el conocimiento de cada uno de los algoritmos que previamente seleccionamos a continuación se detalla de manera general cada uno de ellos a fin de poder de una mejor manera el presente trabajo.

Al momento de nuestra elección sobre que algoritmos vamos a utilizar depende en gran medida del caso de uso que le vayamos a dar, para ello puede influir elementos como:

### SECCION 1 “ALGORITMO DE CENTRALIDAD”

## CERCANÍA CENTRALIDAD

El algoritmo calcula la suma de sus distancias a todos los demás nodos, la suma resultante se invierte y determina la puntuación de proximidad para ese nodo. Es importante destacar que, a pesar de ser unos algoritmos altamente funcionales, estos funcionan de forma óptima en grafos conectados, es decir, que su aplicación en grafos no conectados puede arrojarnos algunos errores de análisis. Pudiésemos obtener una distancia infinita entre dos nodos en componentes conectados por separado. Al ocurrir esto se estaría otorgando una puntuación de centralidad de cercanía infinita cuando hagamos la sumatoria de las distancias de un vértice determinado.

### SECCION 2 “ALGORITMO DE DETENCION DE LA COMUNIDAD”

#### LOUVAIN

Los algoritmos de detección de comunidades cumplen con una función de gran importancia para el análisis de datos en grafos que contienen información compleja, Estos tomando en cuenta sus características, formas de relación y diferentes tipos de conexiones nos ayudan a establecer o determinar la existencia de comunidades complejas de datos.

Dentro de este conjunto importante de algoritmos podemos contar el algoritmo de Louvain, el algoritmo de propagación de etiquetas y dos tipos interesantes de algoritmos orientados a estudiar diferentes tipos de grafos. Para entender el funcionamiento de grafos dirigidos contamos con el algoritmo de componentes fuertemente conectados y para grafos no dirigidos el algoritmo de componentes débilmente conectados.

### SERVICIOS DE IBM

#### WATSON ASSISTANT

IBM Watson Assistant es la solución empresarial de IBM para que las empresas puedan crear sus propios asistentes personales.

## WATSON LANGUAGE TRANSLATOR

Language Translator traduce texto de un idioma a otro. Recopile noticias de todo el mundo y preséntelas en su idioma. Comuníquese con sus clientes en su propio idioma, y más.

## WATSON LANGUAGE UNDERSTANDING

Analiza texto para extraer metadatos de contenido, tales como conceptos, entidades, palabras clave, categorías, sentimientos, emociones, relaciones y roles semánticos utilizando la comprensión del lenguaje natural.

## 2 DESARROLLO

### 2.1 DESCRIPCION DETALLADA DEL PROBLEMA Y SU PROPUESTA DE SOLUCION.

Le ha pasado que algún momento, ha necesito hacer algún plato típico o alguna comida especial para sus seres queridos, pero quizá no tiene la disponibilidad para navegar por internet y saber que víveres necesita comprar, cabe mencionar que los platos están localizados por lugares donde son más conocidos, por nuestro **chatbot** está pensado para ese tipo de personas, para ello se tendrá el siguiente menú:

1. Ceviche de camarón
2. Encocada, pescado y camarón
3. Corviches
4. Arroz con camarón
5. La TIMBUSHCA
6. La Carne Colorada
7. Mote Pata
8. Fritada Cuencana

Toda la información se ha obtenido de una estudiante del Carrera de Hotelería y Turismo de Universidad de Cuenca.

#### Problema 1

##### Descripción del problema

Una persona que accede a nuestro chatbot desear conocer el plato más cotizado por listado de víveres.

#### Problema 2

##### Descripción del problema

Una persona que accede a nuestro chatbot desear conocer el plato más vendido por listado de víveres.

#### Propuesta de solución al problema 1

Para dar solución al problema se tiene en la base de Neo4j registro de los lugares con el plato de comida más comercial, para ello se usa el algoritmo de

centralidad por cercanía la cual devuelve el nombre del plato con mayor peso de centralidad de todos los otros platos.

#### Propuesta de solución al problema 2

Para esta solución se utiliza el algoritmo LOUVAIN el cual nos brinda el nodo con los mayores vecinos, para ello se carga una base de datos con la información de los platos.

### 2.2 DIAGRAMA ESQUEMATICO QUE EXPLIQUE COMO SE DESARROLLO Y MODELO DE LA PROPUESTA.

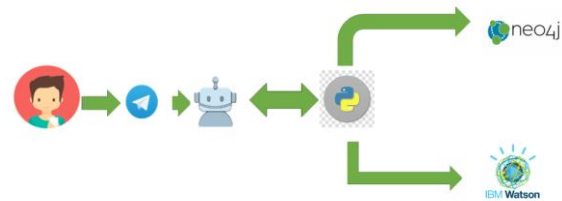


Fig. 1. Diagrama Esquelético del Sistema Chatbot.

Nuestro programa va a ser relativamente sencillo, el usuario se conectará con la aplicación de TELEGRAM, se contará con el asistente en TELEGRAM(Bot) el mismo que se conectará con la base de datos de NEO4J, con los servicios de IBM Watson todo esto conectado con Python.

### 2.3 ARQUITECTURA DEL SISTEMA DE SIMULACION

Como se mencionó en el problema para el procesamiento de todos nuestros datos se utilizará la base orientada a grafos de Neo4j.

El Bot a utilizar será el que nos proporciona TELEGRAM

IBM Watson nos proporciona los servicios de Watson Assistant, Watson Language Translator y Watson Natural Language Understanding.

### 2.4 DESCRIPCION DE LA SOLUCION Y PASOS SEGUIDOS (NIVEL INVESTIGATIVO).

Para la creación del Bot, primero es necesario tener una cuenta en la red social **TELEGRAM**.

Luego debemos conectar con BotFather.

Botfather es el Chatbot oficial de Telegram para crear nuevos chatbots (lo cual es bastante irónico). Escribiéndole unos pocos comandos, Botfather va a crear tu nuevo chatbot y te dará una clave única para que puedas controlarlo.

Empieza por **iniciar una conversación** con Botfather visitando [este link](#). Tanto si estás en el móvil como en

el PC, se ejecutará la aplicación de Telegram y se abrirá un chat con Botfather.

Pulsa el botón “Start” o escribe `/start` para iniciar la conversación con el chatbot. Recibirás como respuesta un mensaje que contiene todos los comandos básicos para configurar tu chatbot.

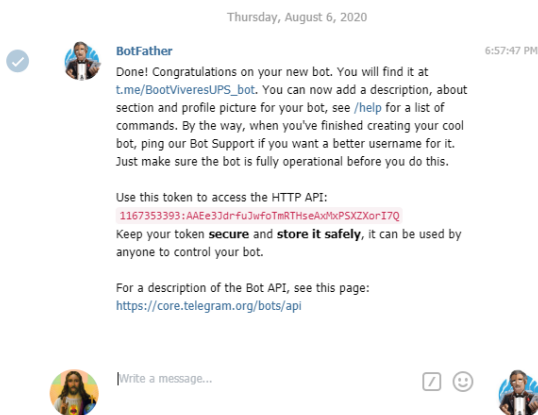


Fig. 2. Creación del Bot.

Para la implementación de solución para todos los problemas que se mencionó anteriormente todo esto se realizó en el lenguaje de Python, para ello se tuvo que investigar la forma en la que podía conectar el entorno de Python con Neo4j, cabe mencionar que nuestro servidor de base de datos se encuentra desplegado en la Nube, para ello utilizando la plataforma GOOGLE CLOUD, el enlace es el siguiente:

```
22 from neo4j import GraphDatabase
23
24 uri = "bolt://35.184.105.166:7687"
25 driver = GraphDatabase.driver(uri, auth=("neo4j", "admin"), encrypted=False)
26
```

Fig. 3. Conexión desde Python a Neo4.

Implementación del método para conexión con el Telegram.

```
111 def update(offset):
112     #Llamar al metodo getUpdates del bot, utilizando un offset
113     respuesta = requests.get(URL + "getUpdates" + "?offset=" + str(offset))
114     #Telegram devuelva todos los mensajes con id IGUAL o SUPERIOR al offset
115
116
117     #Decodificar la respuesta recibida a formato UTF8
118     mensajes_js = respuesta.content.decode("utf8")
119
120     #Convertir el string de JSON a un diccionario de Python
121     mensajes_diccionario = json.loads(mensajes_js)
122
123     #Devolver este diccionario
124     return mensajes_diccionario
125
126 def leer_mensaje(mensaje):
127
128     #Extraer el texto, nombre de la persona e id del último mensaje recibido
129     texto = mensaje["message"]["text"]
130     persona = mensaje["message"]["from"]["first_name"]
131     id_chat = mensaje["message"]["chat"]["id"]
132
133     #Calcular el identificador unico del mensaje para calcular el offset
134     id_update = mensaje["update_id"]
135
136     #Devolver las dos id, el nombre y el texto del mensaje
137     return id_chat, persona, texto, id_update
138
139 def enviar_mensaje(idchat, texto):
140     #Llamar al metodo sendMessage del bot, pasando el texto y la id del chat
141     requests.get(URL + "sendMessage?text=" + texto + "&chat_id=" + str(idchat))
```

Fig. 4. Conexión con Telegram.

Creación de la instancia para la utilización del servicio de Watson Assistant.

```
101 authenticator = IAMAuthenticator('CpT1M1HepcKhs5C0M8-mYhC8cWkrtC6_2zy30U')
102 assistant = AssistantV2(
103     version='2018-04-01',
104     authenticator=authenticator
105 )
106 assistant.set_service_url('https://api.us-south.assistant.watson.cloud.ibm.com/instances/6c4a58c-6f8d-4736-a997-41804c8')
107 session = assistant.create_session(
108     assistant_id='6c4a58c-6f8d-4736-a997-41804c8',
109     session_id='6c4a58c-6f8d-4736-a997-41804c8',
110     session_properties={}
111 )
112 message = AssistantMessage(
113     text='Hola, ¿cómo estás?',
114     input={
115         'message_type': 'text',
116         'text': 'mensaje de prueba'
117     },
118     session=session
119 )
120 get_result = assistant.get_result(message)
121 jsonData = json.dumps(get_result, indent=2)
122 jsonToPython = json.loads(jsonData)
123 v = jsonToPython['output']
124 res = v['generic']['0']['text']
```

Fig. 5. Creación de la instancia.

Creación de la instancia para la utilización del servicio de Watson Language Translator.

```
1 ## METODO PARA TRANSCUR
2 from ibm_cloud_sdk_core.authenticators import IAMAuthenticator
3 from ibm_watson import LanguageTranslatorV3
4
5 authenticator = IAMAuthenticator('InR0-d01or3Ctpy2H1Z6K7gagradVW0U91STEnPGU')
6 language_translator = LanguageTranslatorV3(
7     version='2018-05-01',
8     authenticator=authenticator
9 )
10
11 language_translator.set_service_url('https://api.us-south.language-translator.watson.cloud.ibm.com/instances/9472b6a7-295e-4b1d-8010-3c7p2H1Z6K7gagradVW0U91STEnPGU')
12
```

Fig. 6. Creación de la instancia.

Creación de la instancia para la utilización del servicio de Watson Language Understanding.

```
1 ## NATURAL LANGUAGE UNDERSTANDING
2 import json
3 from ibm_watson import NaturalLanguageUnderstandingV1
4 from ibm_watson.natural_language_understanding_v1 import Features, EntitiesOptions, EmotionOptions, KeywordsOptions, Metadata
5 from ibm_cloud_sdk_core.authenticators import IAMAuthenticator
6
7 authenticator = IAMAuthenticator('r7m4178suQL4743EDASg3g71nEhe-jpT884KMB11A')
8 service = NaturalLanguageUnderstandingV1(
9     version='2018-05-16',
10     authenticator=authenticator
11 )
12 service.set_service_url('https://api.us-south.natural-language-understanding.watson.cloud.ibm.com/instances/c54f98e9-4089-4a')
```

Fig. 7. Creación de la instancia.

Implementación del método de cercanía o centralidad.

```
22 from neo4j import GraphDatabase
23
24 uri = "bolt://35.184.105.166:7687"
25 driver = GraphDatabase.driver(uri, auth=("neo4j", "admin"), encrypted=False)
26
27 result=[]
28 parques = []
29 centralidad = []
30 def get_mo(tx):
31     result = tx.run("CALL gds.alpha.closeness.stream({nodeProjection: 'Plato',
32     'relationshipProjection: 'LINK' })"
33     "YIELD nodeId, centrality "
34     "RETURN gds.util.asNode(nodeId).name AS user, centrality "
35     "ORDER BY centrality DESC")
36     for record in result:
37         #print(record)
38         parques.append(record["user"])
39         centralidad.append(record["centrality"])
40
41 def recibir():
42     with driver.session() as session:
43         ciudades = session.read_transaction(get_mo)
44         #print(ciudades)
45         #print("Solución")
46         #print("Nodo -> Centralidad")
47         for i in range(len(parques)):
48             resultado=parques[0]
49             #print(parques[0])
50             #print(parques[1], " -> ", centralidad[1])
51             #print(parques[2], " -> ", centralidad[2])
52         #driver.close()
53         #print(resultado)
54     return resultado
55
```

Fig. 8. Método de cercanía.

Implementación del método LOUVAIN.

```
59 uri = "bolt://35.184.105.166:7687"
60 driver = GraphDatabase.driver(uri, auth=("neo4j", "admin"), encrypted=False)
61
62 resultado=[]
63 parques = []
64 centralidad = []
65 def get_mo(tx):
66     result = tx.run("CALL gds.louvain.stream({nodeProjection: 'Plato',
67     'relationshipProjection: {
68     'TYPE:{
69     'type: 'LINK',
70     'orientation: 'undirected',
71     'aggregation: 'NONE'
72     }
73     },
74     'includeIntermediateCommunities: true'
75     })YIELD nodeId, communityId, intermediateCommunityIds "
76     "RETURN gds.util.asNode(nodeId).name AS name, communityId, intermediateCommun
77     for record in result:
78         #print(record)
79         parques.append(record["name"])
80         centralidad.append(record["communityId"])
81
82 def recibir():
83     with driver.session() as session:
84         ciudades = session.read_transaction(get_mo)
85         #print("Solución")
86         #print("Nodo -> Centralidad")
87         for i in range(len(parques)):
88             resultado=parques[i]
89             resultado=parques[7]
90             #print(parques[i], " -> ", centralidad[i])
91         #print(resultado)
92         #driver.close()
93     return resultado
94
```

Fig. 9. Método de Louvain.

Implementación del método de traducción del idioma.

```
1 def tipoLenguaje(mensaje):
2     language = language_translator.identify(
3     mensaje).get_result()
4     r= json.dumps(language, indent=2)
5     rp= json.loads(r)
6     return rp["languages"][0]["language"]
7
8 def metodoTraducir(mensaje):
9     tam=tipoLenguaje(mensaje)
10    if(tam == "fr" or tam=="ca" or tam=="en"):
11        print("Idioma: ",tam)
12        translation = language_translator.translate(
13        text=mensaje,
14        model_id=tam+"-es").get_result()
15        r= json.dumps(translation, indent=2)
16        rp= json.loads(r)
17        return rp["translations"][0]["translation"],tam
18    else:
19        return mensaje,"es"
20
21 def traducirTexto(mensaje,tam):
22    if(tam == "fr" or tam=="ca" or tam=="en"):
23        translation = language_translator.translate(
24        text=mensaje,
25        model_id="es-"+tam).get_result()
26        r= json.dumps(translation, indent=2)
27        rp= json.loads(r)
28        return rp["translations"][0]["translation"]
29    else:
30        return mensaje
```

Fig. 10. Método de traducción.

Implementación del método para el análisis de una página web.

```
104 def Find(string):
105     regex = r"(?i)\b((?:https://www|http|ftp|rtsp|mms|mailto|tel|file|sftp|ssh|ldap|nntp|irc|irc6|irc7|irc8|irc9|irc10|irc11|irc12|irc13|irc14|irc15|irc16|irc17|irc18|irc19|irc20|irc21|irc22|irc23|irc24|irc25|irc26|irc27|irc28|irc29|irc30|irc31|irc32|irc33|irc34|irc35|irc36|irc37|irc38|irc39|irc40|irc41|irc42|irc43|irc44|irc45|irc46|irc47|irc48|irc49|irc50|irc51|irc52|irc53|irc54|irc55|irc56|irc57|irc58|irc59|irc60|irc61|irc62|irc63|irc64|irc65|irc66|irc67|irc68|irc69|irc70|irc71|irc72|irc73|irc74|irc75|irc76|irc77|irc78|irc79|irc80|irc81|irc82|irc83|irc84|irc85|irc86|irc87|irc88|irc89|irc90|irc91|irc92|irc93|irc94|irc95|irc96|irc97|irc98|irc99|irc100|irc101|irc102|irc103|irc104|irc105|irc106|irc107|irc108|irc109|irc110|irc111|irc112|irc113|irc114|irc115|irc116|irc117|irc118|irc119|irc120|irc121|irc122|irc123|irc124|irc125|irc126|irc127|irc128|irc129|irc130|irc131|irc132|irc133|irc134|irc135|irc136|irc137|irc138|irc139|irc140|irc141|irc142|irc143|irc144|irc145|irc146|irc147|irc148|irc149|irc150|irc151|irc152|irc153|irc154|irc155|irc156|irc157|irc158|irc159|irc160|irc161|irc162|irc163|irc164|irc165|irc166|irc167|irc168|irc169|irc170|irc171|irc172|irc173|irc174|irc175|irc176|irc177|irc178|irc179|irc180|irc181|irc182|irc183|irc184|irc185|irc186|irc187|irc188|irc189|irc190|irc191|irc192|irc193|irc194|irc195|irc196|irc197|irc198|irc199|irc200|irc201|irc202|irc203|irc204|irc205|irc206|irc207|irc208|irc209|irc210|irc211|irc212|irc213|irc214|irc215|irc216|irc217|irc218|irc219|irc220|irc221|irc222|irc223|irc224|irc225|irc226|irc227|irc228|irc229|irc230|irc231|irc232|irc233|irc234|irc235|irc236|irc237|irc238|irc239|irc240|irc241|irc242|irc243|irc244|irc245|irc246|irc247|irc248|irc249|irc250|irc251|irc252|irc253|irc254|irc255|irc256|irc257|irc258|irc259|irc260|irc261|irc262|irc263|irc264|irc265|irc266|irc267|irc268|irc269|irc270|irc271|irc272|irc273|irc274|irc275|irc276|irc277|irc278|irc279|irc280|irc281|irc282|irc283|irc284|irc285|irc286|irc287|irc288|irc289|irc290|irc291|irc292|irc293|irc294|irc295|irc296|irc297|irc298|irc299|irc300|irc301|irc302|irc303|irc304|irc305|irc306|irc307|irc308|irc309|irc310|irc311|irc312|irc313|irc314|irc315|irc316|irc317|irc318|irc319|irc320|irc321|irc322|irc323|irc324|irc325|irc326|irc327|irc328|irc329|irc330|irc331|irc332|irc333|irc334|irc335|irc336|irc337|irc338|irc339|irc340|irc341|irc342|irc343|irc344|irc345|irc346|irc347|irc348|irc349|irc350|irc351|irc352|irc353|irc354|irc355|irc356|irc357|irc358|irc359|irc360|irc361|irc362|irc363|irc364|irc365|irc366|irc367|irc368|irc369|irc370|irc371|irc372|irc373|irc374|irc375|irc376|irc377|irc378|irc379|irc380|irc381|irc382|irc383|irc384|irc385|irc386|irc387|irc388|irc389|irc390|irc391|irc392|irc393|irc394|irc395|irc396|irc397|irc398|irc399|irc400|irc401|irc402|irc403|irc404|irc405|irc406|irc407|irc408|irc409|irc410|irc411|irc412|irc413|irc414|irc415|irc416|irc417|irc418|irc419|irc420|irc421|irc422|irc423|irc424|irc425|irc426|irc427|irc428|irc429|irc430|irc431|irc432|irc433|irc434|irc435|irc436|irc437|irc438|irc439|irc440|irc441|irc442|irc443|irc444|irc445|irc446|irc447|irc448|irc449|irc450|irc451|irc452|irc453|irc454|irc455|irc456|irc457|irc458|irc459|irc460|irc461|irc462|irc463|irc464|irc465|irc466|irc467|irc468|irc469|irc470|irc471|irc472|irc473|irc474|irc475|irc476|irc477|irc478|irc479|irc480|irc481|irc482|irc483|irc484|irc485|irc486|irc487|irc488|irc489|irc490|irc491|irc492|irc493|irc494|irc495|irc496|irc497|irc498|irc499|irc500|irc501|irc502|irc503|irc504|irc505|irc506|irc507|irc508|irc509|irc510|irc511|irc512|irc513|irc514|irc515|irc516|irc517|irc518|irc519|irc520|irc521|irc522|irc523|irc524|irc525|irc526|irc527|irc528|irc529|irc530|irc531|irc532|irc533|irc534|irc535|irc536|irc537|irc538|irc539|irc540|irc541|irc542|irc543|irc544|irc545|irc546|irc547|irc548|irc549|irc550|irc551|irc552|irc553|irc554|irc555|irc556|irc557|irc558|irc559|irc560|irc561|irc562|irc563|irc564|irc565|irc566|irc567|irc568|irc569|irc570|irc571|irc572|irc573|irc574|irc575|irc576|irc577|irc578|irc579|irc580|irc581|irc582|irc583|irc584|irc585|irc586|irc587|irc588|irc589|irc590|irc591|irc592|irc593|irc594|irc595|irc596|irc597|irc598|irc599|irc600|irc601|irc602|irc603|irc604|irc605|irc606|irc607|irc608|irc609|irc610|irc611|irc612|irc613|irc614|irc615|irc616|irc617|irc618|irc619|irc620|irc621|irc622|irc623|irc624|irc625|irc626|irc627|irc628|irc629|irc630|irc631|irc632|irc633|irc634|irc635|irc636|irc637|irc638|irc639|irc640|irc641|irc642|irc643|irc644|irc645|irc646|irc647|irc648|irc649|irc650|irc651|irc652|irc653|irc654|irc655|irc656|irc657|irc658|irc659|irc660|irc661|irc662|irc663|irc664|irc665|irc666|irc667|irc668|irc669|irc670|irc671|irc672|irc673|irc674|irc675|irc676|irc677|irc678|irc679|irc680|irc681|irc682|irc683|irc684|irc685|irc686|irc687|irc688|irc689|irc690|irc691|irc692|irc693|irc694|irc695|irc696|irc697|irc698|irc699|irc700|irc701|irc702|irc703|irc704|irc705|irc706|irc707|irc708|irc709|irc710|irc711|irc712|irc713|irc714|irc715|irc716|irc717|irc718|irc719|irc720|irc721|irc722|irc723|irc724|irc725|irc726|irc727|irc728|irc729|irc730|irc731|irc732|irc733|irc734|irc735|irc736|irc737|irc738|irc739|irc740|irc741|irc742|irc743|irc744|irc745|irc746|irc747|irc748|irc749|irc750|irc751|irc752|irc753|irc754|irc755|irc756|irc757|irc758|irc759|irc760|irc761|irc762|irc763|irc764|irc765|irc766|irc767|irc768|irc769|irc770|irc771|irc772|irc773|irc774|irc775|irc776|irc777|irc778|irc779|irc780|irc781|irc782|irc783|irc784|irc785|irc786|irc787|irc788|irc789|irc790|irc791|irc792|irc793|irc794|irc795|irc796|irc797|irc798|irc799|irc800|irc801|irc802|irc803|irc804|irc805|irc806|irc807|irc808|irc809|irc810|irc811|irc812|irc813|irc814|irc815|irc816|irc817|irc818|irc819|irc820|irc821|irc822|irc823|irc824|irc825|irc826|irc827|irc828|irc829|irc830|irc831|irc832|irc833|irc834|irc835|irc836|irc837|irc838|irc839|irc840|irc841|irc842|irc843|irc844|irc845|irc846|irc847|irc848|irc849|irc850|irc851|irc852|irc853|irc854|irc855|irc856|irc857|irc858|irc859|irc860|irc861|irc862|irc863|irc864|irc865|irc866|irc867|irc868|irc869|irc870|irc871|irc872|irc873|irc874|irc875|irc876|irc877|irc878|irc879|irc880|irc881|irc882|irc883|irc884|irc885|irc886|irc887|irc888|irc889|irc890|irc891|irc892|irc893|irc894|irc895|irc896|irc897|irc898|irc899|irc900|irc901|irc902|irc903|irc904|irc905|irc906|irc907|irc908|irc909|irc910|irc911|irc912|irc913|irc914|irc915|irc916|irc917|irc918|irc919|irc920|irc921|irc922|irc923|irc924|irc925|irc926|irc927|irc928|irc929|irc930|irc931|irc932|irc933|irc934|irc935|irc936|irc937|irc938|irc939|irc940|irc941|irc942|irc943|irc944|irc945|irc946|irc947|irc948|irc949|irc950|irc951|irc952|irc953|irc954|irc955|irc956|irc957|irc958|irc959|irc960|irc961|irc962|irc963|irc964|irc965|irc966|irc967|irc968|irc969|irc970|irc971|irc972|irc973|irc974|irc975|irc976|irc977|irc978|irc979|irc980|irc981|irc982|irc983|irc984|irc985|irc986|irc987|irc988|irc989|irc990|irc991|irc992|irc993|irc994|irc995|irc996|irc997|irc998|irc999|irc1000|irc1001|irc1002|irc1003|irc1004|irc1005|irc1006|irc1007|irc1008|irc1009|irc1010|irc1011|irc1012|irc1013|irc1014|irc1015|irc1016|irc1017|irc1018|irc1019|irc1020|irc1021|irc1022|irc1023|irc1024|irc1025|irc1026|irc1027|irc1028|irc1029|irc1030|irc1031|irc1032|irc1033|irc1034|irc1035|irc1036|irc1037|irc1038|irc1039|irc1040|irc1041|irc1042|irc1043|irc1044|irc1045|irc1046|irc1047|irc1048|irc1049|irc1050|irc1051|irc1052|irc1053|irc1054|irc1055|irc1056|irc1057|irc1058|irc1059|irc1060|irc1061|irc1062|irc1063|irc1064|irc1065|irc1066|irc1067|irc1068|irc1069|irc1070|irc1071|irc1072|irc1073|irc1074|irc1075|irc1076|irc1077|irc1078|irc1079|irc1080|irc1081|irc1082|irc1083|irc1084|irc1085|irc1086|irc1087|irc1088|irc1089|irc1090|irc1091|irc1092|irc1093|irc1094|irc1095|irc1096|irc1097|irc1098|irc1099|irc1100|irc1101|irc1102|irc1103|irc1104|irc1105|irc1106|irc1107|irc1108|irc1109|irc1110|irc1111|irc1112|irc1113|irc1114|irc1115|irc1116|irc1117|irc1118|irc1119|irc1120|irc1121|irc1122|irc1123|irc1124|irc1125|irc1126|irc1127|irc1128|irc1129|irc1130|irc1131|irc1132|irc1133|irc1134|irc1135|irc1136|irc1137|irc1138|irc1139|irc1140|irc1141|irc1142|irc1143|irc1144|irc1145|irc1146|irc1147|irc1148|irc1149|irc1150|irc1151|irc1152|irc1153|irc1154|irc1155|irc1156|irc1157|irc1158|irc1159|irc1160|irc1161|irc1162|irc1163|irc1164|irc1165|irc1166|irc1167|irc1168|irc1169|irc1170|irc1171|irc1172|irc1173|irc1174|irc1175|irc1176|irc1177|irc1178|irc1179|irc1180|irc1181|irc1182|irc1183|irc1184|irc1185|irc1186|irc1187|irc1188|irc1189|irc1190|irc1191|irc1192|irc1193|irc1194|irc1195|irc1196|irc1197|irc1198|irc1199|irc1200|irc1201|irc1202|irc1203|irc1204|irc1205|irc1206|irc1207|irc1208|irc1209|irc1210|irc1211|irc1212|irc1213|irc1214|irc1215|irc1216|irc1217|irc1218|irc1219|irc1220|irc1221|irc1222|irc1223|irc1224|irc1225|irc1226|irc1227|irc1228|irc1229|irc1230|irc1231|irc1232|irc1233|irc1234|irc1235|irc1236|irc1237|irc1238|irc1239|irc1240|irc1241|irc1242|irc1243|irc1244|irc1245|irc1246|irc1247|irc1248|irc1249|irc1250|irc1251|irc1252|irc1253|irc1254|irc1255|irc1256|irc1257|irc1258|irc1259|irc1260|irc1261|irc1262|irc1263|irc1264|irc1265|irc1266|irc1267|irc1268|irc1269|irc1270|irc1271|irc1272|irc1273|irc1274|irc1275|irc1276|irc1277|irc1278|irc1279|irc1280|irc1281|irc1282|irc1283|irc1284|irc1285|irc1286|irc1287|irc1288|irc1289|irc1290|irc1291|irc1292|irc1293|irc1294|irc1295|irc1296|irc1297|irc1298|irc1299|irc1300|irc1301|irc1302|irc1303|irc1304|irc1305|irc1306|irc1307|irc1308|irc1309|irc1310|irc1311|irc1312|irc1313|irc1314|irc1315|irc1316|irc1317|irc1318|irc1319|irc1320|irc1321|irc1322|irc1323|irc1324|irc1325|irc1326|irc1327|irc1328|irc1329|irc1330|irc1331|irc1332|irc1333|irc1334|irc1335|irc1336|irc1337|irc1338|irc1339|irc1340|irc1341|irc1342|irc1343|irc1344|irc1345|irc1346|irc1347|irc1348|irc1349|irc1350|irc1351|irc1352|irc1353|irc1354|irc1355|irc1356|irc1357|irc1358|irc1359|irc1360|irc1361|irc1362|irc1363|irc1364|irc1365|irc1366|irc1367|irc1368|irc1369|irc1370|irc1371|irc1372|irc1373|irc1374|irc1375|irc1376|irc1377|irc1378|irc1379|irc1380|irc1381|irc1382|irc1383|irc1384|irc1385|irc1386|irc1387|irc1388|irc1389|irc1390|irc1391|irc1392|irc1393|irc1394|irc1395|irc1396|irc1397|irc1398|irc1399|irc1400|irc1401|irc1402|irc1403|irc1404|irc1405|irc1406|irc1407|irc1408|irc1409|irc1410|irc1411|irc1412|irc1413|irc1414|irc1415|irc1416|irc1417|irc1418|irc1419|irc1420|irc1421|irc1422|irc1423|irc1424|irc1425|irc1426|irc1427|irc1428|irc1429|irc1430|irc1431|irc1432|irc1433|irc1434|irc1435|irc1436|irc1437|irc1438|irc1439|irc1440|irc1441|irc1442|irc1443|irc1444|irc1445|irc1446|irc1447|irc1448|irc1449|irc1450|irc1451|irc1452|irc1453|irc1454|irc1455|irc1456|irc1457|irc1458|irc1459|irc1460|irc1461|irc1462|irc1463|irc1464|irc1465|irc1466|irc1467|irc1468|irc1469|irc1470|irc1471|irc1472|irc1473|irc1474|irc1475|irc1476|irc1477|irc1478|irc1479|irc1480|irc1481|irc1482|irc1483|irc1484|irc1485|irc1486|irc1487|irc1488|irc1489|irc1490|irc1491|irc1492|irc1493|irc1494|irc1495|irc1496|irc1497|irc1498|irc1499|irc1500|irc1501|irc1502|irc1503|irc1504|irc1505|irc1506|irc1507|irc1508|irc1509|irc1510|irc1511|irc1512|irc1513|irc1514|irc1515|irc1516|irc1517|irc1518|irc1519|irc1520|irc1521|irc1522|irc1523|irc1524|irc1525|irc1526|irc1527|irc1528|irc1529|irc1530|irc1531|irc1532|irc1533|irc1534|irc1535|irc1536|irc1537|irc1538|irc1539|irc1540|irc1541|irc1542|irc1543|irc1544|irc1545|irc1546|irc1547|irc1548|irc1549|irc1550|irc1551|irc1552|irc1553|irc1554|irc1555|irc1556|irc1557|irc1558|irc1559|irc1560|irc1561|irc1562|irc1563|irc1564|irc1565|irc1566|irc1567|irc1568|irc1569|irc1570|irc1571|irc1572|irc1573|irc1574|irc1575|irc1576|irc1577|irc1578|irc1579|irc1580|irc1581|irc1582|irc1583|irc1584|irc1585|irc1586|irc1587|irc1588|irc1589|irc1590|irc1591|irc1592|irc1593|irc1594|irc1595|irc1596|irc1597|irc1598|irc1599|irc1600|irc1601|irc1602|irc1603|irc1604|irc1605|irc1606|irc1607|irc1608|irc1609|irc1610|irc1611|irc1612|irc1613|irc1614|irc1615|irc1616|irc1617|irc1618|irc1619|irc1620|irc1621|irc1622|irc1623|irc1624|irc1625|irc1626|irc1627|irc1628|irc1629|irc1630|irc1631|irc1632|irc1633|irc1634|irc1635|irc1636|irc1637|irc1638|irc1639|irc1640|irc1641|irc1642|irc1643|irc1644|irc1645|irc1646|irc1647|irc1648|irc1649|irc1650|irc1651|irc1652|irc1653|irc1654|irc1655|irc1656|irc1657|irc1658|irc1659|irc1660|irc1661|irc1662|irc1663|irc1664|irc1665|irc1666|irc1667|irc1668|irc1669|irc1670|irc1671|irc1672|irc1673|irc1674|irc1675|irc1676|irc1677|irc1678|irc1679|irc1680|irc1681|irc1682|irc1683|irc1684|irc1685|irc1686|irc1687|irc1688|irc1689|irc1690|irc1691|irc1692|irc1693|irc1694|irc1695|irc1696|irc1697|irc1698|irc1699|irc1700|irc1701|irc1702|irc1703|irc1704|irc1705|irc1706|irc1707|irc1708|irc1709|irc1710|irc1711|irc1712|irc1713|irc1714|irc1715|irc1716|irc1717|irc1718|irc1719|irc1720|irc1721|irc1722|irc1723|irc1724|irc1725|irc1726|irc1727|irc1728|irc1729|irc1730|irc1731|irc1732|irc1733|irc1734|irc1735|irc1736|irc1737|irc1738|irc1739|irc1740|irc1741|irc1742|irc1743|irc1744|irc1745|irc1746|irc1747|irc1748|irc1749|irc1750|irc1751|irc1752|irc1753|irc1754|irc1755|irc1756|irc1757|irc1758|irc1759|irc1760|irc1761|irc1762|irc1763|irc1764|irc1765|irc1766|irc1767|irc1768|irc1769|irc1770|irc1771|irc1772|irc1773|irc1774|irc1775|irc1776|irc1777|irc1778|irc1779|irc1780|irc1781|irc1782|irc1783|irc1784|irc1785|irc1786|irc1787|irc1788|irc1789|irc1790|irc1791|irc1792|irc1793|irc1794|irc1795|irc1796|irc1797|irc1798|irc1799|irc1800|irc1801|irc1802|irc1803|irc1804|irc1805|irc1806|irc1807|irc1808|irc1809|irc1810|irc1811|irc1812|irc1813|irc1814|irc1815|irc1816|irc1817|irc1818|irc1819|irc1820|irc1821|irc1822|irc1823|irc1824|irc1825|irc1826|irc1827|irc1828|irc1829|irc1830|irc1831|irc1832|irc1833|irc1834|irc1835|irc1836|irc1837|irc1838|irc1839|irc1840|irc1841|irc1842|irc1843|irc1844|irc1845|irc1846|irc1847|irc1848|irc1849|irc1850|irc1851|irc1852|irc1853|irc1854|irc1855|irc1856|irc1857|irc1858|irc1859|irc1860|irc1861|irc1862|irc1863|irc1864|irc1865|irc1866|irc1867|irc1868|irc1869|irc1870|irc1871|irc1872|irc1873|irc1874|irc1875|irc1876|irc1877|irc1878|irc1879|irc1880|irc1881|irc1882|irc1883|irc1884|irc1885|irc1886|irc1887|irc1888|irc1889|irc1890|irc1891|irc1892|irc1893|irc1894|irc1895|irc1896|irc1897|irc1898|irc1899|irc1900|irc1901|irc1902|irc1903|irc1904|irc1905|irc1906|irc1907|irc1908|irc1909|irc1910|irc1911|irc1912|irc1913|irc1914|irc1915|irc1916|irc1917|irc1918|irc1919|irc1920|irc1921|irc1922|irc1923|irc1924|irc1925|irc1926|irc1927|irc1928|irc1929|irc1930|irc1931|irc1932|irc1933|irc1934|irc1935|irc1936|irc1937|irc1938|irc1939|irc1940|irc1941|irc1942|irc1943|irc1944|irc1945|irc1946|irc1947|irc1948|irc1949|irc1950|irc1951|irc1952|irc1953|irc1954|irc1955|irc1956|irc1957|irc1958|irc1959|irc1960|irc1961|irc1962|irc1963|irc1964|irc1965|irc1966|irc1967|irc1968|irc1969|irc1970|irc1971|irc1972|irc1973|irc1974|irc1975|irc1976|irc1977|irc1978|irc1979|irc1980|irc1981|irc1982|irc
```



Fig. 13. Servicio de Traducción.

### 3. Función del servicio del análisis de la página web.



Fig. 14. Servicio de análisis de página web.



Fig. 15. Base de Datos de Neo4j.

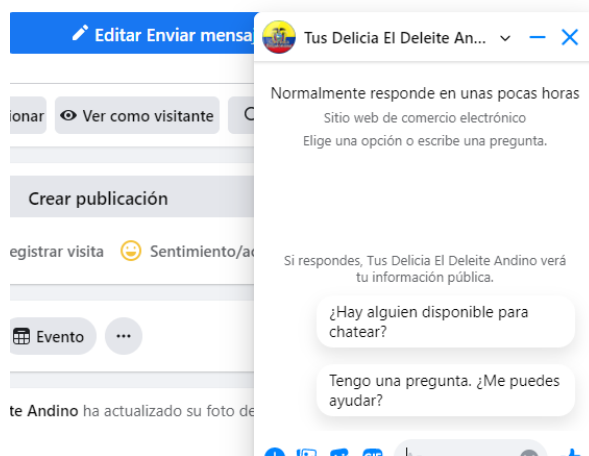


Fig. 16. Integración con Facebook.



Fig. 17. Integración con WhatsApp.

### 3. CONCLUSIONES:

El desarrollo del presente trabajo nos ha permitido poder profundizar nuestros conocimientos y ver que este tipo de algoritmos en los que quizá de alguna manera se intuye inteligencia nos ayudan a poder saber que decisiones poder con una certeza de que las cosas se hacen en base a datos que pueden apoyar nuestra decisión y no quizá por nuestra intuición.

Se implemento los 3 servicios de Watson donde cada uno funciona correctamente, también se realizó el sistema experto y la inserción de datos en Neo4j. Los algoritmos de Cercanía Centralidad y Louvain realizan las búsquedas correctamente. Para consumir todos los servicios se realizó con Telegram.

El desarrollo del ChatBot realmente ha sido una experticia nueva que nos permitido generar nuevas ideas que se podrían implantar en muchos negocios en el futuro.

### 4. RECOMENDACIONES:

-Como una recomendación general se pediría que los estudiantes recurran al profesor para obtener más información en el caso de que no se comprenda el tema a investiga, debido a que como el tema es nuevo para nosotros muchas veces no se entiende claramente que los que el docente nos pide y se corre el riesgo de realizar el proyecto de la forma que no se debe hacer.

La realización del proyecto nos ha permitido generar nuevas destrezas.

Revisar la documentación de cada una de las formas que nos brindan cada uno de los desarrolladores para integrar el chatbot con las plataformas sociales.

### 5. ANEXO

En este apartado adjuntamos el enlace de nuestra página en la que se encuentra de manera detallada todos los aspectos que se explicaron anteriormente

<https://sites.google.com/view/proyectointegradorchatbot>

Toda la información de los algoritmos y sus aplicaciones se encuentra en el cuaderno de Júpiter.

### 6. REFERENCIAS

- [1] Neo4j Graph Platform – The Leader in Graph Databases. (2020). Retrieved 9 August 2020, from <https://neo4j.com/>
- [2] Install Neo4j on Debian (2020). Retrieved 9 August 2020, from <https://neo4j.com/docs/operations-manual/current/installation/linux/debian/>