ISSN: 2594 - 2905

DESARROLLO DE CHATBOT PARA EL INSTITUTO TECNOLÓGICO DE ZACATEPEC

DEVELOPMENT OF CHATBOT FOR THE TECHNOLOGICAL INSTITUTE OF ZACATEPEC

Pérez Calderón Carlos[1], Vázquez Carrillo Luis Ángel[2], Muñiz-Rascado Luis José[3], Velázquez-Santana Eugenio César[4], Castro-Sánchez Noé Alejandro[5]

https://doi.org/10.61117/ipsumtec.v8i2.390

[1]Tecnológico Nacional de México/Instituto Tecnológico de Zacatepec, L19091404@zacatepec.tecnm.mx, https://orcid.org/0009-0003-0590-4885.
[2]Tecnológico Nacional de México/Instituto Tecnológico de Zacatepec, L20091199@zacatepec.tecnm.mx, https://orcid.org/0009-0007-9101-5513.
[3]Tecnológico Nacional de México/Instituto Tecnológico de Zacatepec, luis.mr@zacatepec.tecnm.mx, https://orcid.org/0000-0003-2941-2482.
[4]Tecnológico Nacional de México/Instituto Tecnológico de Zacatepec, eugenio.vs@zacatepec.tecnm.mx, https://orcid.org/0000-0002-7683-7224.
[5]Tecnológico Nacional de México/CENIDET, noe.cs@cenidet.tecnm.mx, https://orcid.org/0000-0002-8083-3891.

Resumen – El presente artículo expone el desarrollo de un chatbot destinado al ITZ Instituto Tecnológico de Zacatepec entrenado con datos relevantes de la institución para proporcionar una atención ágil y natural a las consultas de los usuarios en la página de la universidad. El chatbot propuesto utiliza tecnología de modelos de lenguaje de gran tamaño o LLM (*Large Language Model*), junto con algoritmos avanzados de recuperación de información. El sistema busca reducir la carga administrativa sobre el personal y ofrecer a los usuarios una experiencia interactiva y eficiente. El resultado final es un sistema computacional que recibe consulta del usuario; el algoritmo de recuperación BM25 busca entre un corpus de datos relacionados con la institución, y finalmente el modelo LLM lo procesa, generando una respuesta a la solicitud formulada.

Palabras Clave: BM25, Chatbot, IA, LLM.

Abstract -- The present article presents the development of a chatbot designed for TecNM Instituto Tecnológico de Zacatepec, trained with relevant institutional data to provide agile and natural user assistance on the university's website. The proposed chatbot utilizes Large Language Model (LLM) technology along with advanced information retrieval algorithms. The system aims to reduce the administrative burden on staff and offer users an interactive and efficient experience. The result is a computational system that receives user queries; the BM25 retrieval algorithm searches through a corpus of institution-related data, and finally, the LLM model (GEMMA) processes the context, generating an appropriate response to the request.

Palabras Clave: Chatbot, IA, LLM, BM25.

INTRODUCCIÓN

Los métodos tradicionales de atención a usuarios, como líneas telefónicas y correos electrónicos, resultan insuficientes para manejar un gran volumen de consultas, generando tiempos de espera prolongados y una carga administrativa considerable. No obstante, en la actualidad, la inteligencia artificial (IA) ha cobrado un papel fundamental en la optimización de estos procesos. Según Caldarin et al. [1], la IA incluye diversas áreas como el Procesamiento del Lenguaje Natural (PLN), la síntesis y comprensión del habla, entre otras. Además, su capacidad para sistematizar y analizar grandes volúmenes de información la convierte en una herramienta clave para la automatización de tareas.

El auge de la IA ha impulsado la automatización en múltiples sectores. Un chatbot es un programa informático que simula la conversación humana con un usuario final según IBM [15]. Los chatbots representan potencialmente un nuevo paradigma en la forma en que las personas interactuarán con datos y servicios en el futuro [12], facilitando la atención a usuarios y optimizando la gestión de información.

En este contexto, se propuso el desarrollo de un chatbot para el TecNM Instituto Tecnológico de Zacatepec, con el objetivo de brindar un asistente virtual que resuelva dudas y proporcione información institucional. Esto permitirá reducir la carga de trabajo administrativo y ofrecer respuestas rápidas y eficientes a los usuarios.

DESARROLLO

Para el desarrollo de este proyecto se partió de una investigación cualitativa, en la cual se investigaron varios modelos LLM como tecnología para potenciar el chatbot, los cuales pasaron por un proceso de evaluación para determinar cuál era el más óptimo para el proyecto, después se implementó un algoritmo de búsqueda para localizar documentos relevantes dentro de un corpus de

Recepción: 14/06/2025 | Aceptación: 15/09/2025 | Publicación: 09/10/2025

documentos, para ser usado por el modelo y que lograra usar la información recuperada como contexto para generar las respuestas a las preguntas.

Algoritmo de recuperación BM25.
Este algoritmo fue propuesto por dos investigadores [3], un programador británico licenciado en matemáticas por la Universidad de Cambridge, y Spärck Jones, una científica británica especializada en lingüística computacional. Desarrollaron un modelo matemático basado en principios probabilísticos para la localización de documentos. Su enfoque se centra en determinar la relevancia de ciertas palabras dentro de un corpus de documentos, dividiendo el proceso en dos cálculos fundamentales: TF e IDF.

TF.
Mide la relevancia de un término dentro de un documento, considerando la cantidad de veces que aparece en dicho documento, aunque puede tener un sesgo con documentos largos por el simple hecho de tener más palabras, es por eso que a diferencia de los enfoques simples como TF-IDF, BM25 emplea una versión "normalizada" del TF para manejar mejores situaciones en las que un término aparece muy frecuentemente evitando este sesgo, dicho calculo se obtiene de la ecuación Ec.(1).

$$TF = \left[\frac{f(q_i,\, D) \cdot (k_i + 1)}{f(q_i,\, D) + k_1 \cdot (1 - b + b \cdot \frac{|D|}{avgdl})} \right] \qquad \text{Ec.(1)}$$

Donde:
- $f(q_i,\, D)$: Indica cuántas veces aparece el término q_i en el documento D.
- k_i: Es un parámetro de ajuste que controla la sensibilidad de la frecuencia de término.
- $k_i + 1$ aumenta la sensibilidad para darle más peso a la frecuencia del término.
- b: Parámetro de ajuste que controla la longitud del documento en la fórmula b toma valores entre 0 y 1.
- $|D|$: Nnúmero total de palabras en el documento.
- $avgdl$: Es la longitud promedio de los documentos en el corpus.

IDF
El *Inverse Document Frequency* o Frecuencia Inversa de Documentos por sus siglas en ingles es una medida utilizada para evaluar la importancia de un término dentro de un corpus de documentos. El IDF asigna un peso mayor a los términos que son raros en el corpus y un peso menor a los términos que son comunes. La idea es que los términos comunes (como "el", "la", "de", entre otros.) no aportan mucha información para distinguir entre documentos, mientras que los términos raros (como "algoritmo", "programación", "neuronal") suelen ser más relevantes para identificar el contenido de un documento, la formula clásica se representa con la siguiente ecuación Ec.(2).

$$IDF(t) = \log\left(\frac{N}{df(t)}\right) \qquad \text{Ec.(2)}$$

Donde:
- t es el término o palabra.
- N es el número total de documentos en el corpus.
- $df(t)$ es el número de documentos en los que aparece el término.

Combinación final.
Al combinar estos dos conceptos TF e IDF obtenemos como resultado el algoritmo BM25, el 25 viene de la versión usada para este proyecto [3] Ec.(3).

$$score(D,Q) = \sum_{i=1}^{n} IDF(q_i) \cdot \left[\frac{f(q_i,\, D) \cdot (k_i + 1)}{f(q_i,\, D) + k_1 \cdot \left(1 - b + b \cdot \frac{|D|}{avgdl}\right)} \right] \qquad \text{Ec.(3)}$$

Donde:
- $\mathrm{score(D, Q)}$: Es el puntaje total de relevancia del documento D con respecto a la consulta Q.
- $\sum_{i=1}^{n} \mathrm{IDF(q_i)}$: Aquí se suma el aporte de cada término q_1 de la consulta Q al puntaje final.
- n Es el número total de términos en la consulta.
- $\mathrm{IDF(q_i)}$ es la "Frecuencia Inversa de Documentos".
- $f(q_i,\, D)$: Es la Frecuencia de Término (TF).
- k_i: Es un parámetro de ajuste que controla la sensibilidad de la frecuencia de término $f(q_i,\, D)$.
- b: Es otro parámetro de ajuste que controla la longitud del documento en la fórmula.
- $|D|$: Es la longitud del documento D, número total de palabras en el documento.
- $avgdl$: Es la longitud promedio de los documentos en el corpus.

Índices de lecturabilidad del texto:
Fueron cálculos aplicados al texto para evaluar la complejidad tanto de las respuestas generadas por el LLM como de las preguntas hechas por el usuario.

Fernandez-Huerta.
La lecturabilidad viene a ser la legibilidad lingüística del texto, es decir, si es fácil o difícil de entender. No abarca aspectos tipográficos que influyen mucho en la facilidad de lectura [2], ver Ec.(4).

$$L = 206.84 - 0.60P - 1.02F \qquad \text{Ec.(4)}$$

Donde:
- L = lecturabilidad.
- P = promedio de sílabas por palabra.
- F = media de palabras por frase.

El resultado se interpreta en la Tabla 1.

Recepción: 14/06/2025 | Aceptación: 15/09/2025 | Publicación: 09/10/2025

Tabla 1. *Escala de lecturabilidad Fernández Huerta*.

(L)	Nivel	Grado escolar
90-100	Muy fácil	4º grado
80-90	Fácil	5º grado
70-80	Algo fácil	6º grado
60-70	Normal	7º u 8º grado
50-60	Algo difícil	Preuniversitario
30-50	Difícil	Cursos selectivos
0-30	Muy difícil	Universitario

Fuente. *Medidas sencillas de lecturabilidad (Fernández Huerta, 1959)*.

INFLESZ

La escala de legibilidad INFLESZ mide la facilidad de leer un texto. Fue desarrollada por Inés María Barrio Cantalejo. Está adaptada al lector español medio actual [6] ver Ec.(5)**.**

$$I = 206.835 - 62.3\left(\frac{S}{P}\right) - \left(\frac{P}{F}\right) \qquad \text{Ec.(5)}$$

Donde:

- I = escala INFLESZ.
- S = total de silabas.
- P = cantidad de palabras.
- F = número de frases.

El resultado se interpreta en la Tabla 2.

Tabla 2. *Escala de lecturabilidad INFLESZ.*

(L)	Nivel
80-100	Muy fácil
65-80	Fácil
55-65	Algo fácil
40-55	Normal (para un adulto)
0-40	Difícil

Fuente. *Sistemas Predictivos De Legibilidad Del Mensaje Escrito (Francisco Szigriszt Pazos, 1992).*

Esquema de la aplicación.

El esquema que se analiza en la Figura 1 ilustra la arquitectura del sistema del chatbot, revelando la sinergia entre algoritmos de recuperación de información y el uso del modelo de procesamiento de texto GEMMA 2B.

Figura 1. *Arquitectura del sistema*.

Fuente*. Elaboración propia (2025).*

Evaluación de los modelos.

Se evaluó la complejidad de los prompts y respuestas mediante los índices de Fernández-Huerta e INFLESZ [2], [6], cronometrando el tiempo de generación y entrega de respuestas del modelo. En el **Código 1** se puede observar el código en Python donde se evalúa el texto para determinar los resultados y conocer la complejidad de los *prompts.*

Código 1. *Cálculos de índices en Python*.

```
def fernandez_huerta(texto):
    palabras = contar_palabras(texto)
    oraciones = contar_oraciones(texto)
    silabas = sum(contar_silabas(palabra)
for palabra in
re.findall(r'\b\w+\b'texto))

    return 206.84 - 60 * (silabas /
palabras) - 1.02 * (palabras/oraciones)

def inflesz(texto):
    palabras = contar_palabras(texto)
    oraciones = contar_oraciones(texto)
    silabas = sum(contar_silabas(palabra)
for palabra in
re.findall(r'\b\w+\b'texto))

    return 206.835 - 62.3 * (silabas /
palabras) - (palabras/oraciones)
```

Para determinar el LLM óptimo para el proyecto, fue necesaria una investigación sobre los diferentes modelos disponibles en el sector, con el objetivo de identificar cuál se adapta mejor a las necesidades específicas en función de la cantidad de parámetros, la coherencia de las respuestas y el tiempo de ejecución, ver en Tabla 3.

Tabla 3. *Información general de los modelos LLM*.

MODELO	PARAMETROS	DESARROLLADOR
LLAMA2	7B	Meta
LLAMA 3	70B	Meta
MISTRAL	7B	Mistral AI
GEMMA	2B	Google
PHI3	3B	Microsoft
PHI3	14 B	Microsoft
LLAMA2	7B	Meta

Fuente. *Elaboración propia (2025)*.

Para los requerimientos mínimos para correr el sistema se tiene que contar con lo con lo descrito en la Tabla 4.

Tabla 4. *Requerimiento del modelo LLM inicial*.

CPU	RAM	VRAM
Intel Core i5 / AMD Ryzen 5	16 GB	6 – 8 GB
Intel Core i9 / AMD Ryzen 9	32 GB	16 GB
Intel Core i7 / AMD Ryzen 7	16 GB	6 – 8 GB
Intel Core i5 / AMD Ryzen 5	16 GB	8 GB

Recepción: 14/06/2025 | Aceptación: 15/09/2025 | Publicación: 09/10/2025

Intel Core i5 / AMD Ryzen 5	4 GB	2 GB
Intel Core i5 / AMD Ryzen 5	8 GB	4 GB
Intel Core i5 / AMD Ryzen 5	16 GB	6 – 8 GB

Fuente. *Elaboración propia (2025).*

De los cuales se seleccionaron para pruebas de escritorio las siguientes modelos LLM:

- Gemma 2 billones de paramatros.
- Mistral 7 billones de parámetros.
- Phi3 3 billones de parámetro.
- Llamma2 7 billones de parámetros.

El equipo utilizado para evaluar los modelos contaba con la configuración de *hardware* especificada en la Figura 2 Esta configuración se consideró adecuada para las demandas computacionales de las pruebas.

Figura 2. *Hardware utilizado*.

Fuente. *Elaboración propia (2025).*

Se estableció una relación entre la complejidad de los *prompts* (medida en los índices de Fernández-Huerta e INFLESZ). Para ello, se consideraron tres niveles de complejidad: "fácil", "normal" y "difícil", según la escala de los índices mencionados.

Configuración de los modelos LLM.
Con el fin de garantizar la comparabilidad de los resultados, se mantuvo una configuración uniforme para todos los LLM, ver **Código 2**:

Código 2. *Configuración LLM en Python.*

```
outputs = pipeline(
        prompt,
        max_new_token = 350,
        do_sample = True,
        temperature = 0.7,
        top_k = 50,
        top_p = 0.95.)
```

***Prompts* usados en la evaluación.**
Para los *prompts* que se les inserto a los modelos LLM para esta prueba, sus índices de complejidad se basaron en 3 niveles, y 4 tipos de preguntas para cada nivel correspondiente:

Fácil: Texto fácilmente entendible y que no requiere de estudios técnicos para ser respondido. En la Tabla 5.

Tabla 5. *Prompts con dificultad fácil*.

Prompt	Escala INFLESZ	Escala Fernández Huerta
¿Cuál es la capital de Francia?	86.02	90.26
¿Quién escribió "Cien años de soledad"?	70.45	75.26
¿Cuántos días tiene un año bisiesto?	70.45	75.26
¿Qué es la fotosíntesis?	68.85	73.78

Fuente. *Elaboración propia (2025).*

Normal: texto que es entendible y que requiere de estudios previos para ser respondido, un estudiante de preparatoria a universidad podría responder. En la tabla Tabla 6.

Tabla 6. *Prompts con dificultad normal*.

Prompt	Escala INFLESZ	Escala Fernandez Huerta
Explica cómo funciona la teoría de la relatividad en términos simples.	56.57	61.75
¿Cuáles son los principales argumentos a favor y en contra del cambio climático?	74.74	79.19
Describe el impacto del renacimiento en el arte europeo.	59.74	64.87
¿Cómo se resuelve un sistema de ecuaciones lineales usando matrices?	45.09	50.72

Fuente. *Elaboración propia (2025).*

Difícil: texto que es difícil de entender y que requiere de conocimientos especializados para ser respondido, un estudiante de universidad a posgrado podría responder. En la Tabla 7.

Tabla 7. *Prompts con dificultad difícil*.

Prompt	Escala INFLESZ	Escala Fernandez Huerta
Analiza la influencia de la filosofía existencialista en la literatura contemporánea.	0	0
Discute las implicaciones filosóficas y éticas de la tecnología de edición	23.4	29.27

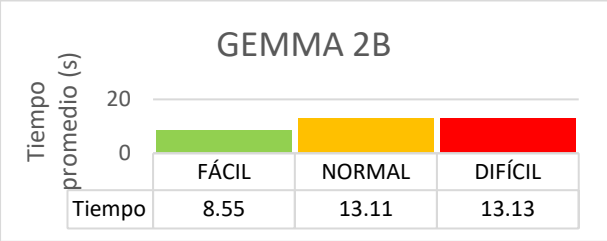
genética en la evolución humana.		
Analiza cómo la teoría cuántica de campos puede reconciliarse con la teoría de la relatividad general.	27.3	33.02
Desarrolla un marco teórico para la integración de la inteligencia artificial en la toma de decisiones éticas en sistemas autónomos.	24.86	30.44

Fuente. *Elaboración propia (2025).*

Medición de los tiempos de ejecución.

Los tiempos de ejecución, medidos en segundos, se obtuvieron para cada modelo evaluado, considerando la complejidad de los *prompts* según los índices de Fernández-Huerta e INFLESZ. Los resultados para el modelo GEMMA 2B se presentan en la Figura 3.

Figura 3. *Tiempo de ejecución de GEMMA 2B.*



Fuente. *Elaboración propia (2025).*

Y en la Tabla 8, se muestra el resultado promedio de los índices para cada nivel de *prompts* en GEMMA 2B.

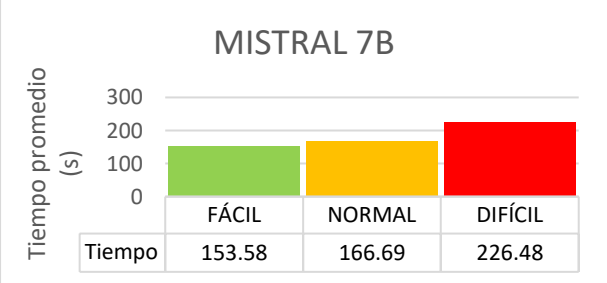
Tabla 8. *Promedio de los índices GEMMA 2B.*

ÍNDICES	FÁCIL	NORMAL	DIFÍCIL
Índice Fernández-Huerta promedio	78.64	64.1325	33.145
Índice INFLESZ promedio	73.9425	59.035	27.1671

Fuente. *Elaboración propia (2025).*

Los resultados obtenidos para el modelo MISTRAL 7B se presentan en la Figura 4.

Figura 4. *Tiempo de ejecución de MISTRAL 7B.*



Fuente. *Elaboración propia (2025).*

Y en la Tabla 9, se muestra el resultado promedio de los índices para cada nivel de *prompt* en MISTRAL.

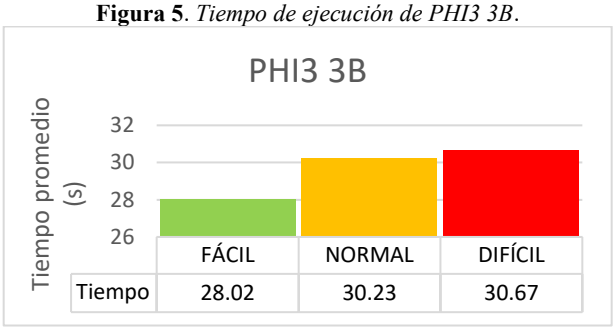
Tabla 9. *Promedio de los índices en MISTRAL 7B.*

ÍNDICES	FÁCIL	NORMAL	DIFÍCIL
Índice Fernández-Huerta promedio	73.4975	56.42	23.1825
Índice INFLESZ promedio	68.8925	51.2875	18.89

Fuente. *Elaboración propia (2025).*

Los resultados obtenidos para el modelo PHI3 3B se presentan en la Figura 5**.**

Figura 5. *Tiempo de ejecución de PHI3 3B.*



Fuente. *Elaboración propia (2025).*

Y en la Tabla 10, se muestra el resultado promedio de los índices para cada nivel de *prompt* en PHI3.

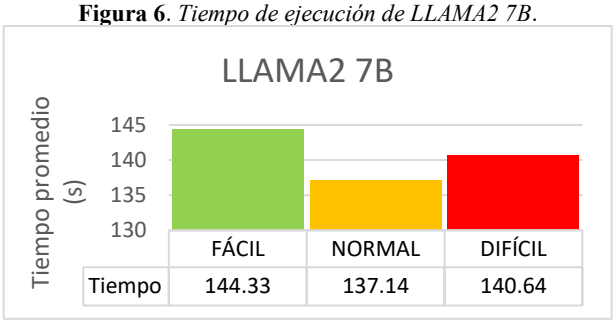
Tabla 10. *Promedio de los índices PHI3 3B.*

ÍNDICES	FÁCIL	NORMAL	DIFÍCIL
Índice Fernández-Huerta promedio	73.4975	56.42	23.1825
Índice INFLESZ promedio	68.8925	51.2875	18.89

Fuente. *Elaboración propia (2025).*

Los resultados obtenidos para el modelo LLAMA2 7B se presentan en la Figura 6**.**

Figura 6. *Tiempo de ejecución de LLAMA2 7B.*



Fuente. *Elaboración propia (2025).*

Y en la Tabla 11, se muestra el resultado promedio de los índices para cada nivel de *prompt* en LLAMA2.

Tabla 11. *Promedio de los índices LLAMA2 7B.*

ÍNDICES	FÁCIL	NORMAL	DIFÍCIL

IPSUMTEC
ISSN: 2594 - 2905

Índice Fernández-Huerta promedio	100	68.95	44.7967
Índice INFLESZ promedio	98.875	65.2825	39.3633

Fuente. *Elaboración propia (2025).*

Normalización del corpus de documentos.
Una vez definido el modelo de lenguaje, se inició la fase de entrenamiento. Para ello, se recopiló un corpus de documentos del Instituto Tecnológico de Zacatepec, como se muestra en la Figura 7. Estos documentos servirán como base para que el modelo aprenda a generar respuestas coherentes y relevantes basados en información real.

Figura 7. *Recopilación de documentos*.

Fuente. *Elaboración propia (2025).*

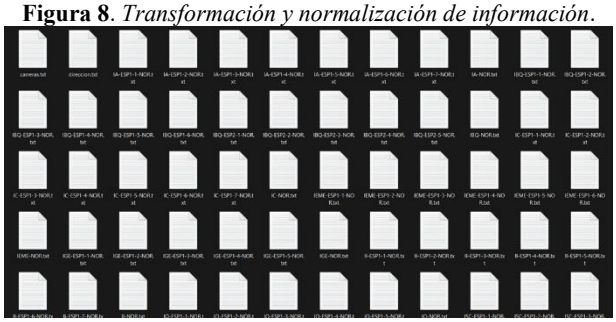
Con el propósito de optimizar el uso del modelo GEMMA 2B, se llevó a cabo un proceso de limpieza de los documentos. Esta etapa consistió en eliminar caracteres especiales, puntuación y *stopwords*, con el fin de proporcionar al modelo un contexto más limpio y relevante **Código 3**.

Código 3. *Limpieza de texto en Python.*

```
def normalizeText(text):
    # Reemplazar caracteres
     text = text.replace('/', '
     ').replace('-', ' ').replace('_', '')

    # Remover URLs
    text =
    re.sub(r'http\S+|www\S+|https\S+',
    '', text, flags=re.MULTILINE)

    # Remover puntuaciones
    text = text.
    translate(str.maketrans('', '',
    string.punctuation))
    text = text.lower()

    # Convertir a minúsculas
    text = unidecode(text)

    # Remover tildes
    words = nltk.word_tokenize(text,
    language='spanish')

    # Tokenizar el texto
    stop_words =
    set(stopwords.words('spanish'))
    #Remover stop words
    words = [word for word in words if word
    not in stop_words]
  return ' '.join(words)
```

Una vez normalizados y convertidos a formato TXT como se observa en la Figura 8, los documentos estuvieron listos para ser utilizados como entrada en el modelo GEMMA 2B. Este formato textual plano es ideal para el entrenamiento de modelos de lenguaje.

Figura 8. *Transformación y normalización de información*.

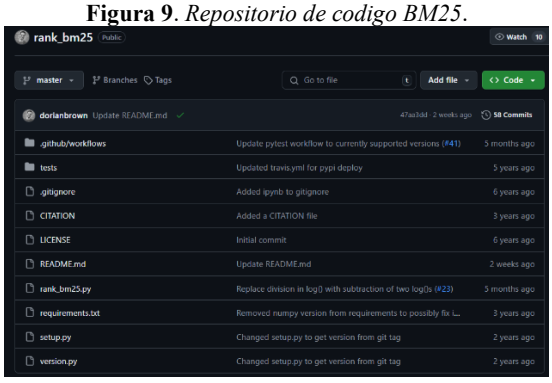


Fuente. *Elaboración propia (2025).*

Integración del algoritmo de búsqueda BM25.
Se utilizó un repositorio de código disponible en GitHub, alojado: https://github.com/dorianbrown/rank_bm25.git

Como base para implementar el algoritmo, el cual se muestra en la Figura 9. La integración del código proporcionado en el repositorio se realizó de manera cuidadosa para garantizar su compatibilidad con el sistema en desarrollo.

Figura 9. *Repositorio de codigo BM25*.



Fuente. *GitHub rank_bm25 (Dorian Brown, 2024).*

Ejecución BM25 integrado con GEMMA 2B.
Para evaluar la funcionalidad del código, se realizaron pruebas prácticas similares a las evaluaciones de modelos LLM, pero enfocadas exclusivamente en GEMMA 2B con BM25 con preguntas del ITZ.

Fácil: Su índice de Fernández-Huerta como de INFLESZ está entre 100-70 Tabla 12.

Tabla 12. *Prompst fáciles con BM25*.

Prompt	Escala INFLESZ	Escala Fernández Huerta
¿Que se ve en bioquímica?	100	100
¿Qué es una base de datos?	100	100
¿Qué se hace en el diseño estructural?	84.14	88.27
¿Qué es un sistema esbelto?	93.74	89.7

Fuente. *Elaboración propia (2025)*.

Normal: Su índice de Fernández-Huerta como de INFLESZ está entre 70-40 Tabla 13.

Tabla 13. *Prompst normales con BM25*.

Prompt	Escala INFLESZ	Escala Fernández Huerta
¿Qué es el emprendimiento turístico?	52.32	57.74
Dame la dirección del tecnológico.	52.32	57.74
¿Qué es la electromecánica?	47.09	52.76
Dame el programa educativo de la licenciatura en turismo	40.3	41.18

Fuente. *Elaboración propia (2025).*

Difícil: Su índice de Fernández-Huerta como de INFLESZ está entre 40-0 Tabla 14.

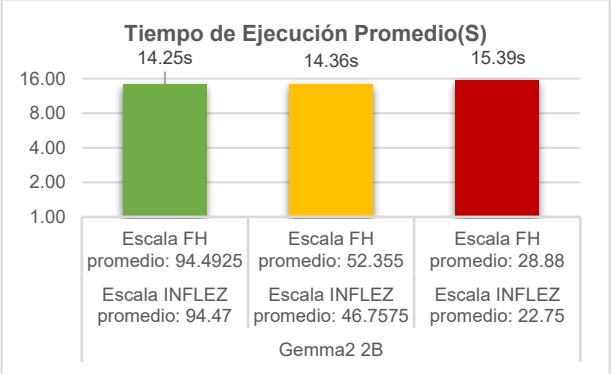
Tabla 14. *Prompst difíciles con BM25*.

Prompt	Escala INFLESZ	Escala Fernández Huerta
¿Qué es la ingeniería en sistemas computacionales?	39.63	45.41
Dame el programa educativo de ingeniería en administración, de la especialidad administración negocios	6.94	13.58
Dame el programa educativo de ingeniería en electromecánica	4.15	11.18
Para qué sirve la programación web y cuáles son sus aplicaciones en el ámbito de la ingeniería en sistemas computacionales	32.28	40.35

Fuente. *Elaboración propia (2025).*

Los tiempos de ejecución, medidos en segundos, se registraron para GEMMA 2B durante la evaluación basada en la lectura del documento relevante seleccionado mediante BM25. Además de considerar la complejidad de los prompts utilizando los índices de Fernández-Huerta e INFLESZ. Los resultados obtenidos, fueron promediados, obteniendo los siguientes valores Figura 10.

Figura 10. *Tiempos de ejecución promedio GEMMA 2B con BM25*.



Fuente. *Elaboración propia (2025).*

Integración de API

Para permitir una interacción fluida con el modelo GEMMA 2B, se diseñó una API RESTful utilizando el framework Flask. Esta API expone un *endpoint* que acepta peticiones POST con un campo "*prompt*". Al recibir una petición, el modelo procesa el *prompt* y genera una respuesta en formato JSON, la cual es enviada de vuelta al cliente. Esto se puede observar en el **Código 4**.

Código 4. *Endpoint para peticiones al modelo en Python.*

```
@app.route('/query', methods=['POST'])
def query():
        data = request.json
        query = data.get('query')

        if not query:
        return jsonify({'error': 'No query
provided'}), 400
```

Después de este proceso, la consulta es normalizada y tokenizada para luego calcular los puntajes de relevancia con el algoritmo BM25. Este proceso se puede observar en el **Código 5**.

Código 5. *Recuperación de documentos relevantes con BM25 en Python.*

```
normalized_query = normalize_text(query)
tokenized_query =
normalized_query.split()
scores =
bm25.get_scores(tokenized_query)

documents_with_scores =
list(zip(filenames, scores))
```

```
documents_with_scores.sort(key=lambda x:
x[1], reverse=True)
n_top_documents = 1
top_documents =
documents_with_scores[:n_top_documents]
```

Una vez que se ha seleccionado el documento relevante, se utiliza el modelo GEMMA 2B para generar un resumen, cómo se observa en el *Código 6*.

Código 6. Generación del resumen con GEMMA 2B en Python.

```
messages = [
   {"user",
    "content": f"Resumen del document
    sobre {query}: {doc_content}"}]

prompt = tokenizer.apply_chat_template(
       messages,
       tokenize = False,
       add_generation_prompt = True)

outputs = gemma_pipeline(
      prompt, max_new_tokens = 256,
      add_special_tokens = True,
      do_sample = True,
      temperatura = 0.7,
      top_k = 50, top_p = 0.95)
summary =
outputs[0]["generated_text"][len(prompt)
:]
```

Resultados.

Para evaluar y demostrar las capacidades del modelo GEMMA 2B, se desarrolló una interfaz de usuario sencilla utilizando las tecnologías web *HTML*, *CSS* y *JavaScript*. Esta interfaz, que simula una conversación en tiempo real, permite a los usuarios enviar mensajes de texto al modelo y recibir respuestas generadas de forma dinámica. Se puede observar esta interfaz en la Figura 11.

Figura 11. *Interacción del modelo en la interfaz del chat*.

Fuente. *Elaboración propia (2025).*

DISCUSIÓN Y ANÁLISIS DE RESULTADOS

La combinación de modelos de lenguaje de gran escala (LLM), como GEMMA 2B, con algoritmos de recuperación de información como BM25, demostró ser altamente eficaz para resolver preguntas cuya estructura no sigue un orden secuencial lineal. Este enfoque permitió identificar fragmentos de texto relevantes dispersos en los documentos base, lo que mejoró la capacidad del chatbot para ofrecer respuestas coherentes y precisas. Que coincide con lo reportado con Hugo (2009) [14] donde profundiza en las bases probabilísticas que hacen de BM25 un marco de relevancia robusto para la recuperación de documentos, lo cual respalda teóricamente los resultados observados en esta investigación.

Los parámetros desempeñan un papel crucial en el rendimiento de un modelo de lenguaje (LLM). Como lo explica Belcic y Stryker [8], En el desarrollo de este proyecto se comprobó que, si bien un número mayor de parámetros puede mejorar el rendimiento, el costo computacional asociado crece de forma significativa. Por ello, resulta fundamental encontrar un equilibrio entre el tamaño del modelo y los recursos disponibles, optimizando el número de parámetros sin sacrificar la calidad de los resultados.

Un hallazgo relevante de esta investigación fue la disminución significativa en la gran oportunidad de disminuir la carga administrativa derivada de la implementación del chatbot en la automatización de respuestas a preguntas frecuentes, el direccionamiento inteligente a secciones relevantes del sistema, y la capacidad del modelo para mantener conversaciones contextualizadas, permitieron reducir la necesidad de intervención humana en tareas repetitivas. Esta observación coincide con los hallazgos de Misischia et al. (2022) [10], en su estudio, los autores señalan que los chatbots desempeñan funciones clave como la resolución de problemas y la personalización de la atención, lo que contribuye de manera directa a elevar la calidad del servicio y a optimizar los recursos humanos.

Un tercer aspecto que se identificó durante el desarrollo del chatbot fue el potencial de la inteligencia artificial generativa para ampliar las capacidades del sistema, permitiéndole responder a consultas más complejas y adaptarse dinámicamente al estilo conversacional del usuario. Esta evolución tecnológica ha demostrado ser especialmente útil para mejorar la experiencia de autoservicio de los usuarios, al reducir la necesidad de programar manualmente respuestas predefinidas y, en su lugar, generar respuestas contextualizadas basadas en una base de conocimientos institucional. De acuerdo con IBM (2023) [15], los chatbots de próxima generación no solo replican respuestas humanas, sino que pueden generar contenido nuevo, interpretar lenguaje natural con mayor precisión y responder con empatía, lo que mejora significativamente la calidad de la interacción. Este avance ha permitido automatizar procesos

administrativos previamente manejados de forma manual, optimizando el tiempo del personal y mejorando la eficiencia general del servicio. Además, la capacidad del chatbot para comprender y atender una mayor variedad de preguntas ha contribuido a ampliar su utilidad como herramienta de atención institucional.

CONCLUSIONES

El chatbot del Instituto Tecnológico de Zacatepec representa un avance en la aplicación de la inteligencia artificial en el ámbito académico. Utilizando el modelo GEMMA 2B y el algoritmo BM25como método de recuperación de datos, mejora la interacción con los usuarios y reduce la carga administrativa. La evaluación con índices de complejidad lingüística ha optimizado su rendimiento, validando su eficacia en la automatización de procesos. Este proyecto establece las bases para futuras aplicaciones de IA en la educación.

En relación con estos hallazgos. Labadze et al. (2023) [5] realizaron una revisión sistemática de 67 estudios sobre chatbots educativos, concluyendo que estos ofrecen beneficios como asistencia personalizada al estudiante, apoyo en tareas administrativas y desarrollo de habilidades lo cual se demostró en este proyecto con su potencial en la reducción de la carga de trabajo. Por otro lado, Misischia et al. (2022) [10], en el marco de la 13ª Conferencia Internacional sobre Sistemas Ambientales, Redes y Tecnologías (ANT 2022), analizaron el impacto de los chatbots en la calidad del servicio al cliente, concluyendo que estos mejoran considerablemente la eficiencia, la inmediatez y la personalización en la atención al cliente en plataformas digitales.

AGRADECIMIENTOS

Expresamos nuestro más sincero agradecimiento al maestro Luis José Muñiz Rascado por su guía y mentoría a lo largo del desarrollo de este proyecto. Su conocimiento en inteligencia artificial generativa y modelos de lenguaje de gran escala fue fundamental para la comprensión y aplicación de estos conceptos en nuestro trabajo. Su orientación y consejos nos permitieron enfrentar los desafíos con una perspectiva más clara y fundamentada.
Además, extendemos nuestro reconocimiento al profesor Eugenio César Velázquez Santana por su constante apoyo y colaboración en el proceso de desarrollo del proyecto. Su ayuda fue clave no solo en la estructuración y mejora del trabajo, sino también en las gestiones necesarias para su publicación. Su disposición y compromiso fueron esenciales para lograr los objetivos planteados.

BIBLIOGRAFÍA

[1] Caldarin J, Jaf A, McGarry B. El papel de los chatbots en la Industria 4.0 [Internet]. [consultado 20 de julio de 2025]. Disponible en: https://www.researchgate.net/publication/373718439_The_role_of_Chatbots_in_Industry_40
[2] Fernández Huerta A. Lecturabilidad [Internet]. Madrid: Legible; [consultado 20 de julio de 2025]. Disponible en: https://legible.es/blog/lecturabilidad-fernandez-huerta/
[3] Steen H, Urban E. Configuración de la puntuación de relevancia BM25 [Internet]. Microsoft; 2024 [consultado 20 de julio de 2025]. Disponible en: https://learn.microsoft.com/es-es/azure/search/index-ranking-similarity
[4] Reznik L. General principles and purposes of computational intelligence. Systems Science and Cybernetics [Internet]. 2009 [consultado 20 de julio de 2025]. Disponible en: https://www.eolss.net/sample-chapters/c02/E6-46-04-01.pdf
[5] Labadze L, et al. A systematic review of educational chatbots: Applications and effectiveness. Computers and Education: Artificial Intelligence. [consultado 20 de julio de 2025] Disponible en: https://educationaltechnologyjournal.springeropen.com/articles/10.1186/s41239-023-00426-1?utm_source=chatgpt.com
[6] Szigriszt Pazos F. Sistemas predictivos de legibilidad del mensaje escrito: fórmula de perspicuidad [Internet]. 1993 [consultado 20 de julio de 2025]. Disponible en: https://webs.ucm.es/BUCM/tesis//19911996/S/3/S3019601.pdf
[7] Bakhshinategh B, Zaiane OR, ElAtia S, Ipperciel D. Educational data mining applications and tasks: A survey of the last 10 years. Educ Inf Technol [Internet]. 2018 [consultado 20 de julio de 2025];23(1):537–553. Disponible en: https://doi.org/10.1007/s10639-017-9616-z
[8] Belcic I, Stryker C. ¿Qué son los parámetros del modelo? [Internet]. IBM; 2024 [citado 2025 Jul 27]. Disponible en: https://www.ibm.com/mx-es/think/topics/model-parameters
[9] Barrio I. El programa Inflesz [Internet]. Legibilidad.com: una web sobre el análisis de la legibilidad de textos escritos en español; 2015 ene 11 [consultado 2025 jul 20]. Disponible en: https://inflesz.com/
[10] Misischia CV, Poecze F, Strauss C. Chatbots in customer service: Their relevance and impact on service quality. In: Proceedings of the 13th International Conference on Ambient Systems, Networks and Technologies (ANT); 2022 Mar 22–25; Porto, Portugal. Procedia Computer Science. 2022;201:1177–1184. Disponible en: https://www.sciencedirect.com/science/article/pii/S1877050922004689
[11] Fayyad U, Stolorz P. Data mining and KDD: Promise and challenges. Future Gener Comput Syst [Internet]. 1997 [consultado 20 de julio de 2025];13(2–3):99–115. Disponible en: https://doi.org/10.1016/S0167-739X(97)00015-0
[12] Brandtzaeg PB, Folstad A. Why people use chatbots: authors' version. SINTEF [Internet]. 2017 [consultado 20 de julio de 2025]. Disponible en: https://sintef.brage.unit.no/sintef-

xmlui/bitstream/handle/11250/2468333/Brandtzaeg_Folstad_why+people+use+chatbots_authors+version.pdf
[13] Brown T. Language models are few-shot learners. NeurIPS [Internet]. 2020 [consultado 20 de julio de 2025]. Disponible en: https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf
[14] Hugo SE. The Probabilistic Relevance Framework: BM25 and Beyond [Internet]. 2009 [consultado 20 de julio de 2025]. Disponible en: https://www.researchgate.net/publication/220613776_The_Probabilistic_Relevance_Framework_BM25_and_Beyond
[15] IBM. Chatbots [Internet]. IBM; 2023 [consultado 20 de julio de 2025]. Disponible en: https://www.ibm.com/mx-es/topics/chatbots

Tabla de Roles de Contribución

Rol	Autor (es)
Administración del proyecto	Luis José Muñiz Rascado
Recursos	Eugenio César Velázquez Santana
Software	Carlos Pérez Calderon (igual) Luis Ángel Vázquez Carrillo (igual)
Validación	Noé Alejandro Castro-Sánchez
Escritura - Preparación del borrador original	Carlos Pérez Calderon (igual) Luis Ángel Vázquez Carrillo (igual)