

UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO
FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA, INFORMÁTICA Y
MECÁNICA
ESCUELA PROFESIONAL DE INGENIERÍA INFORMÁTICA Y DE SISTEMAS



TESIS

**MODELO GENERADOR DE RECETAS PERSONALIZADAS EN
ESPAÑOL DE LA GASTRONOMÍA PERUANA USANDO DEEP
LEARNING**

PRESENTADO POR:

Br. YUVISA QUISPE PALOMINO

**PARA OPTAR AL TÍTULO PROFESIONAL DE
INGENIERO INFORMÁTICO Y DE SISTEMAS**

ASESOR:

Dr. LAURO ENCISO RODAS

CO-ASESOR:

M.Sc. PAUL DANY FLORES ATAUCHI

**CUSCO - PERÚ
2025**

INFORME DE ORIGINALIDAD

(Aprobado por Resolución Nro.CU-303-2020-UNSAAC)

El que suscribe, **Asesor** del trabajo de investigación/tesis titulada:

Modelo generador de recetas personalizadas en español de la gastronomía peruana usando deep learning

Presentado por: Yuviga Quispe Palomino DNI N° 48607343

presentado por: DNI N°:

Para optar el título profesional/grado académico de Ingeniero Informático y de Sistemas

Informo que el trabajo de investigación ha sido sometido a revisión por 02 veces, mediante el Software Antiplagio, conforme al Art. 6° del **Reglamento para Uso de Sistema Antiplagio de la UNSAAC** y de la evaluación de originalidad se tiene un porcentaje de 2%.

Evaluación y acciones del reporte de coincidencia para trabajos de investigación conducentes a grado académico o título profesional, tesis

Porcentaje	Evaluación y Acciones	Marque con una (X)
Del 1 al 10%	No se considera plagio.	X
Del 11 al 30 %	Devolver al usuario para las correcciones.	
Mayor a 31%	El responsable de la revisión del documento emite un informe al inmediato jerárquico, quien a su vez eleva el informe a la autoridad académica para que tome las acciones correspondientes. Sin perjuicio de las sanciones administrativas que correspondan de acuerdo a Ley.	

Por tanto, en mi condición de asesor, firmo el presente informe en señal de conformidad y adjunto las primeras páginas del reporte del Sistema Antiplagio.

Cusco, 27 de Junio de 2025



Firma

Post firma Lauro Enciso Rodas

Nro. de DNI 23853228

ORCID del Asesor 0000 - 0001 - 6266 - 9838

Se adjunta:

1. Reporte generado por el Sistema Antiplagio.
2. Enlace del Reporte Generado por el Sistema Antiplagio: oid: 272592470168604

Yuvisa Quispe

Tesis-final Yuvisa unio 2025.pdf

 Universidad Nacional San Antonio Abad del Cusco

Detalles del documento

Identificador de la entrega

trn:oid:::27259:470168604

Fecha de entrega

27 jun 2025, 7:08 a.m. GMT-5

Fecha de descarga

27 jun 2025, 7:20 a.m. GMT-5

Nombre de archivo

Tesis-final Yuvisa unio 2025.pdf

Tamaño de archivo

3.4 MB

116 Páginas

17.396 Palabras

97.263 Caracteres

2% Similitud general

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para ca...

Filtrado desde el informe

- ▶ Bibliografía
- ▶ Texto citado
- ▶ Coincidencias menores (menos de 15 palabras)

Fuentes principales

- 1%  Fuentes de Internet
- 0%  Publicaciones
- 2%  Trabajos entregados (trabajos del estudiante)

Marcas de integridad

N.º de alertas de integridad para revisión

No se han detectado manipulaciones de texto sospechosas.

Los algoritmos de nuestro sistema analizan un documento en profundidad para buscar inconsistencias que permitirían distinguirlo de una entrega normal. Si advertimos algo extraño, lo marcamos como una alerta para que pueda revisarlo.

Una marca de alerta no es necesariamente un indicador de problemas. Sin embargo, recomendamos que preste atención y la revise.

Dedicatoria

Dedico este trabajo de investigación a toda mi familia, y en especial a mi querida mamá, Rosa, por tu amor incondicional, tu cuidado constante y el apoyo que me brindaste en cada etapa de mi vida. Tus palabras y enseñanzas siguen acompañándome y viven en mí cada día.

A mi querido hermano Benjamín, a quien recordamos con profundo cariño. Solo Dios sabe por qué hace las cosas, pero sabemos que tu presencia sigue con nosotros, guiándonos y dándonos fuerza.

A mi papá, Samuel, por tu amor incondicional, por confiar siempre en mí y por seguir cuidándome como cuando era una niña. Gracias por tu fortaleza y por estar presente en cada paso que doy.

A mis hermanos Hilary, Anali, Josue y Andrea, con quienes crecí compartiendo risas, aprendizajes y momentos que marcaron mi vida. Gracias por ser un pilar constante, por su cariño incondicional, por el apoyo en los momentos difíciles y por siempre confiar en mí.

Yuvisa Quispe Palomino

Agradecimientos

Expreso mi más sincero agradecimiento a los docentes de la Escuela Profesional de Ingeniería Informática y de Sistemas por su valiosa formación académica y el acompañamiento constante a lo largo de mi carrera universitaria.

De manera especial, agradezco al Dr. Lauro Enciso Rodas, asesor de este trabajo de tesis, por su orientación y apoyo durante el desarrollo de esta investigación.

Asimismo, extiendo mi agradecimiento al M.Sc. Paul Dany Flores Atauchi, co-asesor de esta tesis, por su guía no solo en el ámbito académico. Sus palabras de aliento y motivación fueron fundamentales para mantenerme firme en momentos difíciles y continuar creciendo tanto a nivel profesional como personal.

Agradezco también a mis dictaminantes, la Dra. Zonia y el M.Sc. Tany, por su acompañamiento en esta etapa final. Aunque el proceso fue extenso por diversas razones, valoro profundamente el tiempo que dedicaron a la lectura de mi trabajo y las observaciones realizadas, las cuales me permitieron fortalecer mis habilidades en investigación y redacción académica.

Agradezco también al M.Sc. Lino Prisciliano Flores Pacheco, por su confianza y apoyo incondicional, expreso mi más sincero reconocimiento.

Finalmente, agradezco con profundo cariño a mi familia y amigos, por su apoyo constante, su paciencia y sus palabras de ánimo a lo largo de este proceso.

Resumen

La generación de lenguaje natural (NLG) es una subárea del procesamiento de lenguaje natural orientada a la producción automática de texto comprensible para humanos. Entre sus múltiples aplicaciones, destaca la generación automática de recetas culinarias, una tarea que requiere no solo conocimientos técnicos, sino también creatividad, sensibilidad cultural y tiempo. Aunque estos procesos resultan complejos para las personas, los modelos de lenguaje permiten automatizarlos de forma eficiente, escalable y personalizada. Modelos como Chef Watson y adaptaciones de GPT-2 entrenadas con el corpus Recipe1M+ han mostrado resultados prometedores en inglés y en contextos internacionales. Sin embargo, la gastronomía peruana ha recibido poca atención en estos sistemas, la mayoría de los cuales fueron diseñados originalmente para el idioma inglés, sin considerar las particularidades culturales y lingüísticas del español. Este estudio presenta un modelo generador de recetas personalizadas en español, centrado en la gastronomía peruana, basado en GPT-2 Spanish, una versión preentrenada con 11,5 GB de texto en español. El modelo fue ajustado mediante fine-tuning utilizando un corpus obtenido mediante web scraping de fuentes confiables de recetas peruanas, e incorporó un módulo de reconocimiento de entidades nombradas (NER) para mejorar la identificación de ingredientes. Se evaluaron cuatro configuraciones: (1) GPT-2 Base, (2) GPT-2 Base + NER, (3) GPT-2 + Fine-tuning, y (4) GPT-2 + Fine-tuning + NER. Los resultados, evaluados mediante las métricas BLEU, BERTScore y COMET, muestran que la variante GPT-2 + Fine-tuning obtuvo el mejor desempeño general, generando recetas más coherentes,

culturalmente auténticas y lingüísticamente adecuadas.

Palabras clave: Gastronomía peruana, Generación de Recetas, Fine-tuning, GPT-2

Abstract

Natural Language Generation (NLG) is a subfield of Natural Language Processing focused on the automatic production of human-readable text. Among its many applications, automatic recipe generation stands out as a task that requires not only technical knowledge but also creativity, cultural sensitivity, and time. Although such processes are complex for humans, language models make them efficient, scalable, and personalized. Models like Chef Watson and adaptations of GPT-2 trained on the Recipe1M+ corpus have shown promising results in English and international contexts. However, Peruvian cuisine has received little attention in these systems, most of which are originally designed for English and do not consider the cultural and linguistic nuances of Spanish. This study presents a personalized recipe generation model in Spanish, focused on Peruvian cuisine, based on GPT-2 Spanish—a version pretrained on 11.5 GB of Spanish text. The model was fine-tuned using a corpus obtained through web scraping from reliable Peruvian recipe sources, and a Named Entity Recognition (NER) module was incorporated to enhance ingredient identification. Four configurations were evaluated: (1) GPT-2 Base, (2) GPT-2 Base + NER, (3) GPT-2 + Fine-tuning, and (4) GPT-2 + Fine-tuning + NER. Results, assessed using BLEU, BERTScore, and COMET metrics, show that the GPT-2 + Fine-tuning variant achieved the best overall performance, generating recipes that are more coherent, culturally authentic, and linguistically appropriate.

Keywords: Peruvian gastronomy, Recipe generator, Fine-tuning, GPT-2

Acrónimos

- **GPT:** Generative Pre-trained Transformer

- **NER:** Named Entity Recognition

- **CNN:** Convolutional Neural Network

- **ROUGE:** Recall-Oriented Understudy for Gisting Evaluation

- **BLEU:** Bilingual Evaluation Understudy

- **BERTScore:** Bidirectional Encoder Representations from Transformers Score

- **COMET:** Crosslingual Optimized Metric for Evaluation of Translation

- **NLG:** Natural Language Generation

- **LSTM:** Long Short-Term Memory

- **RNN:** Recurrent Neural Network

- **ML:** Machine Learning

- **DL:** Deep Learning

- **AI:** Artificial Intelligence

- **ReLU:** Multi Layer Perceptron

- **NLP:** Natural Language Processing

- **MCTS:** Monte Carlo Tree Search

- **KNN:** K-Nearest Neighbours

- **CBOW:** Continuous Bag of Words

- **URL:** Uniform Resource Locator

- **HTML:** HyperText Markup Language

- **JSON:** JavaScript Object Notation

- **LLM:** Large Language Model

- **IEEE** Institute of Electrical and Electronics Engineers

- **ACM** Association for Computing Machinery

Índice General

Indice de Figuras	XV
--------------------------	-----------

Indice de Tablas	XVII
-------------------------	-------------

1. Aspectos Generales	1
------------------------------	----------

1.1. Planteamiento del problema	1
1.1.1. Formulación del Problema	3
1.1.2. Antecedentes	3
1.1.3. Justificación	8
1.1.4. Objetivos	9
1.1.5. Objetivo General	9
1.1.6. Alcances y Limitaciones	9
1.1.7. Metodología	10
1.1.8. Contribuciones	13

2. Marco Teórico	14
-------------------------	-----------

2.1. Inteligencia artificial(IA)	14
2.2. Generación de lenguaje natural (NLG)	14
2.2.1. Métodos generales de NLG	15
2.3. Grandes Modelos de lenguaje (LLM)	15
2.3.1. Ajuste fino	18

2.4. Deep Learning (DL)	20
2.5. Procesamiento de lenguaje natural (PLN)	21
2.5.1. Word Embeddings	21
2.5.2. Pre-trained word embeddings	21
2.6. Transformer	22
2.6.1. Stack Encoder y Decoder	23
2.6.2. Attention	23
2.7. GPT Framework	25
2.7.1. Unsupervised Pre-Training	25
2.7.2. Supervised Fine-Tuning	26
2.8. Gastronomía Peruana	27
2.8.1. Cocina peruana como regalo culinario para el mundo	28
2.8.2. Cocina peruana como patrimonio cultural, identidad cultural y turismo	29
2.8.3. Cocina Peruana como patrimonio cultural 2007	29
2.8.4. Gastronomía del Sur Peruano	29
2.8.5. Gastronomía del Norte Peruano	30
2.8.6. Gastronomía de la selva peruana	31
3. Desarrollo del proyecto	32
3.1. Construcción del dataset	32
3.1.1. Web Scraping	32
3.1.2. Proceso web scraping	33
3.1.3. Proceso de extracción de recetas	33
3.1.4. Resumen y resultados de los sitios web seleccionados	35

3.1.5. Desafíos encontrados durante el proceso de web scraping	35
3.1.6. Flujo del proceso de recopilación automatizada de recetas	36
3.1.7. Formato de almacenamiento de las recetas tras el proceso de web scraping	38
3.2. Preparación del dataset de gastronomía peruana	39
3.2.1. Proceso de limpieza manual de los archivos JSON	39
3.3. Exploración del dataset	40
3.4. Preprocesamiento del dataset	43
3.4.1. Limpieza automática de datos	44
3.5. Personalización del modelo NER para la extracción de entidades	46
3.5.1. Anotación de los ingredientes	47
3.5.2. Entrenamiento del modelo NER personalizado	48
3.6. Diseño e implementación del modelo	50
3.6.1. Etiquetas especiales	51
3.6.2. Selección del modelo	52
3.6.3. Partición del dataset	52
3.6.4. Tokenización	53
3.6.5. Fine Tuning	54
3.6.6. Implementación del modelo y entrenamiento	54
3.6.7. Recursos para el entrenamiento	54
3.6.8. Entrenamiento	55
4. Resultados	56
4.1. Resultados del modelo personalizado de reconocimiento de entidades nom- bradas (NER)	56

4.2. Evaluación del modelo GPT-2 y sus variaciones	58
4.2.1. Formato ejemplo de receta base utilizada para comparar con las recetas generadas	58
4.2.2. Métrica BLEU	60
4.2.3. Métrica BERTScore	60
4.2.4. Métrica COMET	61
4.3. Comparación de Desempeño entre Variantes del Modelo GPT-2 en la Generación de Recetas	62
4.4. Evaluación del modelo con usuarios	64
4.4.1. Resultados de la evaluación obtenida	65
4.4.2. Resumen general de resultados	66
5. Análisis de resultados	67
5.1. Resultados offline y con usuarios	67
5.1.1. Análisis comparativo del desempeño de los modelos	67
5.1.2. Análisis de resultados de la evaluación con usuarios	68
Conclusiones	73
Recomendaciones	74
Trabajos futuros	76
Bibliografía	76
Anexo A – Recetas generadas por el modelo GPT2 y variaciones	79
Anexo B – Recetas utilizadas en el formulario de evaluación con usuarios	88

Índice de Figuras

1.1. <i>Etapas de la metodología propuesta</i>	11
2.1. <i>Tipos de modelado de lenguaje</i>	17
2.2. <i>Trabajos publicados en el tiempo</i>	18
2.3. <i>LLM fine-tuning</i>	19
2.4. <i>Diagrama de Venn que muestra la relación entre Artificial Intelligence, Machine Learning y Deep Learning</i>	20
2.5. <i>Estructura de una arquitectura transformer</i>	22
2.6. <i>Attetion</i>	24
2.7. <i>Plato peruano ceviche</i>	28
3.1. <i>Flujo del proceso de web scraping para la extracción de datos</i>	37
3.2. <i>Flujo de la función de recopilación</i>	38
3.3. <i>Relación entre la frecuencia de instrucciones y el número de ingredientes</i>	41
3.4. <i>Nube de palabras de ingredientes frecuentes en las recetas del dataset</i>	42
3.5. <i>Nube de palabras de términos frecuentes en las instrucciones de las recetas del dataset</i>	42
3.6. <i>Dataset gastronomía peruana</i>	43
3.7. <i>Dataset resumen</i>	44
3.8. <i>Relación entre la frecuencia de ingredientes y la cantidad de instrucciones en el dataset, despues de la limpieza automática</i>	46

3.9. <i>Anotación NER de la entidad FOOD en listas de ingredientes</i>	48
3.10. <i>Entrenamiento del modelo NER con distintos valores de dropout</i>	50
3.11. <i>Ejemplo de una receta con tokens de control personalizados</i>	52
4.1. <i>Extracción de entidades FOOD a partir de la lista de ingredientes</i>	57
4.2. <i>Comparación visual del desempeño de las variantes del modelo GPT-2 en la generación de recetas, evaluadas mediante las métricas BLEU, BERTScore y COMET</i>	64
5.1. <i>Preferencia de los usuarios por la actividad de cocinar</i>	69
5.2. <i>Motivos por los que los usuarios cocinan o no cocinan</i>	69

Índice de Tablas

3.1. Resumen del número de recetas extraídas por sitio web mediante web scraping	35
3.2. Tokens de control personalizados en GPT-2	51
4.1. Desempeño del modelo NER en la extracción de entidades FOOD	58
4.2. Valores obtenidos con la métrica BLEU para las cuatro variantes del modelo GPT-2	60
4.3. Valores obtenidos con la métrica BERTScore para las cuatro variantes del modelo GPT-2	61
4.4. Valores obtenidos con la métrica COMET para las cuatro variantes del modelo GPT-2	62
4.5. Comparación del desempeño de las variantes del modelo GPT-2 en la generación de recetas, evaluadas mediante las métricas BLEU, BERTScore y COMET	63
4.6. Resultados de la evaluación de usuarios de las 10 recetas	65
4.7. Resumen general de la encuesta	66
4.8. Resumen porcentaje de aciertos	66

Capítulo 1

Aspectos Generales

1.1. Planteamiento del problema

Natural Language Generation (NLG), es una subárea del Natural Language Processing (NLP), cuyo objetivo es producir textos comprensibles para los seres humanos. Esta tecnología se aplica en una amplia variedad de tareas, tales como la generación automática de reportes, respuestas en chatbots, traducción de textos, resumen automático de información, redacción de noticias y creación de contenido creativo.

En el ámbito de las aplicaciones creativas, la generación automática de recetas constituye una tarea que requiere no solo conocimientos culinarios, sino también creatividad, sensibilidad emocional y comprensión cultural. Aunque este proceso puede resultar complejo para las personas, los modelos de lenguaje permiten automatizar la generación de recetas de forma rápida, escalable y personalizada.

Actualmente, existen modelos basados en arquitecturas como GPT-2 y Transformers que han sido utilizados para generar recetas de manera automática. No obstante, la mayoría de estos modelos han sido entrenados en inglés y con datos correspondientes a cocinas como la india, la china u otras tradiciones culinarias internacionales. A pesar del reconocimiento global de la gastronomía peruana, no se han desarrollado modelos que

consideren específicamente esta cocina, ni se dispone de conjuntos de datos en español que recopilen recetas peruanas y reflejen sus particularidades lingüísticas y culturales.

Esta ausencia representa una limitación significativa para el desarrollo de herramientas que valoren y difundan adecuadamente esta tradición culinaria. Si bien existen diversos recursos, como libros de cocina y plataformas digitales que ofrecen recetas peruanas, estos no proporcionan funcionalidades que permitan la personalización de recetas en función de la disponibilidad de ingredientes o de las preferencias de los usuarios, ya sea por restricciones alimentarias o elecciones personales.

A pesar de los avances en gastronomía computacional y en el uso del aprendizaje profundo, la generación personalizada de recetas en español centradas en la gastronomía peruana es un área aún poco explorada. Como se ha mencionado, no se ha identificado un conjunto de datos específico que represente esta cocina. En contraste, existen herramientas en línea capaces de generar recetas para otras cocinas, como la india, pero no para la gastronomía peruana ni en idioma español.

Por lo tanto, se plantea la necesidad de desarrollar un generador de recetas personalizadas en español, centrado en la gastronomía peruana y basado en técnicas de aprendizaje profundo. Este generador deberá ser capaz de producir recetas que reflejen las características propias de dicha cocina, considerando además las preferencias individuales de los usuarios. Esto no solo contribuirá a la promoción y preservación de la gastronomía peruana, sino que también facilitará el acceso a recetas adaptadas, fomentando una mayor apropiación cultural y culinaria. Asimismo, el modelo propuesto podrá servir como base para futuras investigaciones en el campo de la gastronomía computacional aplicada a la cocina peruana.

1.1.1. Formulación del Problema

¿Cómo desarrollar un modelo de generación automática de recetas personalizadas en español, centradas en la gastronomía peruana, utilizando técnicas de aprendizaje profundo?

Problemas específicos

- ¿Cómo recopilar y estructurar un conjunto de datos en español de recetas de la gastronomía peruana en español que incluya ingredientes e instrucciones?
- ¿Qué técnicas de preprocesamiento son necesarias para limpiar, normalizar y preparar el dataset de gastronomía peruana, garantizando la calidad de los datos para el entrenamiento del modelo?
- ¿Cuál es la arquitectura de modelo de aprendizaje profundo más adecuada para generar recetas personalizadas en español para la gastronomía peruana?
- ¿Qué modelos preentrenados existentes son más adecuados en la generación de recetas peruanas en español?
- ¿Qué métricas se pueden implementar para medir la calidad de las recetas generadas por el modelo generador de recetas personalizadas en español?

1.1.2. Antecedentes

- (*Goel et al., 2022*) *Computational gastronomy: A data science approach to food*

Este trabajo de investigación presenta una visión detallada de la evolución y los avances recientes en la gastronomía computacional (Computational Gastronomy),

un campo interdisciplinario que abarca áreas como la computación, la biología y la química de los alimentos. Los principales aspectos abordados en este estudio son: los datos de recetas, la combinación de ingredientes, el análisis del sabor, la evolución culinaria, la salud, la nutrición y la generación novedosa de recetas. Los avances recientes en este ámbito han llevado a cuestionarse si los computadores son capaces de generar recetas originales. En este contexto, el modelado del lenguaje emerge como una tarea clave dentro del procesamiento de lenguaje natural (Natural Language Processing, NLP), con múltiples aplicaciones, entre las cuales destaca la generación de texto. Esta última ha mostrado importantes progresos, especialmente en la aplicación de modelos lingüísticos para la generación automatizada de recetas a partir de listas de ingredientes o imágenes. Ejemplos de estos avances son modelos como RecipeGPT y el sistema de cocina inversa (inverse cooking), los cuales utilizan redes neuronales y modelos de transformadores preentrenados para producir recetas con instrucciones culinarias coherentes y adecuadas.

- **(Taneja et al., 2024) Monte Carlo Tree Search for Recipe Generation using GPT-2**

Los autores proponen el método *RecipeMC* para la generación de texto, utilizando el modelo *GPT-2* en combinación con funciones de recompensa basadas en el algoritmo *Monte Carlo Tree Search* (MCTS). El objetivo principal es imponer restricciones durante la generación de texto, mejorando así la *credibilidad* y *calidad* de las recetas producidas.

El método *RecipeMC* fue desarrollado mediante el afinamiento (*fine-tuning*) del modelo *GPT-2* utilizando el conjunto de datos *RecipeNLG*, que contiene aproximadamente 2.1 millones de recetas. Cada receta está compuesta por tres componentes:

el *nombre*, la *lista de ingredientes* y las *instrucciones*. Para distinguir claramente cada parte, se emplearon *tokens separadores* que indican el inicio y el final de cada componente. Posteriormente, el modelo fue entrenado para generar de manera secuencial los tres elementos: nombre de la receta, ingredientes e instrucciones.

Durante la fase de evaluación, *RecipeMC* se comparó con tres métodos base a través de dos tareas específicas: la *generación de ingredientes* a partir del nombre de la receta (**NAME** → **INGR**) y la *generación de instrucciones* a partir del nombre y los ingredientes (**NAME+INGR** → **INST**). Para evaluar el rendimiento del modelo, se utilizaron diversas métricas como *coherencia*, *F1-Score*, *Perplexity*, *ROUGE* y *BLEU*.

Además, se llevó a cabo una evaluación humana similar al *test de Turing*, en la cual se les preguntó a los participantes si consideraban que la receta había sido generada por una persona o por una máquina. Los resultados indicaron que los usuarios prefirieron los ingredientes generados por *RecipeMC* en un 55 % de los casos y las instrucciones en un 62 %, en comparación con las recetas creadas por humanos.

- (*Bieñ et al.*, 2020) *Cooking recipes generator utilizing a deep learning-based language model*

Este trabajo de investigación propone un modelo generador de recetas usando *Deep Learning* que es entrenado con recetas reales, este proyecto utiliza **Recipe1M+** un dataset público de 1 millón de recetas y 13 millones de imágenes de comida, así como también genera un nuevo dataset donde hace uso de técnicas de **Web Scraping** a gran escala para la extracción de recetas de sitios webs de comida.

Posteriormente, se analiza los datos para extraer entidades alimentarias de las listas de ingredientes, para el proceso de procesamiento de datos, se utilizó el conjunto de datos recopilados, para entrenar el modelo transformer, utiliza la arquitectura *GPT-2*. Este modelo se enfoca en la verosimilitud del resultado generado, esto quiere decir, que las recetas generadas por el modelo sean coherentes en términos de la combinación de ingredientes, las instrucciones de preparación y las cantidades utilizadas.

Para la evaluación del modelo generador de recetas usando *Deep Learning*, el modelo se puso a disposición en un sitio web de generador de recetas, para una parte de la evaluación de las recetas generadas se utilizó métricas de *Natural Language Generation(NLG)*, así como también estudios basados en la evaluación humana.

- **(Goel & Bagler, 2022) *Ratatouille: A tool for Novel Recipe Generation***

Este trabajo de investigación propone una herramienta para la generación automática de recetas de cocina. Para ello, se entrenaron modelos de *Deep Learning* como *LSTM* y *GPT-2*, utilizando el conjunto de datos *RecipeDB*, una recopilación estructurada que incluye ingredientes y perfiles nutricionales.

El dataset abarca una amplia variedad de recetas provenientes de diferentes culturas y regiones del mundo, con un total de 118,171 recetas distribuidas entre seis continentes, 26 regiones geoculturales y 74 países. Las recetas generadas por los modelos incluyen tres componentes principales: el título, los ingredientes y las instrucciones de preparación.

Para evaluar la calidad de los textos generados, se utilizó la métrica *BLEU score*.

Los resultados muestran que *GPT-2* alcanzó un puntaje de 0.806, superando significativamente a los modelos *LSTM*, donde el modelo *Char-level LSTM* obtuvo 0.347 y el *Word-level LSTM*, 0.412.

Finalmente, se desarrolló una aplicación llamada **Ratatouille**, que permite a los usuarios generar recetas originales y realistas. La arquitectura de la herramienta está compuesta por *React.js* en el frontend y *Flask* en el backend.

- (Galanis, 2022) *An update on cooking recipe generation with Machine Learning and Natural Language Processing*

Este trabajo presenta una revisión exhaustiva de los principales enfoques y sistemas utilizados para la generación y recomendación de recetas de cocina, aplicando diversas técnicas y metodologías.

Entre los sistemas destacados se encuentran:

- **Taaable**: utiliza un árbol jerárquico para representar recetas, vinculando ingredientes a clases relacionadas y asignando costos asociados.
- **JaDaCook**: emplea un enfoque de *K-Nearest Neighbours* para clasificar y recomendar recetas basadas en ingredientes.
- **CookIIS**: sistema web que utiliza representaciones estructuradas de recetas y criterios de similitud basados en relaciones taxonómicas.
- **CookingCAKE**: generaliza ingredientes y recetas en clases específicas, y utiliza un clasificador *Naive Bayes* para su categorización.

Además, se han aplicado modelos de *Deep Learning* como *GPT-2*, *Recurrent Neural*

Networks (RNNs), LSTMs y Transformers en tareas de generación automática de recetas.

Muchos de estos modelos fueron entrenados con datasets obtenidos mediante técnicas de *web scraping* desde sitios web de cocina, o utilizando conjuntos de datos públicos como *Kaggle, RecipeQA, Recipe1M+, RecipeNLG*, entre otros.

Se concluye que la búsqueda de la receta ideal sigue siendo un desafío tanto para aficionados como para expertos. La generación y recomendación automática de recetas enfrenta retos importantes, entre ellos: la adaptación de ingredientes, la consideración de la diversidad cultural, la coherencia en las combinaciones de ingredientes y la generación de instrucciones claras y preparables mediante técnicas de *Machine Learning, NLP y Deep Learning*.

1.1.3. Justificación

La generación automatizada de texto en idioma español es un área aún poco explorada, especialmente en tareas específicas como la generación de recetas de la gastronomía peruana. El desarrollo de un Modelo Generador de Recetas Personalizadas en español, enfocado en la cocina peruana y basado en técnicas de aprendizaje profundo, permitirá replicar recetas adaptadas a las preferencias individuales y a los ingredientes disponibles de cada usuario. Este modelo representa un aporte innovador en el campo de la gastronomía computacional, al facilitar la personalización culinaria mediante deep learning, y abre nuevas posibilidades para la interacción entre tecnología y cultura gastronómica.

1.1.4. Objetivos

1.1.5. Objetivo General

Desarrollar un modelo generador de recetas personalizadas en español de la gastronomía peruana usando aprendizaje profundo

Objetivos Específicos

Dentro de los objetivos específicos se incluyen:

- Construir un conjunto de datos representativo de la gastronomía peruana en el idioma español .
- Realizar el preprocesamiento del conjunto de datos, incluyendo limpieza y normalización y técnicas de reconocimiento de entidades (NER) para la extracción automatizada de ingredientes relevantes en las recetas .
- Seleccionar y adaptar una arquitectura de aprendizaje profundo basada en un modelo de lenguaje preentrenado, aplicando técnicas de ajuste fino (fine-tuning) para especializarla en la generación automática de recetas en español dentro del dominio culinario peruano.
- Evaluar las recetas generadas por el modelo mediante métricas automáticas (offline) y validación con usuarios finales.

1.1.6. Alcances y Limitaciones

Alcances

Los alcances definidos para el desarrollo final de la investigación son los siguientes:

- Un conjunto de datos de recetas en español, centrado en la gastronomía peruana, recopilado a partir de sitios web relevantes y representativos de dicha tradición culinaria.
- Un modelo generador de recetas personalizadas en idioma español, enfocado en la gastronomía peruana, diseñado para producir instrucciones culinarias coherentes y culturalmente relevantes a partir de ingredientes seleccionados por el usuario.
- Un método automatizado capaz de extraer y estructurar los ingredientes de manera coherente, lo cual facilitará la generación personalizada de recetas por parte del modelo de lenguaje.

Limitaciones

Los siguientes aspectos delimitan el alcance del presente trabajo de investigación:

- La calidad y la cantidad de recetas disponibles de la gastronomía peruana, las cuales pueden afectar la representatividad y diversidad del conjunto de datos.
- La presencia de sesgos de popularidad en la difusión de recetas según la región, lo que podría limitar la variedad cultural reflejada en los datos.
- El acceso limitado a recursos computacionales con GPU, lo que restringe la capacidad para entrenar modelos de lenguaje de gran tamaño o realizar múltiples experimentos.

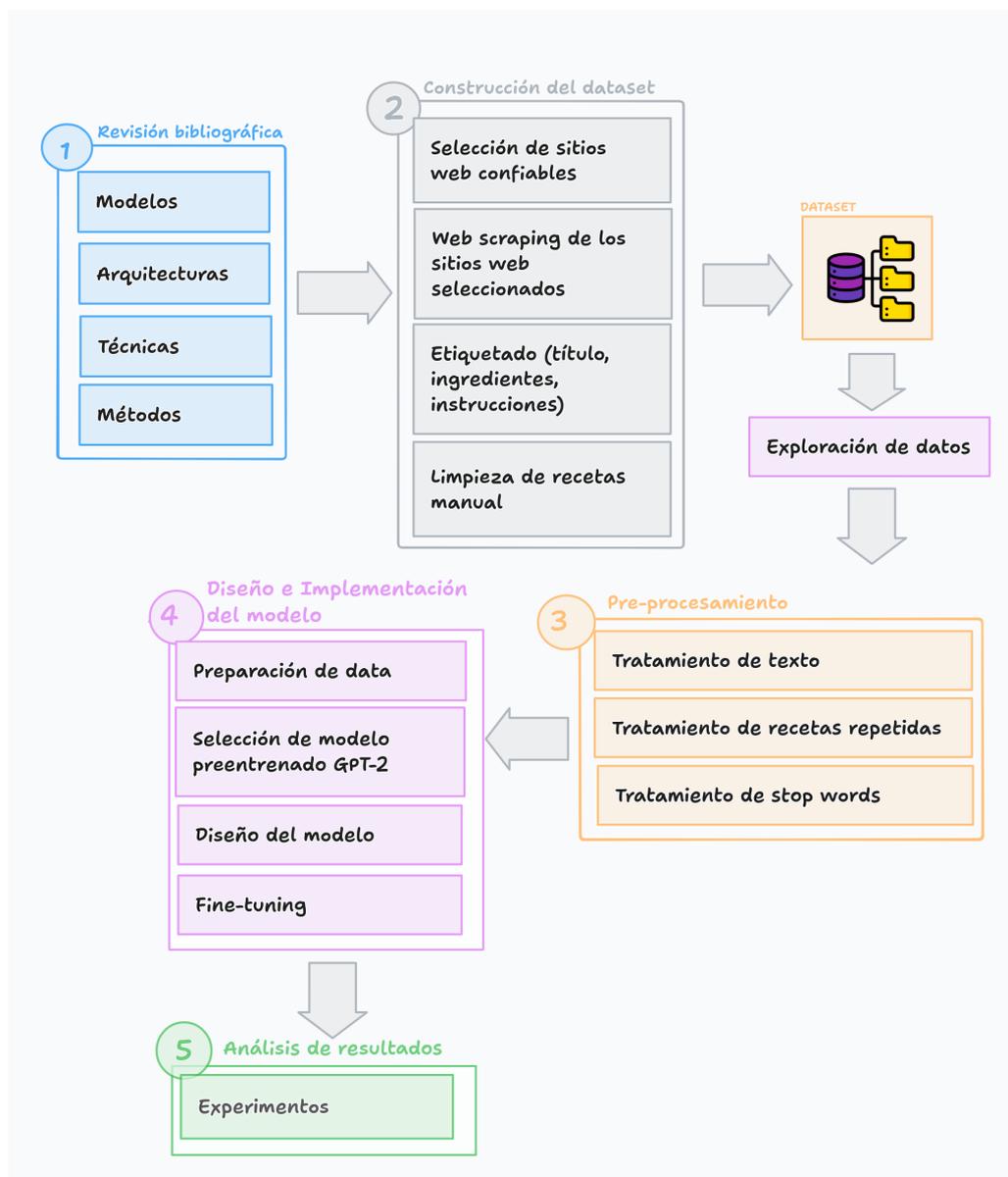
1.1.7. Metodología

Este proyecto de investigación será de tipo aplicado. Según Lozada, [2014](#), la investigación aplicada tiene como objetivo generar nuevo conocimiento orientado a la solución

de problemas concretos. En este contexto, se implemento un modelo generador de recetas personalizadas en español de la gastronomía peruana usando deep learning. Etapas de

Figura 1.1

Etapas de la metodología propuesta



Nota: Elaboración propia.

desarrollo para la investigación propuesta:

1. Revisión bibliográfica

La revisión bibliográfica se basa en artículos científicos, tesis y otros documentos relacionados con la generación automatizada de texto, así como con modelos, ar-

quitecturas, técnicas y métodos aplicados en este ámbito. Para la identificación y recopilación de fuentes relevantes para el desarrollo del proyecto de investigación, se utilizaron bases de datos académicas como Scopus, IEEE y ACM.

2. Construcción del dataset

- Selección de sitios web confiables.
- Extracción de datos mediante técnicas de web scraping aplicadas a los sitios seleccionados.
- Etiquetado estructurado de las recetas, identificando título, ingredientes e instrucciones.
- Proceso de limpieza manual de los datos, que incluye la eliminación de recetas incompletas, consejos adicionales sobre la preparación, comentarios sobre el origen de la receta y cualquier ruido presente en los ingredientes o en las instrucciones. En esta última sección, también se suprime texto redundante, priorizando la claridad y concisión de las acciones descritas.

3. Pre-procesamiento

- Normalización del texto: conversión a minúsculas de todos los campos del conjunto de datos, incluyendo título, ingredientes e instrucciones.
- Eliminación de recetas duplicadas e incompletas (aquellas que contienen únicamente ingredientes o instrucciones).
- Remoción de stop words para reducir el ruido y mejorar la calidad del texto procesado.

4. Diseño e implementación

- Preparación del conjunto de datos para las fases de entrenamiento y validación
- Selección del modelo preentrenado GPT-2 como base para la generación de texto.
- Definición de la arquitectura y los parámetros del modelo, incluyendo el número de épocas, el tamaño del batch, los steps y la tasa de aprendizaje.
- Aplicación de fine-tuning al modelo preentrenado utilizando el conjunto de datos preparado.

5. **Análisis de resultados** : En esta etapa se analizarán los resultados obtenidos, evaluando la coherencia y calidad de las recetas generadas por el modelo de generación de recetas de la gastronomía peruana. La evaluación se realizó de forma offline mediante métricas automáticas, como BLEU y GLEU. Adicionalmente, se llevó a cabo una prueba con usuarios, quienes debían identificar si cada receta había sido generada por un humano o por un sistema automatizado.

1.1.8. Contribuciones

El presente trabajo ofrece las siguientes contribuciones:

- Desarrollo de un modelo generador de recetas personalizadas en español, centrado en la gastronomía peruana, mediante el uso de técnicas de deep learning.
- Construcción de un conjunto de datos (*dataset*) de recetas de la gastronomía peruana en idioma español.

Capítulo 2

Marco Teórico

2.1. Inteligencia artificial(IA)

En 1950, Alan Turing publicó el artículo “ Computing Machinery and Intelligence ”, en el cual planteó la posibilidad de que una máquina pudiera imitar el comportamiento humano. Como parte de su propuesta, introdujo el Test de Turing, una prueba diseñada para determinar si una máquina puede ser considerada inteligente, evaluando su capacidad para generar respuestas indistinguibles de las de un ser humano (Turing, 2012).

La primera publicación relacionada con la inteligencia artificial data de 1984, y a partir de entonces su desarrollo y aplicación comenzaron a intensificarse. Actualmente, se emplean diversos métodos en el campo de la inteligencia artificial, entre ellos los sistemas expertos, las redes neuronales, la lógica difusa, los sistemas basados en conocimiento, los chatbots y la generación automática de texto (Hernandez & Duque, 2020).

2.2. Generación de lenguaje natural (NLG)

La Generación de Lenguaje Natural (NLG) es una subárea del Procesamiento del Lenguaje Natural (PLN) que se encarga de producir texto en lenguaje natural con el fin de cumplir objetivos comunicativos específicos. Estos textos generados pueden variar

desde una sola frase hasta explicaciones completas de varias páginas, a diferencia de la Comprensión del Lenguaje Natural (NLU), que analiza texto existente.

2.2.1. Métodos generales de NLG

Una idea general de la tarea de generación de lenguaje natural consiste en identificar una secuencia óptima de palabras que maximice la probabilidad condicional de generación, dado un conjunto de posibles secuencias y una entrada fuente, esta probabilidad depende tanto de las palabras generadas previamente como de la información inicial proporcionada.

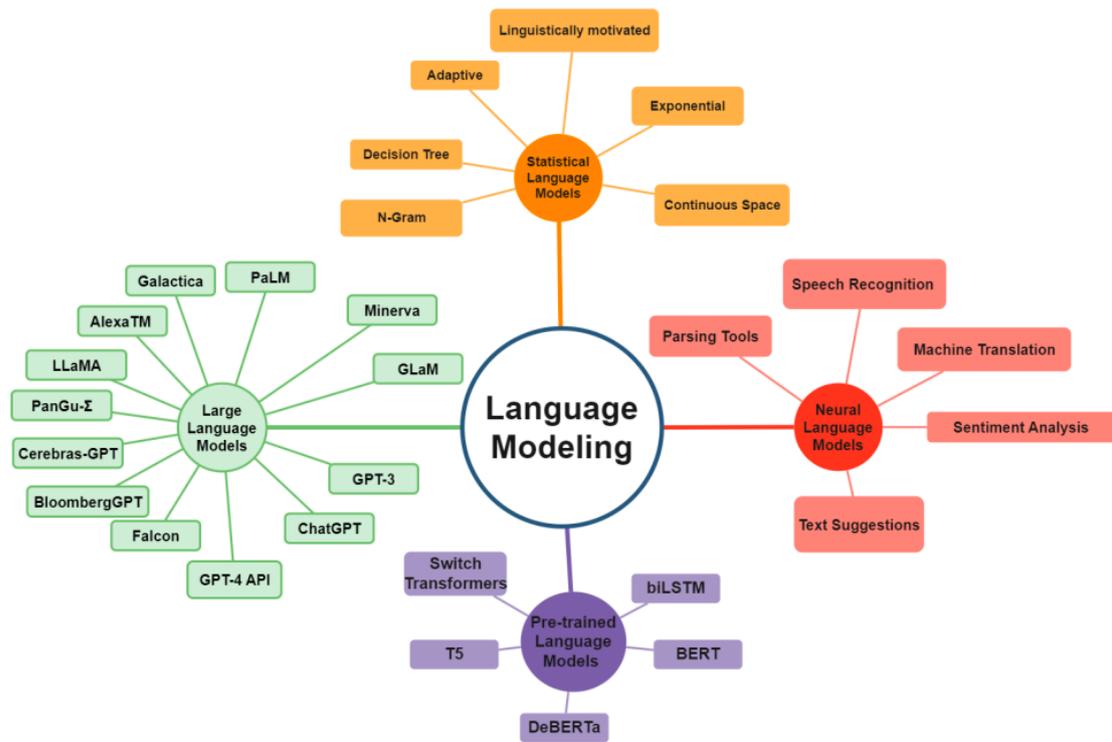
Para abordar esta tarea, se emplean diversos enfoques basados en aprendizaje profundo, entre los que destacan las redes neuronales recurrentes (RNN), los modelos transformer, redes neuronales basadas en grafos (GNNs) y modelos preentrenados.

2.3. Grandes Modelos de lenguaje (LLM)

El lenguaje desempeña un papel fundamental en la comunicación entre seres humanos y en la forma en que interactuamos con tecnologías como asistentes virtuales, motores de búsqueda o traductores. Ante la necesidad de que las máquinas realicen tareas lingüísticas cada vez más complejas, se han desarrollado grandes modelos de lenguaje (LLMs), impulsados por arquitecturas como los transformers, el aumento en la capacidad computacional y la disponibilidad de grandes volúmenes de datos. Estos modelos han alcanzado un desempeño que se aproxima al nivel humano en diversas tareas (Naveed et al., 2023).

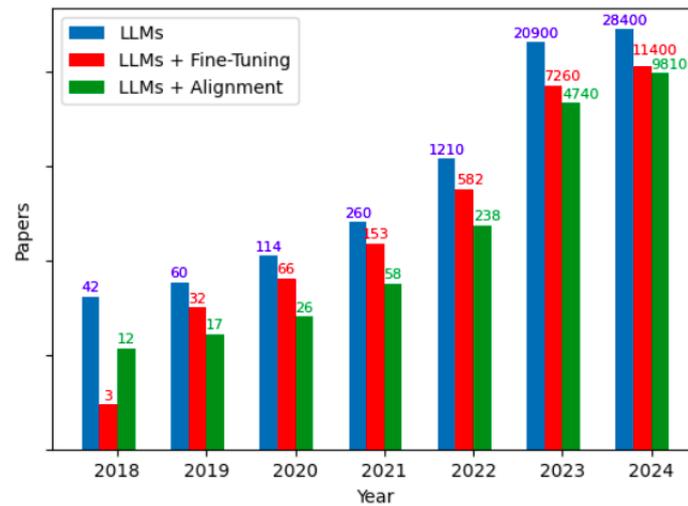
En la Figura 2.1 se presenta una visualización que organiza los principales enfoques

y categorías del modelado de lenguaje, agrupados en cuatro subgrupos. En el nodo central se ubica el concepto general de Language Modeling. El primer grupo, representado en color naranja, corresponde a los modelos estadísticos, basados en la probabilidad y el análisis de secuencias de palabras, como los modelos N-Gram y Decision Tree. El segundo grupo, en color rojo, incluye los modelos neuronales, que emplean redes neuronales profundas para aprender representaciones del lenguaje. Este grupo destaca por su aplicabilidad práctica en tareas como reconocimiento de voz, traducción automática y análisis de sentimientos. El tercer grupo, coloreado en morado, abarca los modelos preentrenados, tales como T5, Switch Transformers, DeBERTa, BERT y biLSTM. Estos modelos marcaron un hito en el campo del procesamiento del lenguaje natural al popularizar el uso del transfer learning. Finalmente, el cuarto grupo, representado en color verde, reúne los modelos de lenguaje de gran escala (LLMs), como GPT-3, GPT-4, LLaMA y PaLM. Estos modelos constituyen el estado del arte en procesamiento de lenguaje natural, destacándose por su capacidad para generar texto coherente y fluido a partir de entradas complejas.

Figura 2.1*Tipos de modelado de lenguaje*

Nota: Extraído de (Hadi et al., 2023).

En la Figura 2.2 se muestra la evolución del número de artículos publicados por año que contienen las palabras clave "Large Language Model" (LLMs), "Large Language Model + Fine-Tuning" y "Large Language Model + Alignment". Cada categoría está representada por un color distintivo: los artículos relacionados con LLMs en general se indican en azul, aquellos que abordan LLMs con técnicas de ajuste fino (fine-tuning) en rojo, y los que tratan sobre alineación (alignment) con objetivos humanos en verde. Esta tendencia refleja el creciente interés de la comunidad científica por los modelos de lenguaje de gran escala y sus aplicaciones, destacando la relevancia del ajuste fino y la alineación en el desarrollo y despliegue responsable de dichos modelos.

Figura 2.2*Trabajos publicados en el tiempo*

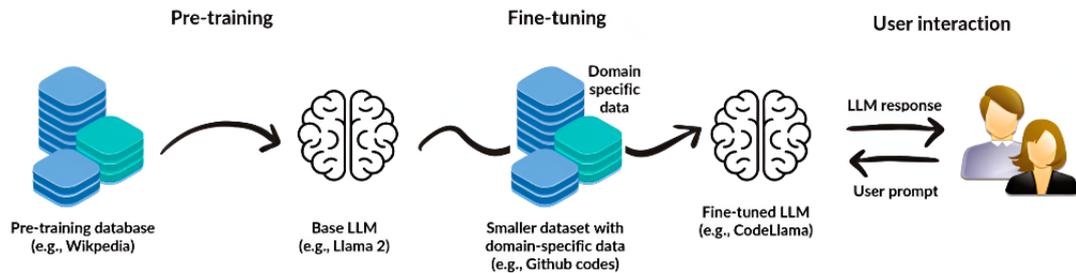
Nota: Extraído de (Naveed et al., 2023).

2.3.1. Ajuste fino

EL Fine-tuning (Ajuste fino), es una técnica utilizada para adaptar un modelo de lenguaje preentrenado a una tarea específica, utilizando un conjunto de datos más pequeños y especializado. Esta estrategia permite aprovechar el conocimiento general aprendido durante el pre-entrenamiento, ajustándolo para alcanzar un alto rendimiento en tareas concretas como clasificación de sentimientos, traducción automática, generación de texto o respuesta a preguntas.

En la Figura 2.3, muestra el flujo completo del desarrollo de un modelo de lenguaje ajustado: desde el entrenamiento con datos generales (pre-training), pasando por la especialización con datos específicos (fine-tuning), hasta su uso final mediante la interacción con el usuario. Este proceso permite transformar un modelo generalista en una herramienta especializada y útil en contextos particulares, tales como la generación de recetas.

Figura 2.3
LLM fine-tuning



Nota: Extraído de (Management Solutions, [2024](#)).

Ventajas del Ajuste fino

- **Aprovechamiento del conocimiento previo:** Al comenzar desde un modelo preentrenado, se capitaliza todo el conocimiento lingüístico adquirido previamente, lo que acelera el aprendizaje y mejora el desempeño en la tarea específica.
- **Menor cantidad de datos y recursos:** Comparado con el entrenamiento desde cero, el fine-tuning requiere menos datos etiquetados y menos poder computacional, lo que lo convierte en una opción más viable y económica para distintas organizaciones.
- **Especialización del modelo:** Permite ajustar el modelo a dominios o tareas específicas, logrando soluciones altamente personalizadas y eficaces.
- **Facilita la transferencia de aprendizaje:** Un modelo ya ajustado puede ser sometido a un nuevo proceso de fine-tuning para tareas relacionadas, reutilizando el conocimiento previamente aprendido con relativamente pocos datos adicionales.

Desafíos del Ajuste Fino

- **Sobreespecialización :** Cuando el modelo se entrena con un conjunto de datos demasiado específico, puede perder su capacidad de generalización, lo que reduce su

rendimiento en datos nuevos o ligeramente diferentes.

Cuando el modelo se entrena con un conjunto de datos en específico, puede perder su capacidad de generalización, lo que reduce su rendimiento en diferentes datos nuevos o con alguna modificación.

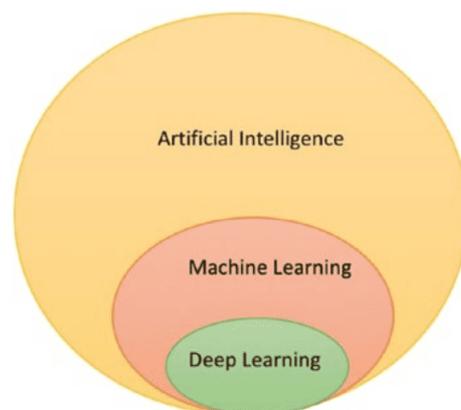
2.4. Deep Learning (DL)

El deep learning, o aprendizaje profundo, es un método de la inteligencia artificial (IA) que aborda el desafío del aprendizaje de representaciones mediante la generación de representaciones complejas a partir de otras más simples. Este enfoque permite construir modelos sofisticados a partir de estructuras básicas, utilizando redes neuronales artificiales compuestas por múltiples capas ocultas (Ian Goodfellow, 2016).

El aprendizaje profundo ha mejorado el estado del arte en áreas como reconocimiento de voz e imágenes. Utiliza backpropagation para ajustar sus parámetros, permitiendo que cada capa aprenda a partir de la anterior (LeCun et al., 2015).

Figura 2.4

Diagrama de Venn que muestra la relación entre Artificial Intelligence, Machine Learning y Deep Learning



Nota: Extraído de (Moolayil et al., 2019).

2.5. Procesamiento de lenguaje natural (PLN)

El procesamiento de lenguaje natural (PLN) es una rama de la inteligencia artificial y de la lingüística computacional que se ocupa de la interacción entre computadoras y seres humanos a través del lenguaje natural (Vajjala et al., 2020).

2.5.1. Word Embeddings

La capacidad de una representación textual para identificar palabras que aparecen en contextos similares se denomina similitud distributiva entre palabras. Esta técnica se fundamenta en la similitud distribucional, la cual emplea el contexto lingüístico para inferir el significado de una palabra (Vajjala et al., 2020).

Word2vec es un modelo de representación vectorial de palabras que emplea redes neuronales para capturar relaciones semánticas entre términos, basándose tanto en la similitud como en la hipótesis distribucional. Las representaciones generadas, conocidas como embeddings, son vectores de baja dimensionalidad y alta densidad. Estas incrustaciones son especialmente útiles en tareas de procesamiento de lenguaje natural y aprendizaje automático (Vajjala et al., 2020).

2.5.2. Pre-trained word embeddings

El entrenamiento de embeddings de palabras puede resultar costoso en términos de tiempo y recursos computacionales. Sin embargo, en muchos casos es posible emplear incrustaciones preentrenadas generadas a partir de grandes corpus de texto, como Wikipedia o colecciones de artículos periodísticos. Estos embeddings consisten en pares de claves y valores que representan las palabras del vocabulario junto con sus respectivos vec-

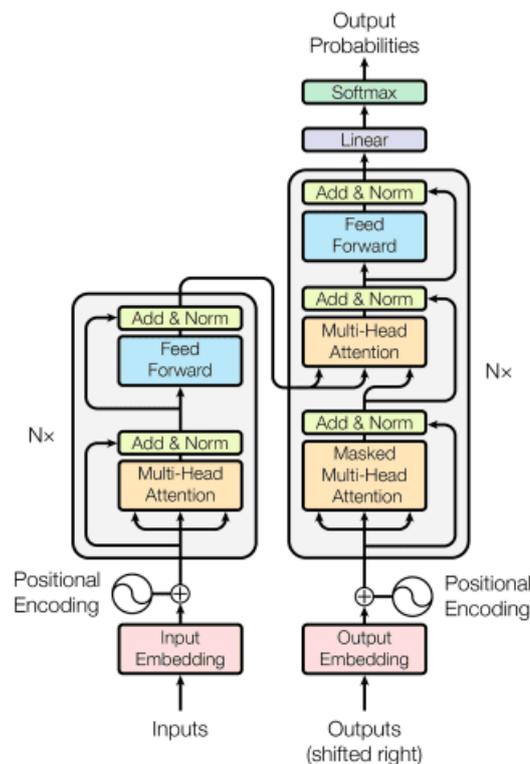
tores. Entre las representaciones preentrenadas más utilizadas se encuentran Word2Vec (desarrollado por Google), GloVe (de Stanford) y fastText (de Facebook) (Vajjala et al., 2020).

2.6. Transformer

Esta arquitectura neuronal, que actualmente se considera el estado del arte en modelos secuenciales, es conocida como Transformer. En el campo del *Natural language procesing(NLP)*, el Transformer resuelve tareas *Seq2Seq* sin los inconvenientes de las dependencias a largo plazo presentes en las *LSTM* o *RNN*. Esta innovadora arquitectura fue propuesta por primera vez en 2017 en un artículo titulado “*Attention Is All You Need*”(Vaswani et al., 2017).

Figura 2.5

Estructura de una arquitectura transformer



Nota: Extraído de (Vaswani et al., 2017).

2.6.1. Stack Encoder y Decoder

La estructura de la arquitectura Transformer consta de un sistema de “Encoder” y “Decoder”, tanto el encoder (parte izquierda) como el decoder (parte derecha), está compuesto por módulos que se amplía entre sí varias veces, esto se observa en la figura con la notación Nx , estos módulos están compuestos por capas de Multi-Head Attention y Feed Forward, que son técnicas utilizadas en el proceso de lenguaje natural (Vaswani et al., 2017).

En este sistema, las frases objetivas, que funcionan como entradas y salidas, no pueden ser utilizadas en su forma original como texto. En cambio, se les asigna una representación numérica en un espacio matemático con múltiples dimensiones. Esta transformación es necesaria debido a que los modelos de procesamiento de lenguaje natural no pueden procesar directamente cadenas de texto (Vaswani et al., 2017).

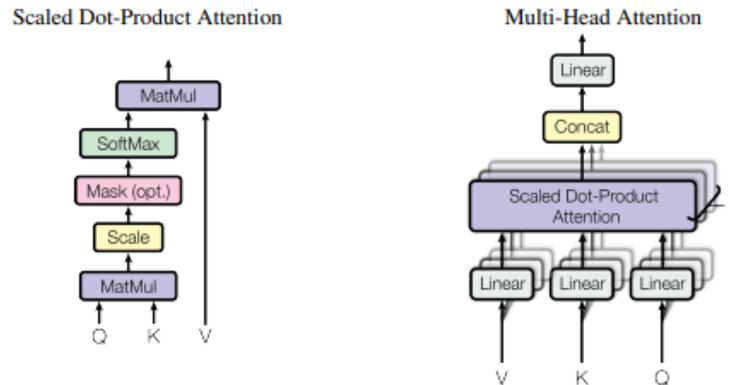
La asignación de una palabra o una parte del texto es esencial en este modelo, para lograr ello, se incorpora información de posición a la representación incrustada de cada palabra, la cual se representa mediante un vector de dimensión N , con ello se logra capturar el orden de los elementos en una secuencia (Vaswani et al., 2017). En la figura 2.6, se muestra que en la (Izquierda) Scaled Dot-Product Attention, (Derecha) Multi-Head Attention, consta de varias capas de atención que funcionan en paralelo.

2.6.2. Attention

A continuación, se describe el contenido y propósito de cada una de las figuras presentadas:

1. Dot-Product Attention: El método “Scaled Dot-Product Attention” es aplicado

Figura 2.6
Attention



Nota: Extraído de (Vaswani et al., 2017).

en algunos modelos y se fundamenta en el cálculo de productos escalados entre consultas y claves. Estos productos se dividen por la raíz cuadrada de la dimensión d_x y se emplea una función *softmax* para obtener los pesos correspondientes a los valores. En la implementación práctica, este cálculo se lleva a cabo en conjuntos de *queries*(Q), *keys*(K) y *values*(V) que son organizados en matrices Q , K y V , respectivamente. Como resultado, se obtiene una matriz de salidas (Vaswani et al., 2017).

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (2.1)$$

2. Multi-Head Attention: En lugar de realizar una única función de atención con *keys*, *values* y *queries*, con dimensión d_{model} , se muestra beneficioso proyectar linealmente *keys*, *values* y *queries* h veces con proyecciones lineales diferentes y aprendidas a dimensiones d_k y d_v , respectivamente. En cada una de estas versiones proyectadas de *keys*, *values* y *queries*, luego realizamos la función de atención en paralelo, obteniendo valores de salida de dimensión d_v (Vaswani et al., 2017).

La atención multi-head permite que el modelo atienda conjuntamente información de diferentes subespacios de representación en diferentes posiciones (Vaswani et al., 2017).

$$MultiHead(Q, K, V) = Concat(head_1, \dots, head_h) W^O \quad (2.2)$$

Where:

$$head_i = Attention(QK_i^Q, KW_i^K, QVW_i^V) \quad (2.3)$$

Donde las proyecciones son matrices de parámetros $W_i^Q \in R^{d_{model} \times d_k}$, $W_i^K \in R^{d_{model} \times d_k}$, $W_i^V \in R^{d_{model} \times d_v}$ y $W^O \in R^{hd_v \times d_{model}}$

2.7. GPT Framework

Generative Pre-trained Transformer es un modelo de lenguaje, esta utiliza la arquitectura Transformer, está diseñada para tareas de Natural Language Processing, en el entrenamiento, este modelo adquiere conocimiento sobre patrones y estructura lingüística, lo cual le capacita para producir texto coherente y pertinente al recibir una entrada específica (Radford et al., 2018).

2.7.1. Unsupervised Pre-Training

Basándonos en un conjunto no supervisado de tokens $U = \{u_1, \dots, u_n\}$, se aplica un objetivo común de modelado del lenguaje para maximizar una probabilidad específica (Radford et al., 2018).

$$L_1(U) = \sum_i \log P(u_i | u_{i-k}, \dots, u_{i-1}; \theta) \quad (2.4)$$

Donde:

- $U = u_1, \dots, u_n$ es un corpus no etiquetado de tokens
- P es la probabilidad condicional, modelada por una red neuronal con parámetros, Θ
- k es el tamaño de window context

2.7.2. Supervised Fine-Tuning

Después de un Unsupervised Pre-Training, se modifica el modelo preentrenado para que se adapte a una tarea particular mediante el uso de datos etiquetados, las entradas del modelo preentrenado se someten a un procesamiento que resulta en activaciones finales. Estas activaciones se introducen en una capa de salida lineal adicional con *softmax*. Durante el entrenamiento, el objetivo principal es maximizar la probabilidad asociada a la tarea específica $L_2(C)$ (Radford et al., [2018](#)).

$$L_2(C) = \sum_{(x,y)} \log P(y|x^1, \dots, x^m) \quad (2.5)$$

Donde:

- C es el conjunto de datos etiquetados
- $x = x^1, \dots, x^m$ es la secuencia de tokens de entrada
- y es una etiqueta

Adicionalmente, se incorpora el modelado del lenguaje como un objetivo secundario. Esto significa que el modelo también aprende a maximizar el objetivo $L_1(C)$ al aplicar el aprendizaje no supervisado a los datos etiquetados. Como resultado, el objetivo combinado se

define de la siguiente manera (Radford et al., 2018):

$$L_3 = L_2(C) + \lambda * L_1(C) \quad (2.6)$$

Donde:

- λ es un peso a equilibrar entre ambos objetivos

2.8. Gastronomía Peruana

La gastronomía del Perú es una expresión cultural diversa y relevante que ha experimentado cambios a lo largo del tiempo. Tiene sus raíces en las civilizaciones precolombinas y ha sido enriquecida por las influencias culinarias de distintas culturas, entre ellas las andinas, amazónicas, europeas, asiáticas y africanas. La biodiversidad y la convivencia de estas culturas en un territorio megadiverso han contribuido a la impresionante variedad de sabores y texturas que caracterizan a la gastronomía peruana (Guardia, 2020).

Figura 2.7*Plato peruano ceviche*

Nota: Extraído de (Oficina de Marketing Institucional, [2023](#)).

2.8.1. Cocina peruana como regalo culinario para el mundo

La cocina ha adquirido un significado simbólico y representa la identidad de una región que supera las fronteras nacionales. Es reconocida como un componente vital de la tradición y el legado cultural que merece ser protegido, al mismo tiempo que desempeña un papel fundamental en la definición y redefinición de identidades. La gastronomía peruana se destaca como un ejemplo destacado a nivel global, sirviendo como un poderoso punto de encuentro entre diversas culturas (Guardia, [2020](#)).

2.8.2. Cocina peruana como patrimonio cultural, identidad cultural y turismo

A partir de la conexión entre la identidad cultural y el patrimonio cultural, en el contexto de la gastronomía peruana, emergen dos aspectos significativos. Primero, la cocina tiene la capacidad de potenciarse y recibir apoyo a través de las declaraciones de patrimonio cultural que se relacionan con sus componentes. Segundo, esta valiosa riqueza cultural que consideramos propia debe ser compartida y vivida, especialmente mediante la promoción del turismo (Guardia, 2020).

2.8.3. Cocina Peruana como patrimonio cultural 2007

Mediante la Resolución Directoral 1362 del año 2007 del Instituto Nacional de Cultura, que en ese momento era la entidad encargada de la cultura, se oficializó la declaración de la Cocina Peruana como patrimonio inmaterial. Los fundamentos de esta declaración reflejan conceptos presentes en el discurso actual sobre la cocina y la gastronomía peruana, Se establece una conexión entre la herencia cultural, el significado simbólico de la producción de alimentos en sociedades originarias, y se subraya la relevancia de las prácticas de cultivo y crianza, así como la amplia gama de productos emblemáticos como la papa, el maíz y la kiwicha (Guardia, 2020).

2.8.4. Gastronomía del Sur Peruano

En el sur del Perú, en lugares como Ica, Tacna, Arequipa, Cusco, Puno y Moquegua, se puede apreciar una gran variedad culinaria. Estas regiones comparten rasgos culinarios similares, que son resultado de los vínculos tradicionales ancestrales (Berríos

Hurtado, 2014).

Platos representativos por región del sur:

- En Cusco, se conmemora la cultura ancestral de los Incas y se pone énfasis a los granos andinos, como el maíz. Además, se hacen referencia a platos típicos correspondientes a diferentes meses del año, como la crema de maíz “q’olla lawa” con huacatay, el “puchero” en febrero, el “chupe de zapallo” durante la Semana Santa, los “ponches de habas” en mayo, y una variedad de carnes que incluyen res, cordero, chanco y cuy, acompañadas de papas, habas, quesos y ocas (Berríos Hurtado, 2014).
- Arequipa, posee entre sus platos más reconocidos el “rocoto relleno”, el “solterito de queso”, la “zarza de machas”, la “ocopa arequipeña” y el “adobo de chanco”, así como también su apreciado licor anisado (Berríos Hurtado, 2014).
- En Puno, posee en su gastronomía platos como el “cancacho” que consiste en lechón o cordero al horno, el “pesque de quinua” que se prepara con leche y queso, el “chairo” que consiste de una sopa con carne de res y cordero, papas, repollo, chalona, trigo y queso, y el “pejerrey frito” elaborado con pejerreyes cultivados en el lago (Berríos Hurtado, 2014).

2.8.5. Gastronomía del Norte Peruano

La gastronomía del norte peruano es diversa y comprende las regiones de Tumbes, La Libertad, Amazonas, Cajamarca y las zonas costeras (Berríos Hurtado, 2014).

- En Tumbes, se resalta la presencia de mariscos frescos en platos como ceviche, sudado y majarisco.

- Piura ofrece mariscos y postres como natillas y algarrobina.
- En la sierra de Piura ofrece platos como mote con arvejas y alfeñique.
- Amazonas y Cajamarca tienen sus especialidades, como plátanos rellenos y humitas en Amazonas, mientras que en Cajamarca posee humitas y picante de cuy.
- Lambayeque es famoso por su seco de cabrito, el arroz con pato y el chinguirito

2.8.6. Gastronomía de la selva peruana

La gastronomía de la selva peruana tienen características étnicas y místicas debido al conocimiento ancestral sobre las plantas, que se les atribuye propiedades curativas y revitalizantes. La selva peruana abarca las regiones de Loreto, San Martín, Ucayali, la ceja de selva y selva alta, donde las tradiciones culinarias ancestrales están arraigadas (Berríos Hurtado, 2014). platos y refrescos representativos de la selva:

- Los “juanes” es un plato popular de la selva, este plato consiste en tamales de arroz rellenos con presas de gallina, huevo, aceitunas y otros ingredientes. Estos tamales están envueltos en hojas de bijao, lo que les confiere un sabor único.
- El tacacho con cecina, el juane, el ceviche de doncella, el venado a la parrilla, el inchicapi, el ninajuane y la patarashca.
- Entre los refrescos destacados se encuentra la aguajina, la carambola y los helados de aguaje

Capítulo 3

Desarrollo del proyecto

En esta sección se describen los pasos y procedimientos realizados durante el desarrollo del proyecto. Se abordan la construcción y exploración del conjunto de datos, las etapas de preprocesamiento, la personalización del modelo NER para la extracción de entidades, la selección del modelo generativo para la elaboración de recetas, así como la implementación mediante fine-tuning y el entrenamiento de los modelos personalizados NER y GPT-2.

3.1. Construcción del dataset

3.1.1. Web Scraping

El web scraping es un proceso de recopilación de datos a partir de sitios o páginas web. Para ello, se emplean scripts que acceden a sitios predefinidos con el objetivo de extraer y almacenar información relevante, que puede ser utilizada posteriormente en diversas tareas, como en este caso, para la generación de recetas basadas en la gastronomía peruana. No obstante, esta técnica presenta varios desafíos, entre ellos el tiempo requerido para la extracción de datos. Para mitigar este problema, el uso de computación paralela permite acelerar el proceso al extraer grandes volúmenes de información de múltiples sitios web de manera simultánea (Mitchell, 2018).

3.1.2. Proceso web scraping

En este trabajo de investigación, la construcción del dataset se llevó a cabo mediante la técnica de web scraping. El proceso se inició con la recopilación de sitios web que contuvieran recetas representativas de la gastronomía peruana. Para ello, se realizó una búsqueda utilizando palabras clave como: comida peruana, platos peruanos, gastronomía peruana, platos representativos de la gastronomía peruana, recetas de chefs de renombre, recetas de la sierra, costa y selva. Esta etapa fue fundamental para garantizar la diversidad y representatividad de los datos.

Posteriormente, se procedió al almacenamiento de las URLs correspondientes a cada sitio web identificado. Dado que cada uno presentaba una estructura distinta, fue necesario desarrollar scripts específicos para cada fuente. Una vez recopiladas las URLs, se ejecutaron los respectivos scripts para la extracción de recetas. Para optimizar el tiempo de procesamiento, se aplicaron técnicas de paralelización que permitieron realizar la recolección de datos de manera simultánea. El flujo detallado de este proceso de extracción se describe en los párrafos siguientes.

3.1.3. Proceso de extracción de recetas

El proceso de extracción de datos de las recetas a partir de los sitios web se llevó a cabo de la siguiente manera:

- Identificación de sitios web que contienen recetas representativas de la gastronomía peruana.
- Verificación manual de cada sitio web recolectado, evaluando tanto la cantidad como la calidad de las recetas. Se comprobó que cada página incluyera los tres elementos

fundamentales: título de la receta, lista de ingredientes y pasos de preparación. Esta revisión permitió determinar si el contenido del sitio aportaría valor al dataset y si era adecuado para aplicar web scraping en etapas posteriores.

- Implementación de scripts personalizados para la recolección de datos. Dado que cada sitio web posee una estructura distinta (en términos de paginación, disposición y agrupación de los elementos), se desarrollaron funciones específicas para cada caso. Esta etapa requirió una inspección minuciosa del código HTML, especialmente de las etiquetas (por ejemplo, div, li, entre otras), con el fin de localizar con precisión las secciones correspondientes al título, ingredientes y pasos de preparación de cada receta.
- Ejecución de los scripts, considerando que cada uno requiere un tiempo específico de procesamiento para la extracción de los datos. Como resultado, se generó un archivo en formato JSON que almacena las recetas recolectadas de manera estructurada.

En resumen, el proceso de extracción de datos de las recetas implicó una etapa inicial de búsqueda de sitios web que contuvieran contenido relacionado con la gastronomía peruana, como paso previo al web scraping. Durante la validación manual, se descartaron aquellos sitios que no presentaban una estructura HTML adecuada, ya que esto dificultaría la automatización del proceso. Se tuvo en cuenta la paginación y la organización de los elementos clave de cada receta, como el título, los ingredientes y los pasos de preparación. Finalmente, como resultado del web scraping aplicado a cada sitio web seleccionado, se obtuvo un archivo en formato JSON que almacena de forma estructurada todas las recetas extraídas.

3.1.4. Resumen y resultados de los sitios web seleccionados

En la Tabla [3.1](#), se presenta un resumen de los sitios web seleccionados para la extracción de datos mediante web scraping. Asimismo, se muestra la cantidad de recetas extraídas desde cada uno de estos sitios.

Tabla 3.1

Resumen del número de recetas extraídas por sitio web mediante web scraping

Nro	Sitio web	Nro recetas
1.	Recetas gratis	173
2.	Yanuq	491
3.	Perú Delicias	311
4.	Come Peruano	253
5.	Comidas Peruanas	429
Total		1657

Nota: Fuente propia.

3.1.5. Desafíos encontrados durante el proceso de web scraping

A continuación se hace mención de los desafíos que se afronto durante el proceso de extracción de datos de las recetas.

- Calidad versus cantidad de recetas, es importante destacar que, para el entrenamiento efectivo del modelo, se requiere no solo una gran cantidad de recetas, sino también que estas sean de alta calidad. En este sentido, resulta fundamental priorizar la extracción masiva de recetas que cumplan con criterios estructurales y semánticos adecuados, garantizando así que los datos utilizados sean representativos y útiles para el propósito de esta investigación.
- Se realizó un análisis exhaustivo de la estructura de cada sitio web seleccionado, prestando especial atención a las etiquetas HTML. Esta etapa fue fundamental para facilitar la implementación de scripts personalizados que permitieran automatizar

de manera eficiente el proceso de extracción de datos de las recetas.

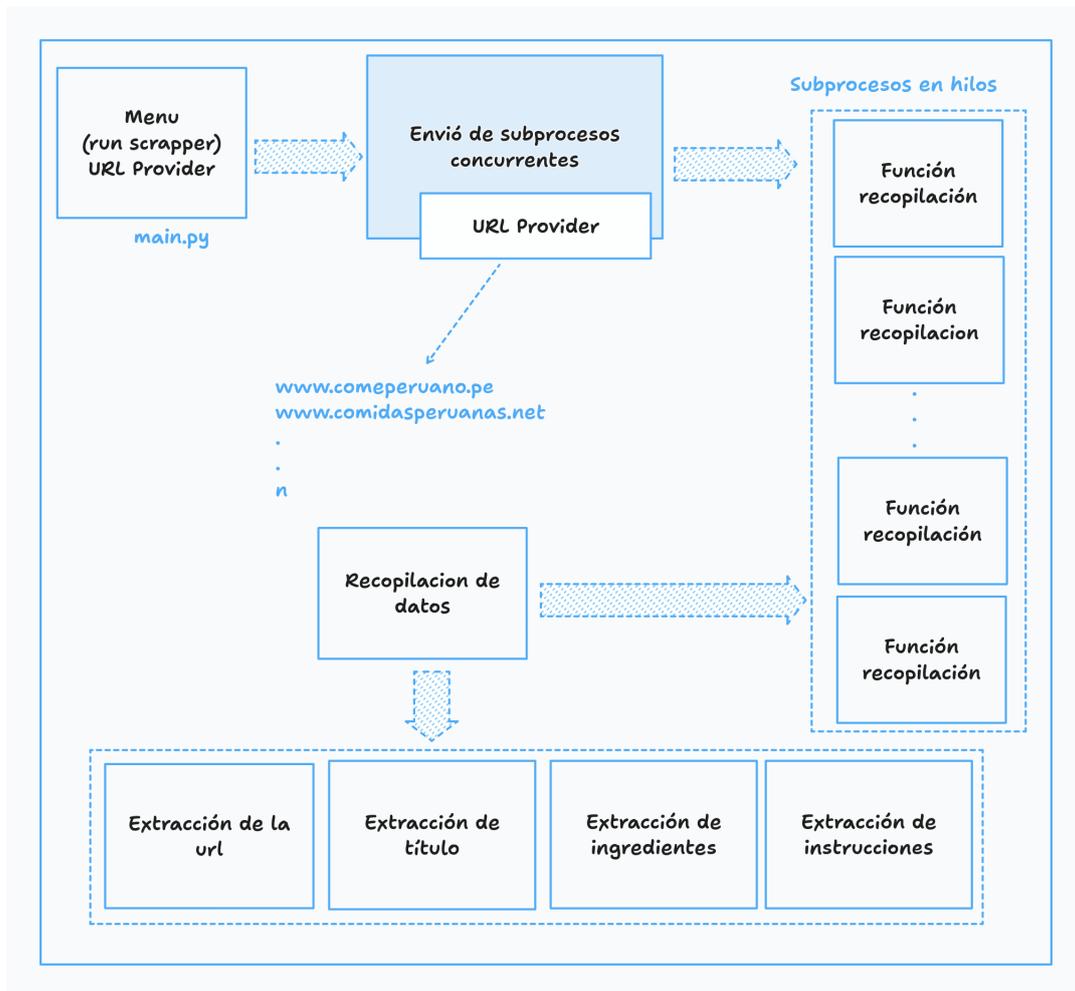
3.1.6. Flujo del proceso de recopilación automatizada de recetas

El script para la recolección de recetas fue desarrollado utilizando el lenguaje de programación Python 3. Para realizar las solicitudes HTTP, se empleó la biblioteca `urllib`, mientras que el módulo `concurrent.futures` permitió la ejecución de código en paralelo, optimizando así el tiempo de procesamiento. Además, se utilizó la biblioteca `Beautiful Soup`, ampliamente empleada para la extracción de información de sitios web y el análisis de documentos en formato HTML y XML. Finalmente, el módulo `json` fue utilizado para almacenar las recetas en formato estructurado JSON.

En la Figura [3.1](#) se muestra el flujo del proceso de web scraping empleado para la extracción automatizada de recetas desde los sitios web. El procedimiento comienza con la ejecución del script `scraper.py`, al cual se le proporciona como parámetro la URL del sitio web seleccionado. Posteriormente, se lanzan subprocesos en paralelo mediante los cuales se ejecuta la función `recopilación`. Esta función se encarga de extraer cuatro elementos clave de cada receta: la URL original, el título, la lista de ingredientes y los pasos para la preparación.

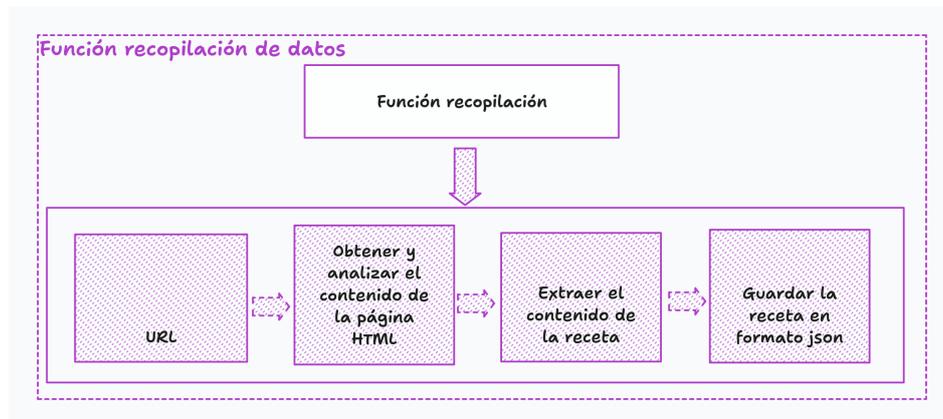
Figura 3.1

Flujo del proceso de web scraping para la extracción de datos



Nota: Fuente propia.

En la Figura [3.2](#), se observa el flujo de la función de recopilación, es aquí donde todo se ejecuta, ya que en esta función analiza el html de la página web, extrae el contenido del sitio web que contiene la receta y posteriormente almacena la receta en un formato json para su posterior preprocesamiento.

Figura 3.2*Flujo de la función de recopilación***Nota:** Fuente propia.

3.1.7. Formato de almacenamiento de las recetas tras el proceso de web scraping

El formato utilizado para almacenar las recetas tras el proceso de web scraping se presenta a continuación. Este esquema permite diferenciar claramente cada uno de los componentes de una receta, lo que facilita tanto su procesamiento automático como su posterior revisión manual.

- Url: cadena de tipo string que indica la dirección del sitio web de donde fue extraída la receta.
- Título: cadena de tipo string que representa el nombre de la receta.
- Ingredientes: arreglo (array) de cadenas de texto (string), donde cada elemento corresponde a un ingrediente utilizado en la preparación de la receta.
- Instrucciones: arreglo (array) de cadenas de texto (string) que contiene los pasos o instrucciones secuenciales necesarios para elaborar la receta.

3.2. Preparación del dataset de gastronomía peruana

3.2.1. Proceso de limpieza manual de los archivos JSON

Para esta etapa del desarrollo del proyecto, la primera tarea consistió en realizar la limpieza manual de cada archivo JSON generado a partir del proceso de web scraping aplicado a los distintos sitios web. Con el objetivo de facilitar la edición y revisión de los datos, los archivos fueron temporalmente convertidos al formato CSV, permitiendo una manipulación más accesible. Una vez finalizado el proceso de revisión, los datos se volvieron a guardar en su formato original JSON.

La validación de cada receta se llevó a cabo mediante la apertura manual del sitio web correspondiente en una nueva pestaña del navegador. Esto permitió comparar el contenido original con la información extraída, verificando así la integridad y completitud de cada receta. Durante este proceso, se establecieron criterios específicos para identificar y corregir posibles errores o, en su caso, eliminar aquellas recetas que no cumplieran con los estándares definidos.

- Se eliminaron las recetas incompletas, es decir, aquellas que no contenían alguno de los tres componentes esenciales: título, ingredientes o instrucciones, sin importar el orden.
- Se descartaron recetas redundantes o con instrucciones excesivamente extensas que dificultaban su procesamiento.
- Se suprimieron fragmentos de texto en la sección de instrucciones que incluían consejos, explicaciones sobre el origen de la receta o referencias a imágenes, ya que el proyecto se centra exclusivamente en el uso de texto.

- Se editaron expresiones con diminutivos en los ingredientes e instrucciones (por ejemplo: poquito, cucharadita, doraditas, sabrositas, calentito), reemplazándolas por formas más estándar y uniformes (como poco a poco, cucharada, doradas, sabrosas, caliente).
- Se reformularon instrucciones demasiado largas, sustituyendo estructuras extensas por oraciones más breves que conservaran el mismo significado y acción.
- Se corrigieron errores ortográficos presentes en los textos de las recetas.
- Se segmentaron instrucciones que estaban escritas como un único bloque de texto, dividiéndolas en líneas separadas para mejorar su legibilidad y procesamiento.
- Se eliminaron descripciones adicionales o recomendaciones en la lista de ingredientes, priorizando la mención precisa del insumo. Por ejemplo: 1kg de papa fresca fue simplificado a 1kg de papa, y 1kg de pollo fresco/de corral a 1kg de pollo.

3.3. Exploración del dataset

En esta etapa de exploración se realiza un análisis y visualización del contenido del dataset, con el objetivo de obtener una comprensión general de sus características. Este análisis permite identificar la frecuencia de aparición de ingredientes e instrucciones, así como los ingredientes más utilizados y las palabras más comunes en la preparación de las recetas.

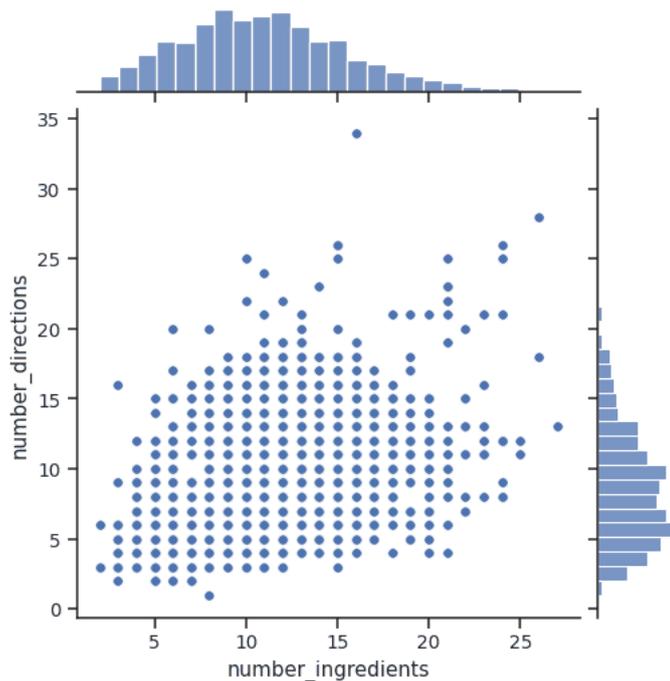
La exploración preliminar resulta fundamental no solo para conocer la estructura y distribución de los datos, sino también para orientar las decisiones en el proceso de preprocesamiento. A partir de este diagnóstico inicial, es posible diseñar e implementar

funciones adicionales de limpieza que optimicen la calidad del dataset para su uso en etapas posteriores del proyecto.

En la Figura 3.3, se muestra la distribución de la cantidad de instrucciones e ingredientes presentes en las recetas del dataset.

Figura 3.3

Relación entre la frecuencia de instrucciones y el número de ingredientes



Nota: Fuente propia.

En la Figura 3.4, se realizó una exploración de las palabras más frecuentes presentes en los ingredientes de las recetas del dataset. Como resultado, se presenta una nube de palabras (WordCloud) que visualiza de forma intuitiva y destacada los términos más comunes utilizados en la descripción de los ingredientes.

3.4. Preprocesamiento del dataset

El preprocesamiento es una etapa clave para preparar el dataset con el fin de garantizar su calidad y coherencia antes del modelado. En esta fase se llevan a cabo tareas como la limpieza de datos, normalización de texto, tokenización y procesamiento específico de los campos de ingredientes e instrucciones.

En la Figura 3.6 se muestra la estructura del dataset, el cual está compuesto por tres componentes principales: título (title), ingredientes (ingredients) e instrucciones (directions). Por otro lado, la Figura 3.7 presenta un resumen que permite visualizar la cantidad de recetas únicas y repetidas contenidas en el dataset.

Figura 3.6

Dataset gastronomía peruana

	title	ingredients	directions
0	Sangría peruana	[1 litro de Vino tinto, ½ litro de Gaseosa de ...	[cortar la manzana y pera en cubos pequeños, ...
1	Receta de Tallarines verdes peruanos	[1 manojo de espinacas, 1 rama de albahaca, 10...	[Separa las hojas de los tallos, tanto de la...
2	Receta de Aji de atún	[½ taza de Aceite, 1 Cebolla, 2 dientes de Ajo...	[Verter la leche y el agua en un recipiente, ...
3	Receta de Causa limeña	[3 ajíes amarillos, 1 kilogramo de papas amari...	[Pelar las papas y llévar a hervir con un poc...
4	Receta de Tallarines verdes	[1 paquete de Fideos, 1 pizca de Sal, 500 gram...	[En una olla poner 2 litros y medio de agua co...

Nota: Fuente propia.

Figura 3.7

Dataset resumen

Dataset size BEFORE validation 1440
 Dataset size AFTER validation 1193
 Number of incomplete recipes 247

	title	ingredients	directions
0	sangría peruana	[1 litro de vino tinto, ½ litro de gaseosa de ...	[cortar la manzana y pera en cubos pequeños, e...
1	tallarines verdes peruanos	[1 manojo de espinacas, 1 rama de albahaca, 10...	[separa las hojas de los tallos tanto de las e...
2	ají de atún	[½ taza de aceite, 1 cebolla, 2 dientes de ajo...	[verter la leche y el agua en un recipiente re...
4	tallarines verdes	[1 paquete de fideos, 1 pizca de sal, 500 gram...	[en una olla poner 2 litros y medio de agua co...
5	humitas de choclo	[½ docena de choclos, ½ cebolla rallada, 2 cu...	[poner aceite en una sartén y freír la cebolla...

Nota: Fuente propia.

3.4.1. Limpieza automática de datos

En esta etapa se implementaron diversas funciones destinadas a la limpieza y estandarización del dataset. A continuación, se detallan los principales procedimientos aplicados:

- Eliminación de recetas vacías, aunque este procedimiento también se abordó durante la limpieza manual, algunos casos pasaron inadvertidos debido al cansancio ocular. Por ello, se implementó una función automatizada para eliminar aquellas recetas que carecieran de alguno de los componentes esenciales: título, ingredientes o instrucciones, sin importar el orden. La presencia de registros incompletos podría afectar negativamente el entrenamiento del modelo y comprometer la calidad de los resultados.
- Cambio a minúsculas el dataset, todo el contenido textual del dataset fue transformado a minúsculas con el objetivo de normalizar las recetas y evitar inconsistencias derivadas del uso desigual de mayúsculas.
- Eliminación de caracteres especiales del dataset, se eliminaron los caracteres espe-

ciales del texto con el fin de reducir el ruido en los datos y facilitar su procesamiento durante el entrenamiento del modelo.

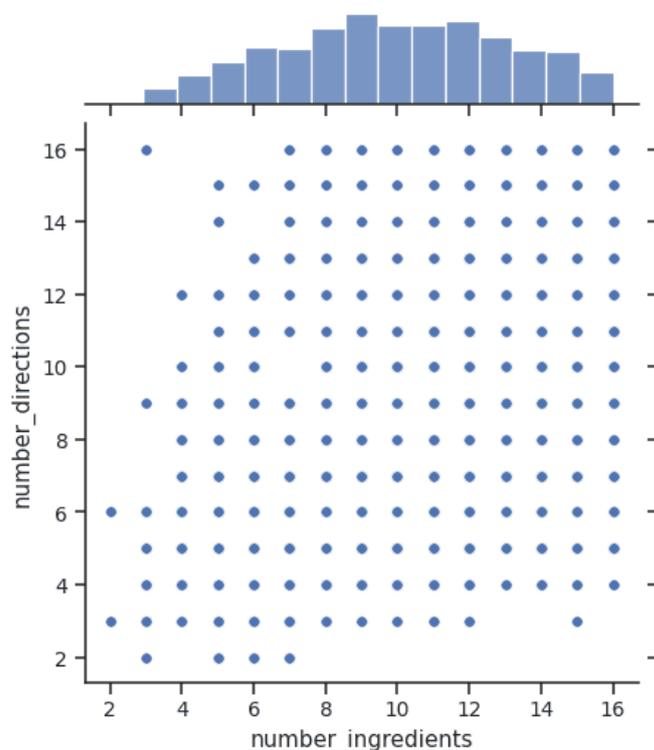
- Filtrado de recetas extensas, se ajustó el tamaño de cada receta estableciendo límites predefinidos tanto para los ingredientes como para las instrucciones. En el caso de los ingredientes, se consideró un mínimo de 2 y un máximo de 16 elementos. Para las instrucciones, se fijaron también entre 2 y 16 pasos. Esta medida busca mejorar la eficiencia del entrenamiento del modelo, evitando recetas excesivamente largas y favoreciendo la generación de resultados concisos y fáciles de replicar.
- Eliminación de la palabra receta en el título, se eliminó la expresión "receta de" de los títulos, al considerarse redundante y no aportar información relevante. En el contexto de este trabajo, dicha expresión se considera ruido que podría afectar negativamente la calidad en la generación automática de recetas.

En resumen, la aplicación de estos tratamientos durante el preprocesamiento de los datos resulta fundamental para garantizar la coherencia y la correcta estructuración del dataset antes de su uso en el entrenamiento del modelo y la posterior generación de recetas. Cada uno de estos pasos aborda aspectos específicos, desde la eliminación de información irrelevante hasta la adaptación de los datos a un formato estandarizado que facilite su procesamiento por parte del modelo.

Finalmente, como resultado del preprocesamiento, en la Figura [3.8](#) se presenta la distribución de la cantidad de ingredientes e instrucciones por receta. Esta visualización permite observar cómo se estructuran las recetas tras la aplicación de los criterios de limpieza y filtrado definidos en esta etapa.

Figura 3.8

Relación entre la frecuencia de ingredientes y la cantidad de instrucciones en el dataset, despues de la limpieza automática



Nota: Fuente propia.

3.5. Personalización del modelo NER para la extracción de entidades

En esta etapa del proyecto se llevó a cabo la personalización de un modelo de reconocimiento de entidades nombradas (Named Entity Recognition, NER) utilizando la biblioteca spaCy¹, con el objetivo principal de extraer automáticamente los ingredientes presentes en las recetas del dataset. Esta tarea resulta fundamental para la generación automática de recetas, ya que permite establecer correlaciones entre los ingredientes y las instrucciones de preparación, mejorando así la coherencia y relevancia del contenido

¹<https://pypi.org/project/spacy/>

generado.

La entrada para el modelo consiste en una lista de ingredientes extraídos de cada receta, donde se requiere específicamente la entidad FOOD, correspondiente al nombre del ingrediente. No obstante, muchos de estos ingredientes contienen información adicional como cantidades y cualidades, por ejemplo: “250 gramos de queso fresco, mozzarella, gouda”. En este caso, elementos como “250 gramos de” (cantidad) y “fresco, mozzarella, gouda” (cualidad) constituyen información secundaria que se considera irrelevante para esta etapa, ya que tales detalles deben ser generados posteriormente en las instrucciones. Por tanto, la correcta extracción y aislamiento de la entidad FOOD es esencial, pues permite establecer relaciones directas entre los ingredientes y las acciones descritas en la preparación, favoreciendo así la coherencia semántica en la generación automática de recetas.

3.5.1. Anotación de los ingredientes

Named Entity Recognizer Se anotaron manualmente las entidades FOOD en las listas de ingredientes utilizando la herramienta spaCy NER Text Annotator², la cual permite resaltar directamente en el texto qué elementos corresponden a la entidad deseada. En este caso, se etiquetaron únicamente los nombres de los ingredientes, excluyendo cantidades, unidades de medida o descriptores.

Para este proceso, se seleccionaron 450 recetas del dataset, a partir de las cuales se procesaron un total de 4,762 ingredientes individuales con el fin de extraer las entidades FOOD. Este conjunto anotado sirvió como base para el posterior entrenamiento del modelo NER personalizado.

²<https://arunmozhi.in/ner-annotator/>

En la Figura 3.9, se puede observar el proceso de anotación de las entidades FOOD en cada una de las listas de ingredientes correspondientes a las recetas seleccionadas. Este procedimiento se realizó de forma manual con el objetivo de garantizar la calidad y precisión en la extracción de las entidades, asegurando que únicamente se etiquetaran los elementos relevantes para el entrenamiento del modelo NER.

Figura 3.9

Anotación NER de la entidad FOOD en listas de ingredientes

1 unidad de **Manzana** FOOD ✖

Nota: Fuente propia.

3.5.2. Entrenamiento del modelo NER personalizado

En esta etapa se llevó a cabo el entrenamiento del modelo NER para la extracción de las entidades FOOD a partir de las listas de ingredientes. Previo al proceso de entrenamiento, se definió la métrica de evaluación que se utilizó para medir la calidad del modelo, la cual se detalla a continuación.

- Para evaluar la eficacia del modelo NER en la extracción de la entidad FOOD, se utilizó F1-Score, una métrica que combina precisión y sensibilidad en una sola medida, permitiendo valorar tanto los aciertos como los errores del modelo de forma equilibrada.
- Precisión, indica la proporción de predicciones positivas que son correctas respecto del total de predicciones positivas realizadas por el modelo, es decir, cuántas de las entidades identificadas como FOOD realmente lo son.
- Recall, mide la capacidad del modelo para identificar correctamente las instancias

positivas, es decir, la proporción de entidades FOOD correctamente detectadas entre todas las que realmente están presentes en el texto.

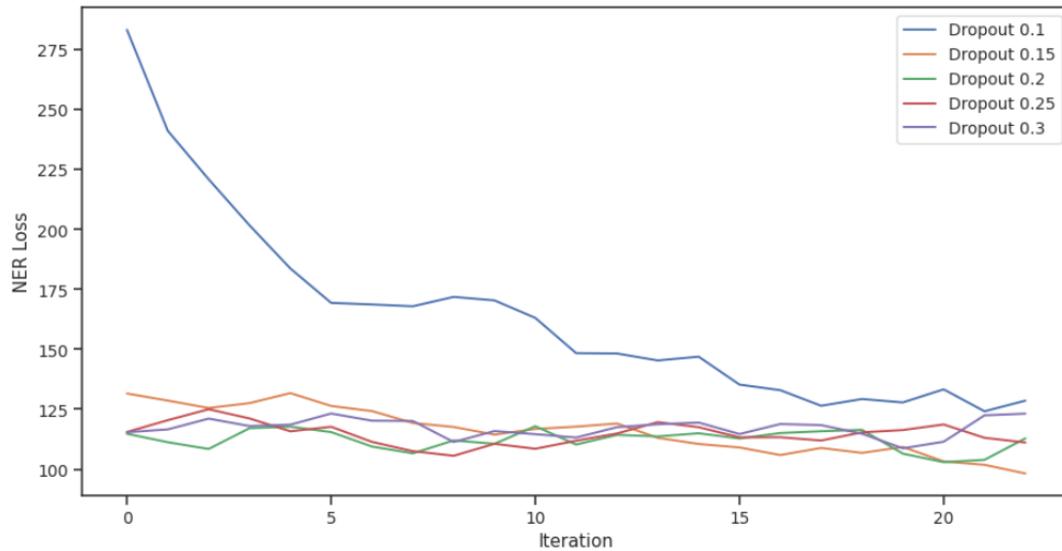
Para el entrenamiento, los datos se dividieron en una proporción del 80 % para el conjunto de entrenamiento (training) y 20 % para el conjunto de prueba (test). Asimismo, se definieron los parámetros del modelo empleados durante el proceso de ajuste. Estos son los siguientes:

- Dropout, se utiliza para prevenir el overfitting durante el entrenamiento del modelo. Consiste en desactivar aleatoriamente un porcentaje de neuronas en cada iteración, lo que obliga al modelo a no depender excesivamente de ciertas conexiones y, en consecuencia, mejora su capacidad de generalización.
- Número de épocas, se definió un total de 25 épocas, lo que implica que el modelo recorre el conjunto de entrenamiento completo 25 veces durante el proceso de aprendizaje.
- Tasa de aprendizaje, es un valor positivo que escala el gradiente de la función de pérdida con respecto a los parámetros del modelo. Controla la magnitud de los ajustes realizados a los pesos en cada iteración. Para este caso, se estableció una tasa de aprendizaje de 0.15, basado en la experimentación este valor dio mejores resultados.

En la Figura [3.10](#), se observa el rendimiento del modelo utilizando diferentes valores de dropout: 0.10, 0.15, 0.20, 0.25 y 0.30. Los resultados obtenidos durante el proceso de entrenamiento mostraron que el valor de 0.15 produjo un mejor comportamiento de la función de pérdida, evidenciando una mayor estabilidad y desempeño del modelo en la tarea de extracción de entidades.

Figura 3.10

Entrenamiento del modelo NER con distintos valores de dropout



Nota: Fuente propia.

3.6. Diseño e implementación del modelo

En esta etapa del proyecto se diseñó e implementó un modelo generativo basado en la arquitectura GPT-2, con el objetivo de generar recetas en español, tomando como entrada una lista de ingredientes. Para ello, se utilizó el modelo preentrenado DeepESP/gpt2-spanish, adaptado posteriormente a la tarea específica mediante técnicas de fine-tuning.

El diseño del modelo incluyó la definición de una estructura de entrada estandarizada, el uso de tokens especiales para delimitar secciones clave de las recetas, y la preparación de los datos mediante una función de transformación que convierte cada receta en un único texto continuo. La implementación fue realizada utilizando la biblioteca Transformers de Hugging Face y PyTorch, lo que permitió un entrenamiento flexible y eficiente sobre el conjunto de datos personalizado.

3.6.1. Etiquetas especiales

Con el propósito de adaptar el modelo preentrenado GPT-2 a la tarea específica de generación de recetas, se introdujeron tokens especiales que delimitan explícitamente la estructura del texto. Estos tokens, definidos como frases de control en minúsculas y encerradas entre corchetes angulares (por ejemplo, “<token >”), actúan como marcadores semánticos que orientan al modelo durante el proceso de entrenamiento y generación.

Para este trabajo de investigación, cada receta inicia con el token “<|startoftext|>”, seguido por la secuencia correspondiente a los ingredientes y posteriormente por las instrucciones de preparación. Finalmente, se incorpora el token de cierre “<|endoftext|>”. La inclusión de estos delimitadores personalizados tiene como finalidad facilitar la comprensión y reproducción de la estructura formal de una receta por parte del modelo, promoviendo así una generación más coherente y estructurada del contenido.

En la Tabla [3.2](#) se puede observar cada uno de los tokens especiales definidos, los cuales desempeñan un papel fundamental en la delimitación de la estructura de las recetas.

Tabla 3.2

Tokens de control personalizados en GPT-2

Token	Descripción
“< startoftext >”	Inicio del texto de la receta
“< endoftext >”	Fin del del texto de la receta
“< unknown >”	Etiqueta que se refiere a un token desconocido
“< pad >”	Etiqueta que se refiere a un token de padding (relleno)

Nota: Fuente propia.

En la Figura 3.11 se puede observar, el resultado de la preparación y estructuración de las recetas con los tokens de control para entrenar el modelo y posterior generación de recetas.

Figura 3.11

Ejemplo de una receta con tokens de control personalizados

```
'<|startoftext|>Ingredients:\nmasa de wantán, 1 filete de pierna de pollo, cebolla china, sal, pimienta blanca, azúcar, aceite eno moler 1 taza de filete de pierna de pollo sin piel y añadir 2 cucharadas de la parte blanca de la cebolla china picada en as de aceite de ajonjolí y gotas de sillao, rellenar la masa de wantán al medio con todo el relleno realizado previamente y d a masa y pegamos dos de las puntas del triángulo, freímos los wantanes con abundante aceite y fuego medio<|endoftext|>'
```

Nota: Fuente propia.

3.6.2. Selección del modelo

Para el presente trabajo de investigación se emplea el modelo GPT-2 preentrenado para el idioma español, específicamente la variante DeepESP/gpt2-spanish³

- DeepESPgpt2-spanish, este modelo utilizado corresponde a una arquitectura de generación de lenguaje entrenada exclusivamente con textos en español. Su entrenamiento se realizó sobre un corpus de 11.5GB, compuesto por 3.5GB de artículos de Wikipedia y 8GB de textos narrativos, que incluyen cuentos, teatro, poesía y ensayos.

3.6.3. Partición del dataset

Para el proceso de entrenamiento, el dataset de recetas fue dividido en dos subconjuntos: un 92 % se destinó al conjunto de entrenamiento, mientras que el 8 % restante se utilizó como conjunto de prueba (test set). Esta partición permitió evaluar el desempeño del modelo sobre datos no vistos durante el aprendizaje, garantizando así una medición más objetiva de su capacidad de generalización.

³<https://huggingface.co/DeepESP/gpt2-spanish>

3.6.4. Tokenización

Para la tarea de tokenización se utilizó el modelo GPT2 TokenizerFast de la biblioteca Transformers de Hugging Face. En esta etapa, se incorporaron tokens especiales personalizados, tal como se muestra en la Tabla 3.2, con el objetivo de proporcionar al modelo preentrenado un contexto estructurado y controlado sobre cada receta presente en el dataset. Estos tokens delimitan explícitamente las secciones fundamentales de una receta, como los ingredientes y las instrucciones, facilitando así la comprensión de su formato por parte del modelo.

Dado que el conjunto de datos utilizado para el entrenamiento específico difiere significativamente del corpus con el que fue preentrenado el modelo GPT-2, el proceso de tokenización cumple un rol crucial al permitir que el modelo interprete y genere recetas de manera coherente y conforme a la estructura esperada.

Sin embargo, se identificó una limitación durante este proceso: algunos ingredientes menos comunes, ausentes en el vocabulario original del modelo, fueron descompuestos en subpalabras durante la tokenización. Esta fragmentación puede afectar levemente la calidad de la generación si no se maneja adecuadamente en etapas posteriores.

Cabe destacar que el modelo GPT-2 admite un máximo de 1024 tokens por secuencia de entrada. En este trabajo, se optó por concatenar todas las recetas del dataset en una única secuencia de texto, utilizando los tokens especiales para separar cada receta. Esta estrategia permitió aprovechar eficientemente el contexto disponible por el modelo, manteniendo la coherencia estructural a lo largo de todo el corpus.

3.6.5. Fine Tuning

Como se mencionó anteriormente, en este trabajo de investigación se emplea el modelo preentrenado GPT-2 para el idioma español, específicamente la variante DeepESP/gpt2-spanish. Se llevará a cabo un proceso de ajuste fino (fine-tuning) sobre los hiperparámetros de la red neuronal con el fin de adaptar el modelo a nuestro conjunto de datos y permitir, posteriormente, la generación automática de recetas representativas de la gastronomía peruana.

3.6.6. Implementación del modelo y entrenamiento

Una vez tokenizado el conjunto de datos, este fue cargado en el modelo GPT-2 para proceder con el proceso de entrenamiento.

3.6.7. Recursos para el entrenamiento

En esta sección se describen las herramientas y tecnologías empleadas durante la implementación y entrenamiento del modelo generador de recetas de la gastronomía peruana:

- Pytorch, si bien el modelo GPT-2 fue entrenado utilizando el framework Tensorflow, para este trabajo de investigación se hizo uso del framework Pytorch.
- Google colab, el desarrollo del proyecto de investigación se realizó en la plataforma de google colab, donde se hizo uso de servidores gpu para todo el desarrollo del proyecto preprocesamiento, exploración del dataset, entrenamiento y pruebas.
- Hugging face transformer, es una biblioteca que proporciona una variedad de modelos pre entrenados para distintas tareas, es de acceso libre.

3.6.8. Entrenamiento

Para el entrenamiento del modelo, como se mencionó anteriormente, se adoptaron dos enfoques diferenciados. El primero corresponde al entrenamiento estándar, mientras que el segundo se refiere al entrenamiento con NER. Ambos enfoques utilizan los mismos parámetros de entrada, con una diferencia fundamental: en el enfoque NER, además de la lista completa de ingredientes, se incorpora una entrada adicional que contiene únicamente las entidades de tipo FOOD, extraídas previamente mediante el modelo NER personalizado.

La inclusión explícita de estas entidades permite al modelo establecer una relación más directa entre los ingredientes y las instrucciones generadas, lo cual favorece la coherencia y relevancia del contenido producido. Este enfoque busca mejorar la capacidad del modelo para generar instrucciones ajustadas a los ingredientes clave, promoviendo una mayor fidelidad en la estructura y lógica de las recetas.

Capítulo 4

Resultados

En esta sección se presentan los resultados obtenidos a partir del ajuste fino del modelo GPT-2 y sus variaciones, así como del modelo personalizado de reconocimiento de entidades nombradas (NER) para la extracción de entidades food a partir de la lista de ingredientes. Se analizaron cuatro configuraciones: GPT-2 Base, GPT-2 con ajuste fino (fine-tuning), GPT-2 Base + NER y GPT-2 con ajuste fino + NER.

Para evaluar el rendimiento del modelo NER personalizado, se emplearon las métricas de precisión, recall y F1-score. Por otro lado, la calidad de la generación de texto de las 4 variantes del modelo GPT-2 fue evaluada mediante las métricas BLEU, BERTScore y COMET. Asimismo, en esta sección se presentan los resultados obtenidos a partir de la evaluación realizada por los usuarios sobre la generación de recetas las 4 variantes del modelo GPT-2 a la gastronomía peruana.

4.1. Resultados del modelo personalizado de reconocimiento de entidades nombradas (NER)

- Cualitativo: En esta etapa, se implementa el modelo personalizado de reconocimiento de entidades nombradas (NER), previamente entrenado con la categoría FOOD,

con el objetivo de identificar automáticamente los ingredientes dentro de listas textuales. En la Figura 4.1, se muestra un ejemplo del resultado obtenido, donde se visualiza el proceso de extracción y anotación de las entidades food identificadas en cada entrada de la lista de ingredientes. Este procedimiento permite validar la capacidad del modelo para reconocer ingredientes de manera estructurada, facilitando su posterior procesamiento y uso en la generación de las recetas.

Figura 4.1

Extracción de entidades FOOD a partir de la lista de ingredientes

```
6 unidades de Plátano maduro Entities [('Plátano', 'FOOD')]
2 cucharadas soperas de Salsa de tomate Entities [('Salsa de tomate', 'FOOD')]
4 unidades de papas amarillas o blancas Entities [('papas', 'FOOD')]
500 gramos de Espinacas Entities [('Espinacas', 'FOOD')]
1 pizca de Sal Entities [('Sal', 'FOOD')]
5 ajíes Entities [('ajíes', 'FOOD')]
½ taza de Queso parmesano Entities [('Queso', 'FOOD')]
1 pizca de Sal Entities [('Sal', 'FOOD')]
1 pizca de pimienta Entities [('pimienta', 'FOOD')]
1 litro de Vino tinto Entities [('Vino', 'FOOD')]
2 dientes de Ajo molidos Entities [('Ajo', 'FOOD')]
2 unidades de Huevos cocidos Entities [('Huevos', 'FOOD')]
5 unidades de Jitomates gordos Entities [('Jitomates', 'FOOD')]
3 limones Entities []
1 pizca de Pimienta Entities [('Pimienta', 'FOOD')]
```

Nota: Fuente propia.

- **Cuantitativo:** En la Tabla 4.1, se muestra los resultados cuantitativos obtenidos por el modelo personalizado de reconocimiento de entidades nombradas (NER) en la tarea de identificación de entidades de tipo FOOD. Las métricas utilizadas para medir el rendimiento fueran precision, recall y f1-score. El modelo alcanzó un valor de precisión de 0.902, un recall de 0.900 y un f1-score de 0.903.

Tabla 4.1*Desempeño del modelo NER en la extracción de entidades FOOD*

precision	recall	f1-score
0.902	0.900	0.903

Nota: Fuente propia.

4.2. Evaluación del modelo GPT-2 y sus variaciones

Con el objetivo de evaluar el desempeño del modelo GPT-2 y sus variantes en la tarea de generación de recetas de la gastronomía peruana, se emplearon las métricas BLEU, BERT-Score y COMET. Estas métricas permiten analizar la calidad del texto generado desde distintas perspectivas, como la correspondencia léxica, la similitud semántica y la adecuación contextual.

4.2.1. Formato ejemplo de receta base utilizada para comparar con las recetas generadas

Ingredientes:

- $\frac{1}{2}$ kilo de pechugas de pollo deshuesadas
- $\frac{1}{4}$ taza de aceite vegetal
- 2 cebollas medianas
- $\frac{1}{2}$ cucharada de ajo molido
- 2 ajíes molidos
- 1 cucharada de palillo
- $\frac{1}{2}$ taza de caldo de pollo

- 2 tazas de papa
- 1 cucharada de jugo de limón
- 1 $\frac{1}{2}$ 2 cucharadas de hierba buena
- sal
- pimienta
- pizca de comino

Instrucciones:

1. calentar el aceite en una olla mediana
2. agregar la cebolla y freír a fuego medio hasta que esté suave
3. incorporar los ajos ají molido y palillo
4. mezclar y cocinar un minuto más
5. sazonar con sal pimienta y comino al gusto
6. agregar el caldo de pollo pollo y papa pelada en cuadrados
7. llevar a hervir a fuego medio hasta que el pollo y las papas estén cocidos
8. agregar la hierba buena jugo de limón y mezclar
9. rectificar la sazón
10. servir con arroz blanco

4.2.2. Métrica BLEU

Bilingual Evaluation Understudy (BLEU) permite cuantificar la calidad de un texto generado a partir de la comparación con frases de referencia. En este caso, se comparó `ground_truth_recipe` (receta base) con la receta generada por el modelo: GPT-2 Base, GPT-2+Ajuste fino (fine-tuning), GPT-2 Base+NER y GPT-2+Ajuste fino (fine-tuning)+NER (recetas generadas por cada modelo se visualiza en anexos A). En la Tabla [4.2](#) se presentan los valores obtenidos mediante la métrica BLEU.

Tabla 4.2

Valores obtenidos con la métrica BLEU para las cuatro variantes del modelo GPT-2

Modelo	Valores
GPT-2 Base	0,1575
GPT-2+Ajuste fino(fine-tuning)	0,3239
GPT-2 Base+NER	0,0894
GPT-2+Ajuste fino(fine-tuning)+NER	0,1980

Nota: Fuente propia.

4.2.3. Métrica BERTScore

BERTScore es una métrica que permite evaluar la calidad semántica de un texto generado mediante la comparación contextual entre la receta de referencia (`ground_truth_recipe`) y la receta generada por los modelos: GPT-2 Base, GPT-2 con ajuste fino (fine-tuning), GPT-2 Base + NER y GPT-2 con ajuste fino + NER. A diferencia de métricas tradicionales como BLEU, BERTScore se basa en representaciones vectoriales contextualizadas, lo que permite una comparación más precisa del significado de las palabras y frases (Zhang et al., [2020](#)).

En la Tabla [4.3](#) se presentan los valores obtenidos mediante la métrica BERTScore para cada una de las variantes del modelo (las recetas generadas se visualizan en el Anexo

A).

Tabla 4.3

Valores obtenidos con la métrica BERTScore para las cuatro variantes del modelo GPT-2

Modelo	Valores
GPT2 Base	0,6638
GPT2+Fine-tuning	0,8017
GPT2 Base+NER	0,6020
GPT2+Fine-tuning + NER	0,7407

Nota: Fuente propia.

4.2.4. Métrica COMET

La métrica COMET es una métrica de evaluación automática basada en modelos de aprendizaje profundo que considera tanto la fluidez como la adecuación semántica del texto generado respecto a una referencia. A diferencia de métricas tradicionales como BLEU, COMET utiliza representaciones contextuales y un modelo entrenado para predecir juicios humanos de calidad, lo que la convierte en una herramienta robusta para evaluar generación de texto (Rei et al., 2020).

En este estudio, se comparó la receta base (ground.truth.recipe) con las recetas generadas por los modelos: GPT-2 Base, GPT-2 con ajuste fino (fine-tuning), GPT-2 Base + NER y GPT-2 con ajuste fino + NER. En la Tabla 4.4 se presentan los valores obtenidos mediante la métrica BERTScore para cada una de las variantes del modelo (las recetas generadas se visualizan en el Anexo A).

Tabla 4.4

Valores obtenidos con la métrica COMET para las cuatro variantes del modelo GPT-2

Modelo	Valores
GPT2 Base	0,3440
GPT2+Fine-tuning	0,5305
GPT2 Base+NER	0,3034
GPT2+Fine-tuning + NER	0,4650

Nota: Fuente propia.

4.3. Comparación de Desempeño entre Variantes del Modelo GPT-2 en la Generación de Recetas

La Tabla [4.5](#) presenta los resultados obtenidos por las cuatro variantes del modelo GPT-2 en la tarea de generación de recetas de la gastronomía peruana. Las variantes evaluadas son: GPT-2 Base, GPT-2 con ajuste fino (fine-tuning), GPT-2 Base con extracción de entidades nombradas tipo food (NER), y GPT-2 con ajuste fino + NER. Estas configuraciones fueron evaluadas utilizando las métricas automáticas BLEU, BERTScore y COMET, ampliamente empleadas en tareas de evaluación de generación de lenguaje natural.

Los resultados indican que la variante GPT-2 con ajuste fino sin NER alcanza el mejor rendimiento en todas las métricas consideradas, con especial destaque en BERTScore (0,8017) y COMET (0,5305), lo que evidencia una mayor fluidez y calidad semántica en las recetas generadas. En contraste, el modelo GPT-2 Base + NER obtuvo los valores más bajos, especialmente en BLEU (0,0894), lo que sugiere que la incorporación de entidades reconocidas sin un ajuste fino previo no es suficiente para mejorar el desempeño del modelo en tareas de generación de texto.

Tabla 4.5

Comparación del desempeño de las variantes del modelo GPT-2 en la generación de recetas, evaluadas mediante las métricas BLEU, BERTScore y COMET

Modelo	BLEU	BERT-Score	COMET
GPT2 Base	0,1575	0,6638	0,3440
GPT2+Fine-tuning	0,3239	0,8017	0,5305
GPT2 Base+NER	0,0894	0,6020	0,3034
GPT2+Fine-tuning + NER	0,1980	0,7407	0,4650

Nota: Fuente propia.

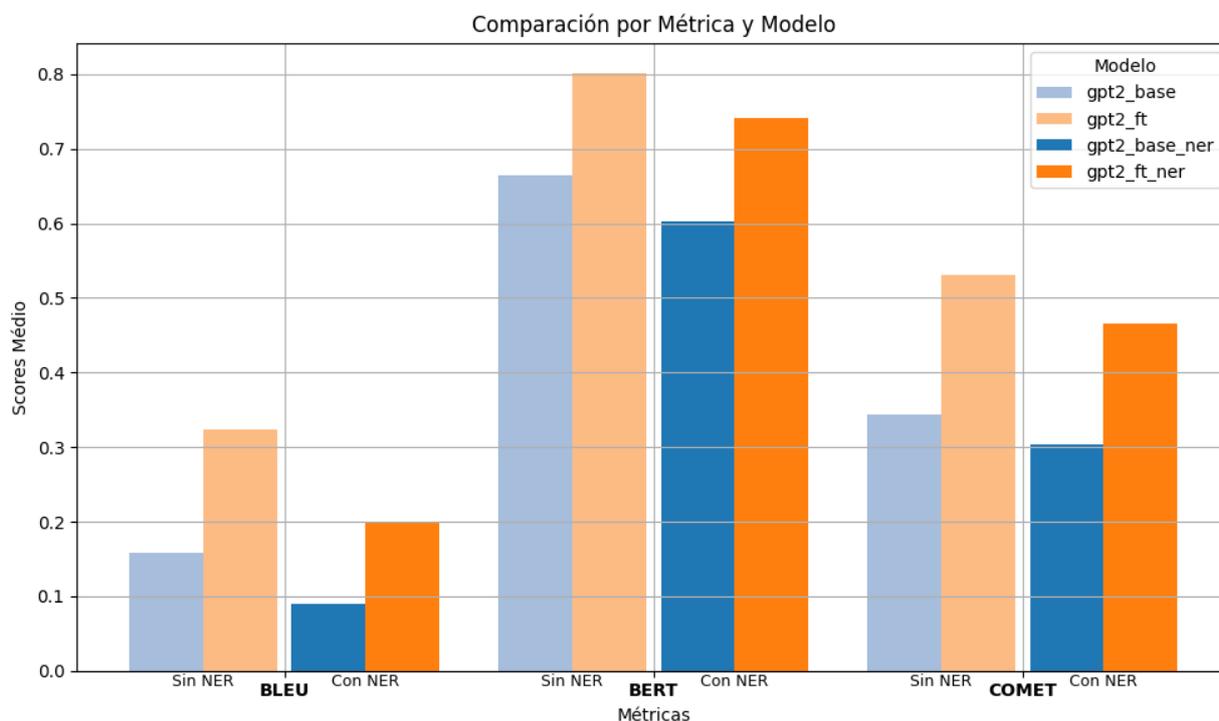
La Figura [4.2](#) presenta una comparación visual del rendimiento de las distintas variantes del modelo GPT-2 en la tarea de generación de recetas de la gastronomía peruana. Los modelos fueron evaluados mediante tres métricas ampliamente utilizadas en la evaluación de generación de texto: BLEU, BERTScore y COMET.

Los resultados muestran que la variante GPT-2 con ajuste fino (fine-tuning) alcanza los puntajes más altos en las tres métricas, evidenciando una mejor capacidad del modelo para generar recetas coherentes, relevantes y semánticamente similares a las recetas de referencia. Le sigue la variante GPT-2 con ajuste fino + NER, que también muestra un rendimiento elevado, aunque ligeramente inferior. En contraste, la variante GPT-2 Base con NER supera marginalmente al modelo GPT-2 Base, que obtuvo el desempeño más bajo en todas las métricas.

Estos hallazgos refuerzan la importancia del ajuste fino sobre datos específicos del dominio culinario peruano, al permitir que el modelo aprenda patrones más relevantes para la tarea, mejorando así la coherencia y la calidad del texto generado.

Figura 4.2

Comparación visual del desempeño de las variantes del modelo GPT-2 en la generación de recetas, evaluadas mediante las métricas BLEU, BERTScore y COMET



Nota: Fuente propia

4.4. Evaluación del modelo con usuarios

Para evaluar la percepción de los usuarios sobre las recetas generadas, se elaboró un formulario en el que se les pidió identificar si una receta había sido creada por un humano o por un computador. Se utilizaron 10 recetas seleccionadas aleatoriamente: 5 extraídas del conjunto de entrenamiento (train) y 5 generadas por el modelo entrenado con y sin NER. Las recetas creadas por humanos y por el modelo fueron distribuidas de forma aleatoria en el formulario.

A continuación, se presentan los resultados y el análisis detallado de la evaluación obtenida a partir de la encuesta.

4.4.1. Resultados de la evaluación obtenida

En la Tabla 4.6 se presenta un resumen de los resultados obtenidos a partir de la evaluación realizada por los usuarios respecto a la identificación de las recetas, indicando si consideraron que fueron generadas por un humano, por un computador o si no lograron diferenciarlas.

En total, participaron 18 personas en la encuesta, cuyas informaciones personales fueron debidamente anonimizadas. A continuación, se muestran los resultados obtenidos.

Tabla 4.6

Resultados de la evaluación de usuarios de las 10 recetas

Nro	Generada	Humano (%)	Computador (%)	No se diferencia (%)
Receta 1	Humano	72.2 %	11.1 %	16.7 %
Receta 2	Computador	38.9 %	33.3 %	27.8 %
Receta 3	Computador	27.8 %	44.4 %	27.8 %
Receta 4	Humano	50 %	11.1 %	38.9 %
Receta 5	Computador	38.9 %	33.3 %	27.8 %
Receta 6	Humano	44.4 %	22.2 %	33.3 %
Receta 7	Computador	55.6 %	27.8 %	16.7 %
Receta 8	Humano	50 %	5.6 %	44.4 %
Receta 9	Humano	44.4 %	27.8 %	27.8 %
Receta 10	Computador	44.4 %	16.7 %	38.9 %

Nota: Fuente propia.

4.4.2. Resumen general de resultados

En la Tabla 4.7 se presenta un resumen general de la evaluación, incluyendo el número de usuarios que participaron en la encuesta, la cantidad de recetas generadas por humanos y por computador, así como el porcentaje de aciertos en la identificación de recetas como humanas, generadas por computador y no diferenciadas. En la Tabla 4.8 se

Tabla 4.7

Resumen general de la encuesta

Descripcion	Valores
Número de usuarios participantes en la encuesta	18
Número total de recetas	10
Número de recetas generadas por humanos	5
Número de recetas generadas por computador	5
Porcentaje de recetas identificadas como generadas por humanos	52.2 %
Porcentaje de recetas identificadas como generadas por computador	41.12 %

Nota: Fuente propia.

presentan los porcentajes de aciertos en los casos en que las recetas fueron correctamente identificadas como generadas por un humano o por un computador. Asimismo, se muestran los porcentajes de error, es decir, cuando una receta generada por un humano fue clasificada incorrectamente como generada por un computador, o viceversa. También se incluye el porcentaje de casos en los que los usuarios no lograron diferenciar si la receta había sido creada por un humano o por un computador.

Tabla 4.8

Resumen porcentaje de aciertos

	% Humano	% Computador	% No diferencia
Humano	52.20 %	15.56 %	32.24 %
Computador	31.10 %	41.12 %	27.78 %

Nota: Fuente propia.

Capítulo 5

Análisis de resultados

5.1. Resultados offline y con usuarios

5.1.1. Análisis comparativo del desempeño de los modelos

Los resultados presentados en la Tabla [4.5](#) y la Figura [4.2](#), permiten evidenciar diferencias sustanciales en el rendimiento de las distintas variantes del modelo GPT-2 en la tarea de generación de recetas. En particular, el modelo GPT-2 con ajuste fino (fine-tuning) se destaca de forma consistente al obtener los puntajes más altos en las métricas BLEU, BERTScore y COMET, lo cual sugiere una mayor capacidad del modelo para producir recetas más coherentes, gramaticalmente correctas y semánticamente alineadas con las referencias humanas.

Por otro lado, la incorporación del modelo de reconocimiento de entidades nombradas (NER) sobre el modelo base (GPT-2 Base + NER) no muestra mejoras significativas en el rendimiento, e incluso reduce los puntajes en algunas métricas. Esto puede deberse a que la extracción de entidades por sí sola no es suficiente para enriquecer la generación de texto si no está acompañada de un proceso de ajuste fino que permita al modelo integrar de manera efectiva dicha información en la estructura semántica del contenido generado.

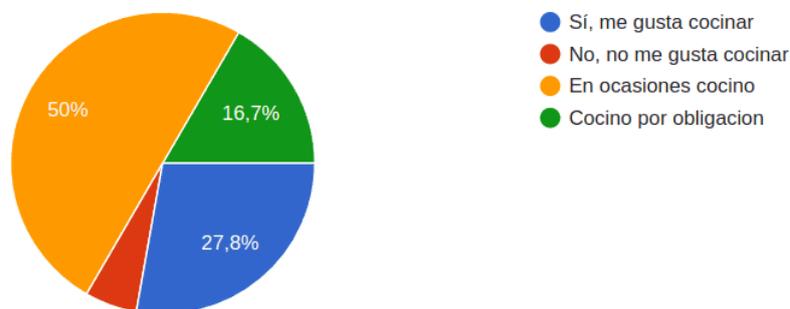
Asimismo, la combinación de ajuste fino + NER presenta mejoras intermedias, aunque no logra superar al modelo con solo ajuste fino. Este comportamiento sugiere que, si bien NER puede aportar información útil, su integración debe realizarse cuidadosamente para no introducir ruido o desbalance en el modelo.

En conjunto, los resultados confirman que el proceso de ajuste fino sobre datos específicos del dominio (en este caso, recetas peruanas) es el factor que más contribuye a mejorar la calidad del texto generado. Esta observación respalda la hipótesis de que adaptar modelos de lenguaje general a contextos específicos puede aumentar significativamente su desempeño y utilidad práctica.

5.1.2. Análisis de resultados de la evaluación con usuarios

De acuerdo con los porcentajes presentados en la Tabla 4.8, se observa que más de la mitad de los usuarios lograron identificar correctamente las recetas generadas por humanos. Este resultado sugiere que los participantes poseen cierto nivel de experiencia en el ámbito culinario. La Figura 5.1 complementa esta información al mostrar que 17 de los 18 encuestados cocinan con alguna frecuencia, ya sea porque les gusta, lo hacen ocasionalmente o por obligación. Solo un pequeño porcentaje 5.5% indicó que no le gusta cocinar.

Asimismo, en la Figura 5.2 se detallan los motivos por los cuales los usuarios cocinan. Entre las razones más frecuentes se encuentran el gusto por la comida preparada en casa, el hábito de cocinar durante los fines de semana o cuando se dispone de tiempo, y la necesidad de seguir dietas restringidas. No obstante, también se evidencia la presencia de participantes que no saben cocinar, así como de personas que, aunque disfrutan de la actividad, no disponen del tiempo necesario debido a compromisos laborales.

Figura 5.1*Preferencia de los usuarios por la actividad de cocinar***Nota:** Fuente propia**Figura 5.2***Motivos por los que los usuarios cocinan o no cocinan***Nota:** Fuente propia

En la Tabla [4.8](#), los resultados reflejan la capacidad de los usuarios para identificar correctamente el origen de las recetas (humano o computador). En el caso de las recetas realmente generadas por humanos, el 52.20 % fue identificado correctamente como tal, mientras que el 15.56 % fue erróneamente atribuido a un computador, y el 32.24 % de los participantes no logró diferenciar su origen.

Por otro lado, para las recetas generadas por el computador, el 41.12 % fue correctamente identificado, mientras que el 31.10 % fue confundido con recetas humanas, y el

27.78 % no pudo ser clasificado por los usuarios.

Estos resultados sugieren que, aunque las recetas humanas tienden a ser reconocidas con mayor precisión, las recetas generadas por el modelo GPT-2 con ajuste fino, alcanzan un nivel de naturalidad suficientemente alto como para inducir confusión. El porcentaje significativo de respuestas “No diferencia” en ambos casos indica que la frontera entre lo humano y lo generado automáticamente no siempre es evidente para los usuarios, lo cual puede ser interpretado como una señal positiva respecto al desempeño del modelo generador.

Conclusiones

- Se construyó un conjunto de datos representativo de la gastronomía peruana en idioma español, mediante la recopilación de recetas provenientes de diversas regiones del país.
- El preprocesamiento del conjunto de datos se llevó a cabo de forma semiautomática, combinando etapas manuales con funciones programadas para la estandarización de los registros. Este proceso permitió depurar y homogeneizar la información, lo que resultó en un conjunto final de 1,440 recetas, caracterizado por una mayor consistencia y calidad.
- Se implementaron técnicas personalizadas de reconocimiento de entidades nombradas (NER) utilizando la biblioteca spaCy, entrenadas con etiquetas diseñadas específicamente para el dominio de la gastronomía peruana. El modelo resultante permitió la extracción automatizada y precisa de ingredientes relevantes, clasificados como entidades del tipo FOOD.
- Se seleccionó el modelo de lenguaje preentrenado GPT-2, al cual se aplicaron técnicas de ajuste fino (fine-tuning) con el objetivo de adaptarlo a la tarea de generación de recetas en el ámbito de la gastronomía peruana. Para su evaluación, se consideraron cuatro configuraciones:
 - GPT-2 Base,
 - GPT-2 con ajuste fino (fine-tuning),

- GPT-2 Base con reconocimiento de entidades nombradas (NER) y
- GPT-2 con ajuste fino (fine-tuning) más NER.

La variante con ajuste fino obtuvo un desempeño notablemente superior frente a las demás, como lo evidencian las métricas BLEU, BERTScore y COMET, que reflejaron mejoras significativas en coherencia sintáctica, adecuación semántica y similitud contextual respecto a las recetas de referencia.

- Se evaluaron cuatro variantes del modelo GPT-2: la versión base, la base con incorporación de reconocimiento de entidades (NER), la versión con ajuste fino (fine-tuning) y la combinación de ajuste fino con NER. Para ello, se emplearon métricas automáticas (BLEU, BERTScore y COMET), con el objetivo de medir la calidad sintáctica, semántica y contextual de las recetas generadas. Los resultados demostraron que el modelo con ajuste fino obtuvo el mejor desempeño global (BLEU = 0,3239; BERTScore = 0,8017; COMET = 0,5305), destacando por su coherencia y adecuación al dominio de la gastronomía peruana. En comparación, el modelo base mostró un rendimiento significativamente inferior, mientras que la integración de NER sin ajuste fino no aportó mejoras relevantes. La variante que combinó ajuste fino con NER presentó un desempeño intermedio, lo que confirma que el ajuste fino fue el principal factor que contribuyó a la mejora en la generación de recetas.

Por otro lado, la validación con usuarios reveló que el 31,10 % de las recetas generadas fueron percibidas como creadas por humanos, y en el 27,78 % de los casos no se pudo distinguir su origen, lo que sugiere un alto nivel de naturalidad en los textos producidos por el modelo.

Recomendaciones

A partir de la experiencia adquirida durante la realización de este proyecto de investigación, se proponen las siguientes recomendaciones, con el objetivo de orientar a futuros investigadores interesados en desarrollar trabajos similares:

- Incrementar el número de recetas en el conjunto de datos, con el propósito de mejorar la capacidad del modelo para generalizar y generar recetas más variadas y representativas.
- Emplear herramientas basadas en inteligencia artificial para automatizar el proceso de web scraping de recetas de la gastronomía peruana, con el objetivo de agilizar la recolección de datos y mejorar la eficiencia en la construcción del corpus.
- Contar con un equipo que disponga de unidad de procesamiento gráfico (**GPU**) para facilitar el entrenamiento eficiente del modelo y realizar experimentos en menor tiempo.
- Sustituir o complementar la limpieza manual del conjunto de datos por procesos automatizados mediante herramientas de inteligencia artificial, como ChatGPT, con el fin de reducir significativamente el tiempo de procesamiento y asegurar una mayor coherencia, calidad textual y estructuración adecuada de las recetas.
- Ampliar el número de participantes en la evaluación con usuarios, considerando diversos perfiles (edad, conocimientos culinarios, uso de tecnología), para obtener una perspectiva más representativa sobre la calidad de las recetas generadas.

Trabajos futuros

A partir de los resultados obtenidos y las limitaciones observadas en el desarrollo de este trabajo, se identifican diversas oportunidades para ampliar y mejorar la investigación en futuras etapas. Estas posibles líneas de trabajo permitirían fortalecer el enfoque propuesto, así como explorar nuevas aplicaciones y adaptaciones del modelo.

- Extensión del modelo a dietas específicas: Adaptar el modelo para generar recetas peruanas compatibles con distintas necesidades alimentarias, como dietas keto, veganas, sin gluten o bajas en carbohidratos. Esto permitiría diversificar la utilidad del sistema y atender a un público más amplio con restricciones o preferencias nutricionales.
- Desarrollo de generación controlada de recetas: Diseñar un sistema que permita generar recetas personalizadas a partir de parámetros específicos definidos por el usuario, tales como ingredientes disponibles, tipo de dieta, tiempo de preparación o nivel de dificultad. Esta funcionalidad aumentaría la flexibilidad y utilidad del modelo en contextos prácticos.
- Aplicaciones prácticas como asistentes de cocina: Integrar el modelo en aplicaciones móviles o asistentes virtuales orientados a la cocina, que puedan ofrecer sugerencias de recetas, pasos de preparación guiados o recomendaciones personalizadas según las preferencias del usuario y los ingredientes disponibles en casa.
- Adaptación del enfoque a otras cocinas regionales: Aplicar el enfoque desarrollado

a otras gastronomías locales de América Latina, como las de Chile, Bolivia y otros países hispanohablantes. Esto permitiría evaluar la portabilidad del modelo y su capacidad de adaptación cultural y lingüística a distintas tradiciones culinarias.

- Explorar otros modelos de lenguaje diseñados específicamente para el idioma español: Se propone investigar el uso de modelos entrenados exclusivamente en español, como MarIA, desarrollado por el Barcelona Supercomputing Center. Este tipo de modelos, al estar adaptados lingüística y culturalmente, podrían mejorar la fluidez, coherencia y adecuación de la generación de recetas.

BIBLIOGRAFÍA

- Berríos Hurtado, L. R. (2014). Desarrollo histórico de la gastronomía en el Perú.
- Bień, M., Gilski, M., Maciejewska, M., & Taisner, W. (2020). Cooking recipes generator utilizing a deep learning-based language model.
- Galanis, G. A., Nikolaos Ioannis Papakostas. (2022). An update on cooking recipe generation with Machine Learning and Natural Language Processing. *2022 IEEE World Conference on Applied Intelligence and Computing (AIC)*, 739-744.
- Goel, M., & Bagler, G. (2022). Computational gastronomy: A data science approach to food. *Journal of Biosciences*, 47(1), 12.
- Goel, M., Chakraborty, P., Ponnaganti, V., Khan, M., Tatipamala, S., Saini, A., & Bagler, G. (2022). Ratatouille: A tool for Novel Recipe Generation. *2022 IEEE 38th International Conference on Data Engineering Workshops (ICDEW)*, 107-110. <https://doi.org/10.1109/icdew55742.2022.00022>
- Guardia, S. B. (2020). Gastronomía Peruana: Patrimonio Cultural de la Humanidad.
- Hadi, M. U., Qureshi, R., Shah, A., Irfan, M., Zafar, A., Shaikh, M. B., Akhtar, N., Wu, J., Mirjalili, S., et al. (2023). A survey on large language models: Applications, challenges, limitations, and practical usage. *Authorea Preprints*, 3.
- Hernandez, A. Y. M., & Duque, F. J. V. (2020). Inteligencia artificial al servicio de la auditoría: una revisión sistemática de literatura. *Revista Ibérica de Sistemas e Tecnologías de Informação*, (E27), 213-226.

- Ian Goodfellow, A. C., Yoshua Bengio. (2016). *Deep Learning*. Massachusset Institute of Technology.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436-444.
- Lozada, J. (2014). Investigación aplicada: Definición, propiedad intelectual e industria. *CienciAmérica: Revista de divulgación científica de la Universidad Tecnológica Indoamérica*, 3(1), 47-50.
- Management Solutions. (2024). The Rise of Large Language Models: From Fundamentals to Application. <https://www.managementsolutions.com>
- Mitchell, R. (2018). *Web scraping with Python: Collecting more data from the modern web*. .o'Reilly Media, Inc.”
- Moolayil, J., Moolayil, J., & John, S. (2019). *Learn Keras for deep neural networks*. Springer.
- Naveed, H., Khan, A. U., Qiu, S., Saqib, M., Anwar, S., Usman, M., Akhtar, N., Barnes, N., & Mian, A. (2023). A comprehensive overview of large language models. *arXiv preprint arXiv:2307.06435*.
- Oficina de Marketing Institucional. (2023, febrero). *La Gastronomía Peruana en el Mundo*. Centro de Formación en Turismo (CENFOTUR). <https://www.cenfotur.edu.pe/2023/boletin-1-2023/la-gastronomia-peruana-en-el-mundo/>
- Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al. (2018). Improving language understanding by generative pre-training.
- Rei, R., Stewart, C., Farinha, A. C., & Lavie, A. (2020). COMET: A Neural Framework for MT Evaluation. <https://arxiv.org/abs/2009.09025>
- Taneja, K., Segal, R., & Goodwin, R. (2024). Monte Carlo tree search for recipe generation using GPT-2. *arXiv preprint arXiv:2401.05199*.

- Turing, A. M. (2012). Computing machinery and intelligence (1950). *The Essential Turing: the Ideas That Gave Birth to the Computer Age*, 433-464.
- Vajjala, S., Majumder, B., Gupta, A., & Surana, H. (2020). *Practical natural language processing: a comprehensive guide to building real-world NLP systems*. O'Reilly Media.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.
- Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2020). BERTScore: Evaluating Text Generation with BERT. <https://arxiv.org/abs/1904.09675>

Anexo A – Recetas generadas por el modelo GPT2 y variaciones

A.1 Receta generada por el modelo GPT2 Base

Ingredientes:

- $\frac{1}{2}$ kilo de pechugas de pollo deshuesadas
- $\frac{1}{4}$ taza de aceite vegetal
- 2 cebollas medianas
- $\frac{1}{2}$ cucharada de ajo molido
- 2 ajíes molidos
- 1 cucharada de palillo
- $\frac{1}{2}$ taza de caldo de pollo
- 2 tazas de papa
- 1 cucharada de jugo de limón
- $1\frac{1}{2}$ –2 cucharadas de hierba buena
- Sal
- Pimienta

- Pizca de comino

Instrucciones:

1. El aceite vegetal de pollo deshuesado y el ajo de pollo deshuesado son una parte integral de la producción del aceite vegetal. Las hortalizas son muy finas y contienen una mezcla de tomate, cebolla, cebolla, etc. , de la que se obtiene una gran cantidad de aceite vegetal de la que se obtiene una gran cantidad de aceite vegetal.
2. En la receta se recomienda el aceite vegetal de pollo deshuesado.
3. El aceite vegetal de pollo deshuesado y el ajo de pollo deshuesado son un ingrediente especial de la producción del aceite vegetal de pollo deshuesado.
4. Las verduras son muy finas y están muy ricas en aceite vegetal, pero también tienen propiedades muy especiales.
5. Añadir el aceite vegetal de pollo deshuesado en el aceite vegetal de pollo deshuesado y el ajo de pollo deshuesado es una receta muy popular en la cocina tradicional.
6. El aceite vegetal de pollo deshuesado y el ajo de pollo deshuesado son una receta muy popular en la cocina tradicional de la cocina tradicional.
7. El aceite vegetal de pollo deshuesado es una receta muy popular en la cocina tradicional.
8. En la receta se recomienda el aceite vegetal de pollo deshuesado.
9. En la receta se recomienda el aceite vegetal de pollo deshuesado.
10. El aceite vegetal de pollo deshuesado es un ingrediente especial del aceite vegetal de pollo deshuesado, que se suele reservar en la cocina tradicional.

11. El aceite vegetal de pollo deshuesado es un ingrediente especial en la cocina tradicional de la cocina tradicional.
12. La cebolla deshuesada es una receta muy popular en la cocina tradicional de la cocina tradicional.
13. El aceite vegetal de pollo deshuesado es una receta muy popular en la cocina tradicional de la cocina tradicional.
14. En la receta se recomienda el aceite vegetal de pollo deshuesado, la cebolla deshuesada y el ajo deshuesado.
15. En la receta se recomienda el aceite vegetal de pollo deshuesado.

A.2 Receta generada por el modelo GPT2+Ajuste fino(fine-tunning)

Ingredientes:

- $\frac{1}{2}$ kilo de pechugas de pollo deshuesadas
- $\frac{1}{4}$ taza de aceite vegetal
- 2 cebollas medianas
- $\frac{1}{2}$ cucharada de ajo molido
- 2 ajíes molidos
- 1 cucharada de palillo
- $\frac{1}{2}$ taza de caldo de pollo
- 2 tazas de papa
- 1 cucharada de jugo de limón
- $1\frac{1}{2}$ –2 cucharadas de hierba buena
- sal
- pimienta
- pizca de comino

Instrucciones:

1. Poner a hervir el pollo deshuesado en una olla mediana y gruesa

2. Cuando esté cocido añadir el aceite y la cebolla a la olla y dejar que se cocine a fuego lento
3. Agregar el ajo el ají molido y el palillo
4. Remover todo hasta que el pollo esté cocido
5. Agregar la papa cortada en trozos y el caldo que sobró de la olla
6. Dejar que se enfríe y licuar
7. Servir en un plato hondo con arroz blanco y papas sancochadas

A.3 Receta generada por el modelo GPT2 Base+NER

Ingredientes:

- Hierba buena
- Cebollas
- jugo limón
- Pimienta
- Aceite vegetal
- Comino
- Palillo
- Ajíes molidos
- Papa
- Ajo molido
- Caldo pollo
- Pechugas pollo
- Sal

Instrucciones:

1. Coloca el pollo en un plato y cocine los huevos con un tenedor.
2. Coloque el pollo en un plato y cocine los huevos con la cebolla.
3. Coloque el pollo en un plato y cocine los huevos con el tenedor.

4. Coloque los huevos en un plato y cocine los huevos con el tenedor.
5. Coloque los huevos en un plato y cocine los huevos con el tenedor.
6. Coloque los huevos en un plato y cocine los huevos con el tenedor.
7. Coloque los huevos en un plato y cocine los huevos con el tenedor.

A.4 Receta generada por el modelo GPT2+Ajuste fino(fine-tuning)+NER

Ingredientes:

- Hierba buena
- Cebollas
- jugo limón
- Pimienta
- Aceite vegetal
- Comino
- Palillo
- Ajíes molidos
- Papa
- Ajo molido
- Caldo pollo
- Pechugas pollo
- Sal

Instrucciones:

1. Cortar en trozos la cebolla en brunoise
2. sazonar con sal pimienta blanca y 1 cucharada de ajo molido

3. añadir el caldo y dejar que se cocine a fuego medio
4. cuando empiece a hervir el aceite añadir la cebolla picada y dejar que se cocine por 5 minutos
5. pasado el tiempo retirar la cebolla y el tomate que no se quemó
6. agregar el ajo molido pimienta comino y dejar que se cocine por 5 minutos
7. añadir el tomate picado y dejar que se cocine por 2 minutos
8. pasado el tiempo rectificar la sal añadir el caldo la crema de tomate y el perejil
9. añadir la crema de tomate y dejar cocinar por 5 minutos
10. servir el aliño de pollo peruano acompañando con arroz blanco y rodajas de tomate

Anexo B – Recetas utilizadas en el formulario de evaluación con usuarios



Seção 1 de 11

MODELO GENERADOR DE RECETAS PERSONALIZADAS EN EL IDIOMA ESPAÑOL DE LA GASTRONOMÍA PERUANA USANDO APRENDIZAJE PROFUNDO

Está siendo invitado(a) a participar en una investigación.

El propósito de esta investigación es determinar si una receta creada por un humano y una por computador pueden diferenciarse. La información que proporcione será anonimizada. El cuestionario tiene una duración aproximada de 8 minutos.

¿De acuerdo a tu rutina cocinas? *

- Sí, me gusta cocinar
- No, no me gusta cocinar
- En ocasiones cocino
- Cocino por obligación

Con base a tu respuesta anterior, cuéntanos tus motivos *

- Me gusta la comida preparada en casa
- No tengo tiempo por motivos de trabajo
- Tengo una dieta restringida por salud (no puedo comer lacteos, no puedo comer azucares, etc)
- Cocino fines de semana o cuando hay tiempo
- Prefiero comer en la calle(restaurantes y otros), me ahorra tiempo
- Comer en la calle (restaurantes y otros) , es muy costoso
- No me da bien cocinar se me quema hasta el agua
- Otro

Cuando cocinas, ¿te apoyas en algún recurso físico o digital? *

- Recetas familiares que se mantienen vivas, pasándose de generación en generación, tal como las preparaba la abuela
- Libros de cocina
- Videos en Youtube o sitios web de recetas
- Utilizo ChatGPT u otros asistentes
- Tengo un verdadero instinto culinario dentro de mí, sé como crear recetas

¿Qué opinas de la gastronomía peruana? ¿Cuáles son tus platos favoritos o más representativos? *

Sua resposta

Próxima

Limpar formulário

Receta 1

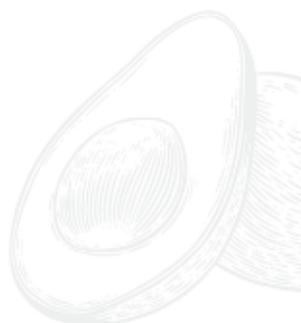
Receta 1 ¿Esta receta fue creada por una máquina o una persona? *



receta

INGREDIENTES

- 1 taza de arroz
- 1 lata de leche evaporada
- 1 lata de leche condensada
- Trozos de cáscara de naranja
- 2 palitos de canela
- 3 clavos de olor
- 1/2 taza de pasas rubias
- 1 cucharada de esencia de vainilla
- canela en polvo

**PREPARACIÓN**

1. Añadir 1 taza de arroz con 2 tazas de agua, agregar las cáscaras de naranja, previamente hervidas, cambiándolo el agua 5 veces para quitar el amargo de la cáscara de naranja
2. Agregar la canela y 3 clavos de olor, dejar que hierva hasta que el agua se consuma
3. Una vez que el arroz ya está seco, quitar las cáscaras, la canela y los clavos de olor
4. Añadir 1 taza de leche evaporada y dejar que hierva siempre moviendo todo para asegurar de que el arroz no se pegue en el fondo de la olla repitiendo este paso cada 5 minutos
5. Pasado unos 20 minutos a fuego lento, añadir 1 lata de leche condensada, dejando cocinar por 10 minutos más
6. Apagar el fuego y agregar 1/2 taza de pasas rubias o negras remojadas en licor
7. Añadir 1 cucharada de esencia de vainilla, servir al gusto espolvoreando canela en polvo para decorar



- No consigo diferenciar
- Humano
- Computador

Con base en tu respuesta anterior; ¿Probarías hacer esta receta? *

- Sí, la receta es corta, la receta parece deliciosa
- No, la receta es corta, la receta no parece deliciosa
- Sí, la receta es extensa, la receta parece deliciosa
- No, la receta es extensa, la receta no parece deliciosa

Voltar

Próxima

Limpar formulário

Receta 2

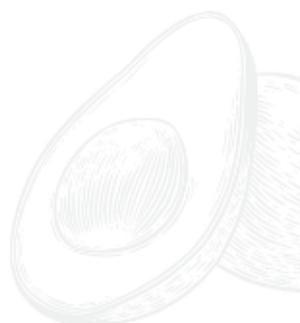
Receta 2 ¿Esta receta fue creada por una máquina o una persona? *



receta

INGREDIENTES

- 2 bistec
- aji amarillo limpio
- 2 tomates
- 1 cebolla
- perejil
- ½ vaso de caldo de ternera
- sal
- ajinomoto al gusto
- salsa de soja
- aceite
- vinagre
- pimienta
- comino
- cucharada de ajos molido



PREPARACIÓN

1. poner a cocinar los bistecs en la sartén con abundante agua, colar y reservar
2. En la misma sartén echar un poco de aceite a fuego medio y dorar la cebolla por 2 a 3 minutos
3. Añadir el ajo molido y mezclar por 2 minutos
4. Añadir el caldo a la sartén y mezclar por 2 minutos
5. Añadir el aji amarillo limpio y dejar que se sofrita
6. Añadir la cebolla a la vez y mezclar por 2 minutos



- Humano
- Computador
- No consigo diferenciar

Con base en tu respuesta anterior ¿Probarías hacer esta receta? *

- Si, la receta es corta, la receta parece deliciosa
- No, la receta es corta, la receta no parece deliciosa
- Si, la receta es extensa, la receta parece deliciosa
- No, la receta es extensa, la receta no parece deliciosa

[Voltar](#)

[Próxima](#)

[Limpar formulário](#)

Receta 3

Receta 3 ¿Esta receta fue creada por una máquina o una persona? *



receta

INGREDIENTES

- 3 oz de pisco
- 1 oz de jugo de limón
- 1 ½ oz de sirope de caña de azúcar
- 5 cubos de hielo
- ½ clara de huevo
- 1 gota de amargo de angostura

PREPARACIÓN

1. Colocar el hielo en una licuadora, el pisco, el jugo de limón, sirope de caña de azúcar y clara de huevo.
2. Licuar hasta que el hielo esté bien picado y colar el cóctel
3. Servir en copas y agregar amargo de angostura



- No consigo diferenciar
- Humano
- Computador

Con base en tu respuesta anterior ¿Probarías hacer esta receta? *

- Si, la receta es corta, la receta parece deliciosa
- No, la receta es corta, la receta no parece deliciosa
- Si, la receta es extensa, la receta parece deliciosa
- No, la receta es extensa, la receta no parece deliciosa

Voltar

Próxima

Limpar formulário

Receta 4

Receta 4 ¿Esta receta fue creada por una máquina o una persona? *



receta

INGREDIENTES

- 4 maracuyás
- 1 onza de pisco
- 1 onza de jarabe de goma
- 5 cubos de hielo

PREPARACIÓN

1. Corta los maracuyás por la mitad y con una cuchara retira la pulpa
2. Poner en un colador sobre un tazón y presionar con la cuchara para romper las burbujas de jugo que envuelven las semillas
3. Reservar el jugo y descartar las pepitas
4. En la licuadora procesar el jugo con el pisco, jarabe de goma y hielo



- No consigo diferenciar
- Humano
- Computador

Con base en tu respuesta anterior ¿Probarías hacer esta receta? *

- Si, la receta es corta, la receta parece deliciosa
- No, la receta es corta, la receta no parece deliciosa
- Si, la receta es extensa, la receta parece deliciosa
- No, la receta es extensa, la receta no parece deliciosa

Voltar

Próxima

Limpar formulário

Receta 5

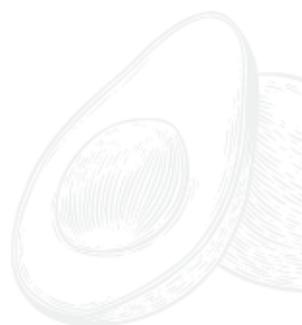
Receta 5 ¿Esta receta fue creada por una máquina o una persona? *



receta

INGREDIENTES

- 1 lb de filete de pescado
- Sal al gusto
- Pimienta al gusto
- Jugo de ½ limón
- ½ toza de harina sin preparar
- Aceite vegetal para freir
- Salsa tártara
- Salsa de aji
- Chifles

**PREPARACIÓN**

1. Corta el filete de pescado en láminas finas, de aproximadamente medio centímetro de grosor.
2. Sazona las láminas de pescado con sal, pimienta y jugo de limón. Deja reposar por unos minutos para que absorba los sabores
3. Pasa las láminas de pescado por la harina, cubriéndolas de manera uniforme
4. En una sartén a fuego medio, una vez caliente, fríe las láminas de pescado
5. Pasarlas a un lado sobre papel absorbente, servir sobre hojas de lechuga



- Computador
- Humano
- No consigo diferenciar

Con base en tu respuesta anterior ¿Probarías hacer esta receta? *

- Si, la receta es corta, la receta parece deliciosa
- No, la receta es corta, la receta no parece deliciosa
- Si, la receta es extensa, la receta parece deliciosa
- No, la receta es extensa, la receta no parece deliciosa

[Voltar](#)

[Próxima](#)

[Limpar formulário](#)

Receta 6

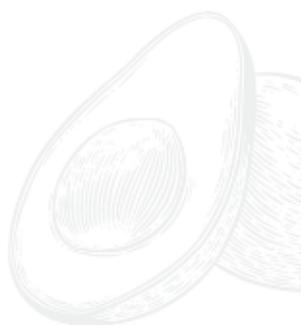
Receta 6 ¿Esta receta fue creada por una máquina o una persona? *



receta

INGREDIENTES

- 800 gr de calabaza
- 250 gr de harina
- 1 yema de huevo
- 150 gr de queso fresco
- 150 gr de queso rallado
- Sal
- Pimienta
- Nuez moscada
- Aceite de oliva
- Salsa para cubrir los ñoquis



PREPARACIÓN

1. Cocinar la calabaza al horno, al vapor o al microondas. No debe ser hervida así no absorbe agua. Debe quedar lo más seca posible.
2. Hacer un puré de calabaza y cuando esté tibia agregar la yema, la sal, la pimienta, la nuez moscada y la harina tamizada. Unirlo hasta formar una masa tierna.
3. Estirar en forma de cilindros de 2 cm. y dar forma de ñoquis.
4. Colocar el agua en la cacerola, tapar y llevar a fuego fuerte hasta que rompa el hervor.
5. Colocar la pasta y hervir a fuego mínimo hasta que suban a la superficie.
6. Apagar el fuego, retirar y colar.
7. Servir con salsa.



- No consigo diferenciar
- Computador
- Humano

Con base en tu respuesta anterior ¿Probarías hacer esta receta? *

- Sí, la receta es corta, la receta parece deliciosa
- No, la receta es corta, la receta no parece deliciosa
- Sí, la receta es extensa, la receta parece deliciosa
- No, la receta es extensa, la receta no parece deliciosa

[Voltar](#)

[Próxima](#)

[Limpar formulário](#)

Receta 7

Receta 7 ¿Esta receta fue creada por una máquina o una persona? *



receta

INGREDIENTES

- 1 pollo
- 1 cucharada sopera de aji panca
- 1 naranja
- 1 limón
- 1 pizca de sal
- 1 pizca de pimienta
- 25 ml de salsa de soja
- 2 cucharadas soperas de aceite de oliva
- 1 kilogramo de papa andina
- 1 cucharada sopera de vinagre
- 1 cucharada de jengibre en polvo



PREPARACIÓN

1. cortar el pollo en trozos pequeños reserva
2. En un tazón combina la cebolla los ajos y el jengibre, la salsa de soja
3. En una sartén, añadir el pollo y cocinar por 5 minutos sin dejar que se queme, incorporar la mezcla anterior, baja el fuego al mínimo y cocina por 4 minutos más
4. Pasado el tiempo de cocción agrega las papas y deja cocinar hasta que estén cocidas
5. Apaga la cocina y sirve con arroz blanco



- No consigo diferenciar
- Humano
- Computador

Con base en tu respuesta anterior ¿Probarías hacer esta receta? *

- Si, la receta es corta, la receta parece deliciosa
- No, la receta es corta, la receta no parece deliciosa
- Si, la receta es extensa, la receta parece deliciosa
- No, la receta es extensa, la receta no parece deliciosa

Voltar

Próxima

Limpar formulário

Receta 8

Receta 8 ¿Esta receta fue creada por una máquina o una persona? *



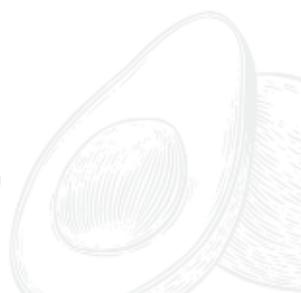
receta

INGREDIENTES

- 2 cebollas medianas
- 1 tomate
- 1 aji limo
- 1 puñado de cilantro
- 1 limón
- 2 cucharadas de aceite de oliva
- pimienta
- sal

PREPARACIÓN

1. Cortamos los bordes de las cebollas, partimos en dos y pelamos
2. Luego cortar otra vez por el medio a las cebollas y quitar las capas internas para poder realizar el corte al estilo juliana y poner en un tazón
3. Ahora picar un puñado de hojas de cilantro y reservar
4. Quitar el cabo del aji y cortar por la mitad y retirar las pepas y poner ambas mitades encima de la otra para picarlas
5. En un recipiente colocar las cebollas previamente cortadas, el cilantro picado, el aji picado, pimienta y sal al gusto y remover bien
6. Agregar el jugo de un limón, dejar reposar por 30 minutos y agregar 2 cucharadas de aceite de oliva



- Humano
- Computador
- No consigo diferenciar

Con base en tu respuesta anterior ¿Probarías hacer esta receta? *

- Si, la receta es corta, la receta parece deliciosa
- No, la receta es corta, la receta no parece deliciosa
- Si, la receta es extensa, la receta parece deliciosa
- No, la receta es extensa, la receta no parece deliciosa

[Voltar](#)

[Próxima](#)

[Limpar formulário](#)

Receta 9

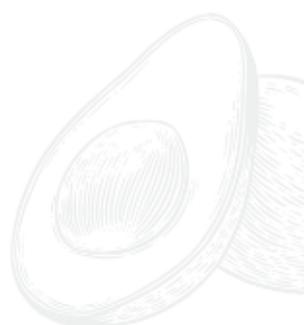
Receta 9 ¿Esta receta fue creada por una máquina o una persona? *



receta

INGREDIENTES

- 1 kilo de carne de res
- 1 cucharada de ajo molido
- 2 cucharadas de aji panca
- sal
- pimienta
- comino
- orégano
- aceite vegetal



PREPARACIÓN

1. cortar la carne en trozos medianos.
2. agregar la cebolla, el ajo, la sal, pimienta, sal y pimienta al gusto
3. mezclar bien y reservar.
4. retirar del fuego y dejar cocinar por unos minutos
5. en una olla, agregar el jugo de limón y mezclar bien,
6. incorporar los ajies y el comino en una sartén con abundante aceite, dejar cocinar a fuego medio hasta que se doren
7. añadir el culantro picado y la salsa de tomate.
8. cocinar por un par de minutos más, cuando el caldo esté listo, añadir el tomate y las verduras
9. servir acompañado de un chorro de aceite caliente
10. colocar en un plato y servir



- No consigo diferenciar
- Humano
- Computador

Con base en tu respuesta anterior ¿Probarías hacer esta receta? *

- Sí, la receta es corta, la receta parece deliciosa
- No, la receta es corta, la receta no parece deliciosa
- Sí, la receta es extensa, la receta parece deliciosa
- No, la receta es extensa, la receta no parece deliciosa

Voltar

Próxima

Limpar formulário

Receta 10

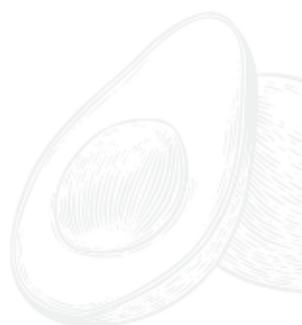
Receta 10 ¿Esta receta fue creada por una máquina o una persona? *



receta

INGREDIENTES

- 1 k de lomo de res
- 1 cebolla
- 1 pimienta
- 2 dientes de ajo
- 1 pizca de orégano
- 3 semillas de malaqueta
- 1 chorro de salsa inglesa
- 1 chorro de aceite para sofreir
- 1 pizca de sal

**PREPARACIÓN**

1. Cortar la cebolla en trozos grandes reserva
2. En una sartén mediana a fuego medio agrega el ajo, la cebolla y el orégano, sofríe por unos minutos hasta que la cebolla esté transparente
3. Agrega los ajíes y cocina por 10 minutos, incorpora el pimienta y cocina unos minutos a fuego medio revolviendo todo
4. Incorpora el lomo saltado el orégano y mezcla, cocinar el lomo saltado por unos minutos hasta que la carne esté tierna
5. Sirve con arroz blanco graneado y hojas de lechuga



- Computador
- Humano
- No consigo diferenciar

Con base en tu respuesta anterior ¿Probarías hacer esta receta? *

- Si, la receta es corta, la receta parece deliciosa
- No, la receta es corta, la receta no parece deliciosa
- Si, la receta es extensa, la receta parece deliciosa
- No, la receta es extensa, la receta no parece deliciosa

Voltar

Enviar

Limpar formulário