Avances de ingeniería del lenguaje, del conocimiento y la interacción humano máquina

Volumen II

Juan Manuel González Calleros
Josefina Guerrero García
Claudia Zepeda Cortés
Darnes Vilariño Ayala
José Luis Carballido Carranza
Maria Josefa Somodevilla Garcia (Editores)

Avances de ingeniería del lenguaje, del conocimiento y la interacción humano máquina

Volumen II

Juan Manuel González Calleros
Josefina Guerrero García
Claudia Zepeda Cortés
Darnes Vilariño Ayala
José Luis Carballido Carranza
María Josefa Somodevilla García
Coordinadores

Esta obra fue dictaminada por pares académicos externos a la institución de adscripción de los autores.

United Academic Journals (UA Journals)

ISBN 978-84-125164-3-2
Digital (suministrado electrónicamente)
Detalle Formato: PDF
Núm. páginas: 110
Español / Castellano 30/10/2022
Huelva
España
Edición digital con tiraje de un ejemplar

Proceso Editorial

La selección de contribuciones en esta obra pasó por un proceso de revisión doble ciego.

Los jueces que fungieron en este proceso están enlistados en la sección de revisores.

La tasa de aceptación de trabajos es del 68% con un total de 13 trabajos publicados. Apoyados por la plataforma web easychair el proceso doble ciego. Después del proceso de selección, se evaluaron nuevamente los trabajos para certificar que cumplían con las correcciones derivadas de las observaciones de la primera ronda de evaluaciones.

El cuidado editorial de esta obra es responsabilidad de los integrantes del cuerpo académico en sistemas y ambientes educativos: Dr. Juan Manuel González Calleros, Dra. Josefina Guerrero García, el cuerpo académico de ingeniería del lenguaje y comunicación representado por la Dra. Darnes Vilariño Ayala y la María Josefa Somodevilla García, y a la Dra. Claudia Zepeda Cortés y el Dr. José Luis Carballido Carranza del cuerpo académico Aplicaciones Tecnológicas para la Enseñanza Aprendizaje.

Avances de ingeniería del lenguaje, del conocimiento y la interacción humano máquina

Volumen II

Revisores

- Amparo Dora Palomino Merino
- Beatriz Beltrán Martínez
- Carmen Cerón Garnica
- Claudia Zepeda Cortés
- Darnes Vilariño Ayala
- Georgina Flores Becerra
- Guillermo De Ita Luna
- Helena Gómez Adorno
- Hilda Castillo Zacatelco
- Iván Olmos Pineda
- Ivo Pineda Torres
- José Andrés Vázquez Flores
- José Arturo Olvera López
- José de Jesús Lavalle Martínez
- José Luis Carballido Carranza
- José Raymundo Marcial Romero
- Josefa Somodevilla García
- Josefina Guerrero García
- Juan Manuel González Calleros
- Luis Enrique Colmenares Guillén
- Manuel I. Martin Ortiz
- María Aurora Diozcora Vargas Treviño
- María de la Concepción Pérez de Celis
- María Auxilio Medina Nieto
- Mario Rossainz López
- Meliza Contreras González
- Mireya Tovar Vidal
- Omar Flores Sánchez
- Rafael de la Rosa Flores
- Reyna Carolina Medina Ramírez
- Sergio Vergara Limón

PRÓLOGO

El uso y adopción en la vida diaria de la inteligencia artificial (IA) es cada vez mayor. Muchos problemas se están resolviendo apoyados de sistemas inteligentes. No obstante, muchas soluciones sin gran innovación en los algoritmos o técnicas de IA, pero con gran impacto en contextos específicos no siempre son accesibles a comunidades académicas. En este libro precisamente nos ocupamos de atender esta realidad y nos proponemos generar la primera de muchas ediciones de divulgación y acceso universal al conocimiento.

En avances de ingeniería del lenguaje, del conocimiento y la interacción humano máquina, presentamos métodos, técnicas, metodologías de aplicación de la IA a problemas en el contexto nacional e internacional. El primer conjunto de trabajos se concentra en la visión por computadora aplicada a la resolución de problemas específicos.

En el Capítulo 1, se presenta el trabajo de Topología de Redes Dinámicas Bayesianas para la Estimación de Desplazamientos de Objetos en Ambientes Vehiculares. Presenta un método basado en Redes Dinámicas Bayesianas (RDB) para inferir las trayectorias de los objetos.

En el Capítulo 2, se presenta una RNC que identifica los parámetros dinámicos de la articulación horizontal de un robot cartesiano. La RNC extrae los residuos paramétricos a partir de una imagen creada con las señales del robot para reconstruir el par con el modelo dinámico.

En el Capítulo 3, se realiza el reconocimiento automático de lengua de señas empleando una red neuronal BiLSTM a través de una metodología que aborda el reconocimiento de lengua de señas ocupando características basadas en componentes manuales y no manuales.

En el Capítulo 4, se propone un método de preprocesamiento orientado a garantizar la homogeneización del tamaño de imágenes médicas, para el desarrollo y posterior evaluación de sistemas automáticos de detección de enfermedades, determinando la región de interés experimental y realzando las estructuras vasculares aplicando una Ecualización Adaptativa de Histograma Limitado por Contraste (EAHLC) para facilitar las etapas posteriores de diagnóstico y clasificación.

En el Capítulo 5, se presenta una estrategia para la autenticación de personas, analizando cada una de las etapas que conforman este proceso: detección de rostros, preprocesamiento, extracción de rasgos faciales y etapa de clasificación.

En el Capítulo 6, se generó un modelo de recomendación basado en recuperación de información para textos turísticos mexicanos en español para la tarea de sistema de recomendación del Rest-Mex 2022.

En el Capítulo 7, se hace una revisión de las aplicaciones del reinforcement learning y sus variantes en el ámbito del cuidado de la salud.

En el Capítulo 8, se presenta una breve descripción de los métodos utilizados para obtener la localización, así como sus ventajas y desventajas; se comparan y analizan 5 arquitecturas presentando sus resultados para el desarrollo de una tarea de localización.

En el Capítulo 9, se presenta un estudio sobre el problema de ambigüedad referencial en pronombres presente en el lenguaje natural. Se describen algunos de los marcos de referencia conocidos como desafío de los esquemas de Winograd (WSC); se describen otras variantes de conjuntos de datos como: DPR, PDP, WNLI, WinoGender, WinoBias, WinoFlexi, WinoGrande. Finalmente, se plantea una conclusión acerca de la importancia que tiene representar conocimiento y razonamiento de sentido común para la comprensión del lenguaje y su repercusión en los sistemas de Inteligencia Artificial de hoy en día.

En el Capítulo 10, se presenta una metodología que permita combinar las estrategias del Procesamiento del Lenguaje Natural con la capacidad de selección de un algoritmo genético para crear un generador de frases automático. Partiendo de textos en el idioma español.

La diversidad de temas de esta obra sin duda ofrecerá al lector disponer de un panorama amplio sobre los avances de ingeniería del lenguaje, del conocimiento y la interacción humano máquina. Además, de sensibilizarlo sobre lo que en contextos como el mexicano es necesario abordar. Esperamos que disfruten de la lectura.

Claudia Blanca González Calleros
Josefina Guerrero García
Claudia Zepeda Cortés
Darnes Vilariño Ayala
José Luis Carballido Carranza
Maria Josefa Somodevilla Garcia

Contenido

Capítulo 1. Topología de Redes Dinámicas Bayesianas para la Estimación de Desplazamientos de Objetos en Ambientes Vehiculares

Lauro Reyes-Cocoletzi[1], Iván Olmos-Pineda[1], J. Arturo Olvera-López[1]
[1]Benemérita Universidad Autónoma de Puebla, Facultad de Ciencias de la Computación,
Av. San Claudio y 18 Sur, Ciudad Universitaria, Puebla, México
e-mail autor por correspondencia. lauro.reyesc@alumno.buap.mx

Resumen. Un punto fundamental para lograr el desarrollo de vehículos terrestres autónomos es el resolver la problemática de evasión de objetos de la misma forma que lo logra realizar un conductor humano. Predecir trayectorias de múltiples objetos en ambientes dinámicos de tráfico es un reto para garantizar que un vehículo autónomo se desplace sin riesgo de colisión. En este trabajo se presenta un método basado en Redes Dinámicas Bayesianas (RDB) para inferir las trayectorias de los objetos. La información del entorno se obtiene a través de vídeo estereoscópico, se calculan los vectores de dirección de múltiples objetos y se destacan las trayectorias con mayor probabilidad de ocurrencia. El enfoque propuesto se evaluó utilizando entornos de prueba que consideraban diferentes ambientes de carreteras y múltiples objetos en escenarios de tráfico del mundo real. Se realiza una comparación de los resultados obtenidos con respecto al ground truth (gt) de las trayectorias seguidas por cada objeto detectado. Los resultados obtenidos permiten realizar una comparación cuantitativa de los trayectos estimados contra el trayecto real observado en video. De acuerdo a los resultados experimentales, el método propuesto obtiene una tasa de predicción de al menos 75% para el cambio de dirección teniendo en cuenta el riesgo de colisión.

Palabras Clave: Probabilidad de Colisión, vectores de Dirección, Redes Dinámicas Bayesianas.

1 Introducción

La conducción autónoma para su aplicación en vehículos terrestres es un campo de investigación que ha retomado interés en los últimos años, sin embargo, aún hay áreas de oportunidad en los métodos implementados para llevar a cabo la estimación desplazamiento en entornos de tráfico real. De forma concreta se requiere mejorar la detección y estimación de movimiento de los objetos encontrados en el recorrido para evitar colisiones.

El desempeño de los métodos propuestos en la literatura depende de la capacidad de procesar la información ambiental para establecer las condiciones de movimiento y calcular los posibles cambios de dirección de los objetos detectados para evitar el riesgo

de colisión (Prabhakar et al, 2017). El enfoque al plantear las características y análisis del entorno vehicular es de suma importancia porque de acuerdo a cómo se aborde el problema la posible solución tendrá relevancia en situaciones particulares o generales.

Por el momento, los modelos más avanzados se basan en el cálculo probabilístico en función de las condiciones cinemáticas y/o dinámicas de los objetos detectados en la carretera, esto permite manejar los problemas de incertidumbre y ambigüedad.

El método propuesto que se aborda en este trabajo de investigación corresponde a un modelo basado en Redes Dinámicas Bayesianas que obtiene un valor de probabilidad $\geq$ 75 % con respecto a la detección y estimación de posición espacial para poder determinar cuantitativamente los cambios del vector de dirección de los objetos en el transcurso de los frames de interés.

El contenido de este trabajo se divide de la siguiente manera: la sección 2tratalos trabajos relevantes relacionados con la problemática a resolver, en la sección 3 se presentan los detalles del método planteado y los parámetros considerados para obtener la solución requerida, en la sección 4 se muestran resultados obtenidos dados los experimentos realizados y en la sección 5 se presentan las conclusiones con respecto al desempeño del método propuesto.

2 Trabajos relacionados

El problema de la estimación de la trayectoria en entornos vehiculares se ha abordado ampliamente en la literatura, por ejemplo, Duan et al. (2020) presentan un método de aprendizaje por refuerzo jerárquico para la toma de decisiones en vehículos autónomos de tal forma que no dependa de una gran cantidad de datos etiquetados para llevar a cabo la conducción. Este enfoque modela y aprende el cambio de dirección como un proceso de decisión Markoviano, por lo que, en cada intervalo de tiempo de interés, el vehículo observa un estado, realiza una acción, recibe una señal escalar y finalmente alcanza el estado siguiente.

Sin embargo, este modelo presenta variación debido al proceso de arranque aleatorio y al número de veces que el vehículo autónomo llega al destino en cada época de acuerdo al modelo de aprendizaje planteado. En comparación, la velocidad de aprendizaje y el rendimiento tiene rango de mejora según el autor.

En el artículo publicado por Zhag et al. (2018), se desarrolla un esquema de normalización de características y se establece una estrategia para construir modelos de regresión de procesos gaussianos tridimensionales a partir de patrones de trayectoria bidimensionales para capturar las características espacio-temporales de situaciones de tráfico. Sin embargo, dado que el entorno del tráfico es un sistema dinámico e incierto, la acción posterior obtenida por el modelo no es óptima al ejecutar una secuencia de decisiones, ya que la velocidad en este proceso se considera invariante, lo que no ocurre en situaciones en entornos reales.

Por otro lado, Schulz et al. (2019) modelan el proceso en una RDB que permite la especificación de las relaciones entre los objetos, así como las dependencias causales y temporales para manejar la incertidumbre de las mediciones. El proceso planteado de tomade decisiones para cada obstáculo visible se compone de tres capas jerárquicas: intención de ruta, intención de maniobra y acción continua. La estructura de la red se

adapta en tiempo de ejecución, creando y eliminando hipótesis de ruta - maniobra, sin embargo, este método solo se ha probado en 3 escenarios de tráfico diferentes.

Como se mencionó en los trabajos anteriores, la problemática por resolver consiste en determinar el conjunto de posibles rutas y maniobras dada la información percibida del ambiente y en algunos casos la información proporcionada a través de un mapa topológico. De manera que la acción continua de cada objeto detectado se especifica mediante modelos de comportamiento dependientes del contexto, sus intenciones de ruta y maniobra (Hou, Chen, y Chen, 2019; Sun et al., 2019; Mo, Xing y Lv,2020; Xie et al., 2018).

De los trabajos mencionados, se plantea que un enfoque con alto rendimiento en la detección de objetos y la estimación de sus trayectorias a largo plazo necesita conjuntar múltiples técnicas para obtener un método robusto.

La propuesta de este trabajo de investigación realiza aportaciones en cuanto a la implementación de detección de múltiples objetos y la predicción de su cambio de dirección con base en técnicas de aprendizaje profundo y modelos de probabilidad (RDB) mediante el enfoque de visión computacional. La siguiente sección describe a detalle el método implementado.

3 Descripción del método propuesto

El enfoque propuesto estima la trayectoria que debe seguir un objeto detectado, el flujo y el cálculo de la información se realiza a través de 4 módulos. En primer lugar, se captura la información de la carretera con un par de cámaras de vídeo para emular la visión estereoscópica y determinar las regiones de interés (ROI).

La detección de las ROI consta de dos módulos que cooperan entre sí: una red neuronal convolucional multicapa (módulo RNC) que señala los objetos detectados en el frame en un cuadro delimitador y el módulo que procesa mapas de disparidad (MD) con respecto a las imágenes izquierda y derecha de la escena para estimar la distancia aproximada de los objetos con respecto al ego-vehículo (vehículo que captura la información del ambiente).

Estas ROI son rastreadas de acuerdo a los datos consecutivos que se adquieren en el tiempo (módulo de seguimiento), lo que permite obtener características físicas sobre el movimiento de los objetos e información relacionada. Por último, la información obtenida, de los módulos previos, se proporciona a un modelo probabilístico RDB que, dado un conjunto de datos de objetos detectados y observados, estima las posibles trayectorias de cada uno de estos y su correspondiente porcentaje de riesgo de colisión.

La Figura 1 muestra los módulos correspondientes, antes mencionados, con respecto al método propuesto.

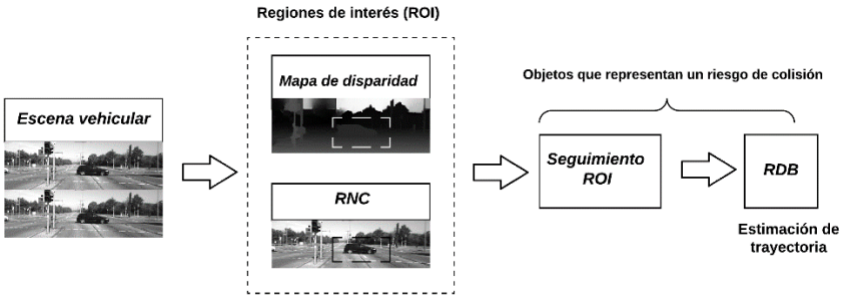


Fig. 1. Módulos para la detección y estimación de trayectorias en ambientes vehiculares.

3.1 Seguimiento de los objetos

El enfoque general de rastreo en video se centra en el seguimiento de múltiples objetos (Multiple Object Tracking, MOT) en dos dimensiones, este enfoque se basa en una sucesión de pasos de detección. Las detecciones consecutivas que se clasifican de forma similar se unen para determinar las trayectorias (Cao et al., 2020).

Los modelos de rastreo se utilizan para predecir las ubicaciones de los objetos de interés de forma independiente y estimar el acierto en la detección por medio de puntuaciones. El problema del seguimiento puede formularse como un problema de optimización, donde en un frame de interés, el conjunto de objetos N^t con localización espacial $\hat{X}^t = \{\hat{x}_i^t\}_{i=1}^{N^t}$ en una imagen actual I^t son elegidos de M^t candidatos del conjunto global $O^t = \{\hat{x}_j^t\}_{i=1}^{M^t}$para maximizar el resultadocomo en las ecuaciones (1) y (2) (Cao et al., 2020):

$$\hat{X}^t = argmaxf(I^t, X^t; a^t, W^{t-1}) \quad \textbf{(1)}$$

$$\sum_i \quad a_{ij}^t \leq 1, a_{ij}^t \in \{0,1\} \quad \textbf{(2)}$$

El parámetro $a^t = \{a^t{}_{ij} \in \{0,1\}\}$ indica la asociación entre el *i–ésim*o objeto rastreado en$\hat{X}^{t-1}$en el frame *t−1* y *la j−ésima* localización en $\hat{X}^t$en el frame en el tiempo *t* como en la siguiente función:

$$a_{i,j}^t = \{1,\ si\ x^{t-1}\ es\ asociado\ con\ x_j^t\ 0,\ en\ otro\ caso \quad \textbf{(3)}$$

cada candidato solo puede asignarse como máximo a un objetivo rastreado, donde$W^t = \{w_i^t\}_{i=1}^{N^t}$es el conjunto de parámetros para modelar cada objeto, que generalmentese aprende a través de un procedimiento de entrenamiento utilizando la información deapariencia o ubicación del objetivo (Cao et al., 2020).

Los parámetros del modelo deben determinarse mediante imágenes anteriores y ubicaciones destino hasta el frame*t−1*, resolver el problema de MOT implica encontrar a^t para cada frame del intervalo de interés.

3.2 Topología propuesta Red dinámica Bayesiana

La topología RDB planteada incorpora análisis de los *estados-posición* de los objetos para rastrear y estimar las trayectorias de estos en la escena vehicular para evitar colisiones. El planteamiento general dela propuesta metodológica implica definir la escena de tráfico incluyendo a varios tipos de objetos participantes con diferentes características de movilidad en el tiempo: velocidad de desplazamiento(v_t^j), distancia con respecto a la referencia (d_t^j), posición espacial en la escena(p_t^j) y ángulo de orientación (ψ_t^j). Se definen las variables en el espacio de estado discreto para facilitar el análisis del problema.

Existen dos tareas principales en este problema, la primera es calcular y proponer la trayectoria (Tr_t^j) de diferentes objetos, así como justificar si las estimaciones corresponden a lo que se observa en el mundo real (c_t^j), la segunda tarea es ver si el modelo puede predecir las inferencias de movimiento y los cambios de trayectoria $P(Tr_t^j | c_t^j)$. El modelo de red Bayesiana se desarrolla durante dos segmentos de tiempo en dependencias condicionales del estado latente (*t-1, t*).

La Figura 2 muestra la topología de la RBD, las líneas continuas y las líneas punteadas son dependencias de observaciones causales y temporales respectivamente.

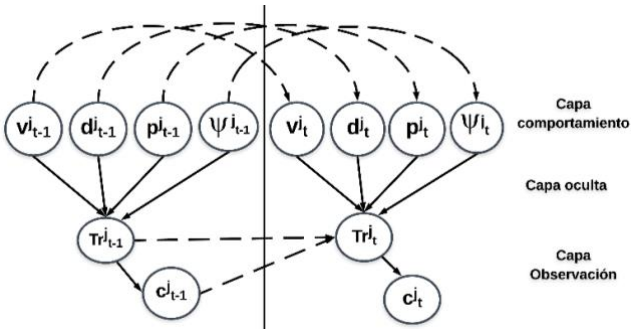


Fig. 2. Topología de la Red Dinámica Bayesiana implementada.

El agrupamiento de las variables a procesar de los objetos de interés detectados se realiza con respecto a los datos correspondientes de los frames analizados. De tal forma que para una escena con más de un objeto detectado se puede definir los grupos de objetos en función de las variables de estado discreto es decir: velocidades estimadas $V_t=(v^1_t, v^2_{t,...}, v^j_t)$, posiciones espaciales aproximadas $P_t=(p^1_t, p^2_{t,...}, p^j_t)$, distancias en profundidad $D_t=(d^1_t, d^2_{t,...}, d^j_t)$, ángulos de dirección $\Gamma_t=(\psi^1_t,\psi^2_{t,...},\psi^j_t)$ respecto al vector de desplazamiento. Los parámetros de salida por consiguiente también se expresan de acuerdo al cálculo del desplazamiento del grupo de objetos en la escena $TR_t=(Tr^1_t, Tr^2_{t,...}, Tr^j_t)$ y la probabilidad de colisión asociada $C_t=(c^1_t, c^2_{t,...}, c^j_t)$.

De forma global se agrupan en la notación $Z_t=[V_t, P_t, D_t,\Gamma_t]$ a los objetos que presentan riesgo de colisión de acuerdo a la dinámica de desplazamiento detectado en el transcurso de los frames de interés. Entonces, para los cambios de desplazamiento detectados se predice según la probabilidad de transición para el estado Z_{t+1}dado como una aproximación de $P(Z_t | Z_{t+1})$.

El modelo de interacción propuesto representa el comportamiento de interacción de múltiples participantes en el tráfico con una categoría arbitraria. Una primera inferencia para el desarrollo del algoritmo propuesto para la estimación de la trayectoria se basa en vectores de estado latentes globales (Z_t), es decir, estados previos específicos de los participantes del tráfico.

La topología propuesta de la RDB propaga las relaciones de dependencia condicional entre las variables de interés y estima su efecto en el intervalo de interés a analizar. Por lo tanto, dado el intervalo de frames $t = 1,2,3,...,T$, con respecto a la topología propuesta y las variables de la capa de comportamiento, la distribución de probabilidad conjunta se expresa por medio dela ecuación 4.

$$P(Z_{t+1}) = \prod_{t=1}^{T} \quad Z_{t-1} \times \prod_{t=1}^{T} \quad P(Z_t | \, P(Z_{t+1})) \qquad \textbf{(4)}$$

Una vez determinado el estado de las variables de interés y la estructura de la red a continuación se estima la distribución de probabilidad condicional entre los nodos principales y secundarios. Los resultados de la aproximación del comportamiento se obtienen en función de la información previamente observada, que puede expresarse dada la relación:

$$Z_t^* = max_{Z_t} P(Z_t | C_{1:t}) \qquad \textbf{(5)}$$

donde Z_t es el comportamiento de los objetos y la dinámica en el momento de la detección t y Z^*_t representa la inferencia resultante más probable basada en la información previamente obtenida (Xie et al., 2018).

Las variables de la estructura de la red implican la estimación de los parámetros en relación a las distribuciones de probabilidad condicional basadas en la estructura de la RDB. En la topología propuesta en este trabajo se definen los parámetros de la capa oculta $H = (TR_t)$, así como los parámetros de la capa observable $Op = (Z_t, C_t)$.

Para la representación compacta de las distribuciones de probabilidad los parámetros correspondientes a la capa oculta se representan en función de la variable h, las relaciones correspondientes a los nodos de la capa observable corresponden a o_{nodos}, por lo tanto, en el caso parcialmente observable, la probabilidad logarítmica se define por la ecuación (6).

$$L = \sum_{nodos} \quad log \left(\sum_h \quad P(H = h, Op = o_{nodos}) \right) \qquad \textbf{(6)}$$

Para optimizar el planteamiento de la ecuación (6), se utiliza la teoría de estimación ML (Maximum Likelihood) para ajustar las relaciones causales y estimar los parámetros posibles más cercanos al valor del riesgo de colisión en función de las distribuciones de probabilidad condicional con respecto a los parámetros de las variables observables de múltiples objetos en la escena. Se utiliza en conjunto el algoritmo de EM (ExpectedMaximization) y la desigualdad de Jensen (Xie et al., 2018) para reescribir el planteamiento de los estados observables en la ecuación (7).

$$L = \sum_{nodos} \quad \sum_h \quad f(h|o_{nodos}) \; log \; log \; (P(H, O)) \; - \sum_{nodos} \quad \sum_h \quad f(h|o_{nodos}) \; log \; log \; (f(h \, |o_{nodos})) \qquad \textbf{(7)}$$

La función de verosimilitud ajusta las condiciones del modelo al estimar los parámetros óptimos dado el resultado de salida correspondiente, este es cercano al valor verdadero del parámetro que cumple con las condiciones de la mencionada desigualdad de Jensen $0 \leq f(h|o_{nodos}) \leq 1$.

4 Experimentos

En esta sección se muestran los experimentos realizados para la inferencia de trayectoria de los objetos, se señalan los resultados cuantitativos y características cualitativas del procesamiento de las escenas vehiculares.

Para la ejecución de las pruebas del algoritmo propuesto y el posterior análisis de resultados obtenidos se utilizaron tres conjuntos de datos: kitti Vision Benchmark, capturas propias y videos de simulador (city car driving).

La cantidad de información procesada corresponde a aproximadamente 650,920frames en entornos reales más 54,000 frames en el entorno simulado.

La información procesada referente a los parámetros de observación durante el recorrido se representa en un esquema espacial, de igual forma se denotan los cambios de dirección vectorial durante el trayecto (Figura 3) así como los frames de interés de la escena analizada. Los cambios de dirección estimados corresponden a los vectores dirección frente, derecha e izquierda, según su magnitud es su preponderancia en el posible cambio de dirección calculado.

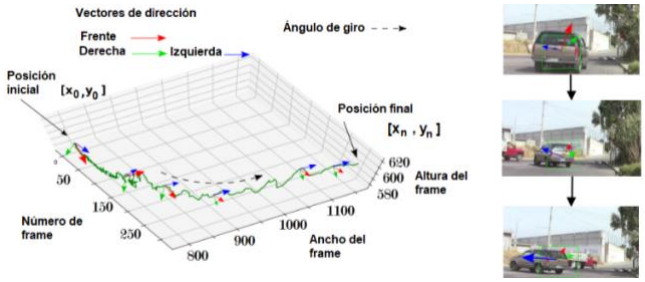


Fig. 3. Representación del desplazamiento de un objeto de interés.

Una vez establecidos los parámetros, variables y el método de estimación se procede a realizar experimentos y mostrar los resultados obtenidos.

De forma específica, por ejemplo, después de llevar a cabo la adquisición y procesamiento de la información de una escena vehicular bajo condiciones generales de desplazamiento, en la Figura 4 se muestra algunos de los frames analizados para dar seguimiento al vehículo detectado (remarcado por un cuadro envolvente).

Fig. 4. Escena vehicular y detección del objeto de interés.

Es posible calcular el cambio de dirección y representarlo de tal forma que se muestre la relación de probabilidad normalizada del cambio de dirección asociada al riesgo de colisión, en la gráfica de la Figura 5 se muestra los resultados del procesamiento realizado en la RBD para el vehículo detectado de la Figura 4, por lo que aquel evento que tiene mayor probabilidad de ocurrir es: no colisionar dado que el objeto detectado mantendrá el rumbo frontal ya que tiene el valor mayor en la gráfica.

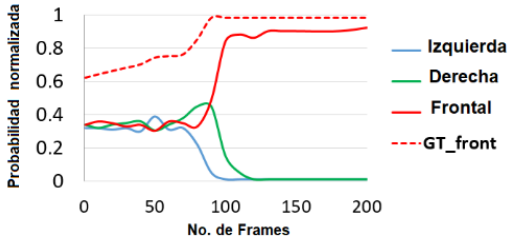


Fig. 5. Descripción del posible cambio de dirección, con respecto a la probabilidad calculada, de un objeto de interés detectado comparado con el ground truth (GT).

4.1 Evaluación de los resultados

El marco de referencia en los trabajos relacionados establece valores objetivo de MOTA (Multiple Object Tracking Accuracy) entre75.70% y 88.88% en la tasa de rendimiento, para MOTP (Multiple Object Tracking Presicion) el valor deseado se encuentra entre 73.42% y 85.64 %. La Tabla 1muestrala comparativa de los parámetros obtenidos de distintos trabajos relacionados contra los resultados obtenidos por la propuesta desarrollada.

Tabla 1. Comparativas métricas de evaluación trabajos relacionados vs la propuesta implementada.

Evaluación de seguimiento (Base de datos kittiVision)			
Modelo	Autores	MOTA(%)	MOTP(%)
AB3DMOT	Weng et al., 2020	80.39	81.26
3DT	Hou et al.,2019	82.25	80.52
mmMOT	Zhang et al.,2018	83.84	85.24
MTPCPCG	Hu et al.,2019	88.88	85.64
Propuesta	Reyes et al	81.80	85.50

Cabe mencionar que la propuesta obtuvo para la métrica MOTA valores en el intervalo entre 81.8% a 82.3%con una desviación estándar de alrededor de $\mu = 3$, para MOTP el rango se encuentra entre84.9% y 85.5% con $\mu = 3$ aproximadamente, lo cual muestra una consistencia en el funcionamiento del seguimiento de los objetos, además la propuesta está por debajo del mejor resultado (MTPCPCG) a 7.08 % en MOTA y de 0.14% en MOTP.

En lo referente a la posición espacial estimada y el cálculo de la probabilidad de cambio de trayectoria de los objetos detectados se realiza una comparación con dos

trabajos relacionados que al igual que la propuesta toman como referencia el GT y la tasa de error calculada, es decir, la posición y el trayecto real que tomaron los objetos detectados en video. Los métodos a tomar en consideración para la comparativa son: el modelo RBDHM (Schulz et al., 2019) y con el modelo ERI (Sun et al., 2019). Para la evaluación de las métricas de estimación de trayectorias de objetos se toman en consideración los resultados obtenidos al procesar secuencias de videos de tres bases de datos.

La detección de la información corresponde a un intervalo de 5 frames, es decir, el vector de dirección se recalcula cada 0.166 segundos, por lo que las secuencias corresponden al análisis del error con respecto al desplazamiento a través de 650 a 1250 frames procesados aproximadamente por escena, con velocidades de desplazamiento de los objetos en escena no mayor a 60 km/hr a distancias en el rango de hasta 45 metros. La métrica utilizada para la evaluación del vector cambio de dirección corresponde a la tasa de error RMSE (Root Mean Square Error) dado por:

$$RMSE = \sqrt{\frac{1}{fr}\sum_{i_t=0}^{fr} \|Z_{p(fr)} - Z_{GT(fr)}\|^2} \quad \textbf{(8)}$$

donde se representa la estimación del movimiento (Z_p) y la trayectoria real ($Z_{GT(fr)}$) en el intervalo de los frames de interés (fr) respecto al objeto en escena.

Dados los resultados obtenidos en la Tabla 2 el desempeño de la propuesta es prácticamente igual comparado con el mejor resultado obtenido por RBDHM pero a diferencia de este modelo, la propuesta no requiere la información de un sensor tipo lidar para la percepción del ambiente. De aquí la razón por la cual el modelo RBDHM no se compara contra la propuesta con la base de datos propia ni con la del simulador, pues estas bases de datos no cuentan con información de lidar.

Con respecto a la comparación con el modelo ERI, la propuesta tiene mejor desempeño del parámetro RMSE al procesar la información de las 3 bases de datos.

Tabla 2. Comparativa estimación de dirección (tasa de error).

Evaluación de estimación de dirección RMSE (probabilidad normalizada)			
Base de datos	Modelo	RMSE	μ
kittiVision	RBDHM	0.176	0.073
	ERI	0.180	0.049
	Propuesta	0.177	0.052
Simulador (City car driving)	ERI	0.155	0.039
	Propuesta	0.161	0.054
Videos capturados propios	ERI	0.181	0.051
	Propuesta	0.169	0.056

5 Conclusiones

En este trabajo se presenta una topología de RDB para inferir las probabilidades de cambio de ruta con respecto a la información obtenida al modelar las características espacio-temporales del movimiento de los objetos detectados

La evaluación de la propuesta de forma cuantitativa para el parámetro MOTA está por debajo del mejor rendimiento (7%), pero tiene menor variabilidad ya que la desviación estándar es menor ($\mu_{RBDHM} > \mu_{propuesta}$) lo que implica menor oscilación entre la estimación de probabilidad del vector de dirección en intervalos de muestreo adyacentes. La tasa de error RMSE de la propuesta (0.177) comparado con el mejor rendimiento (RBDHM = 0.176) difieren por 0.001 unidades lo cual no representa una diferencia significativa, por lo que el desempeño es similar, sin embargo, la variabilidad del modelo RBDHM es mayor ya que $\mu_{propuesta} < \mu_{RBDHM}$.

Finalmente, como trabajo a futuro se plantea incrementar las variables percibidas del ambiente para explorar mayor número de relaciones causales y hacer más robusta la topología RDB para observar el impacto en el rendimiento de las estimaciones realizadas.

6 Agradecimientos

Este trabajo fue realizado con el apoyo de la beca doctoral CONACYT No. 708553.

Referencias

1. Cao, J., Song, C., Peng, S., Song, S., Zhang, X., & Xiao, F. (2020). *Trajectory tracking control algorithm for autonomous vehicle considering cornering characteristics*. IEEE Access, 8, 59470–59484. https://doi.org/10.1109/ACCESS.2020.2982963.
2. Duan, J., Li, S. E., Guan, Y., Sun, Q., & Cheng, B. (2020). *Hierarchical reinforcement learning for self-driving decision-making without reliance on labelled driving data*. IET Intelligent Transport Systems, 14(5), 297–305.
3. Hou, G., Chen, S., &Chen, F. (2019). *Framework of simulation-based vehicle safety performanceassessment of highway systemunder hazardous driving conditions*. Transportation Research Part C: Emerging Technologies, 105, 23–36.https://doi.org/10.1016/j.trc.2019.05.035.
4. Hu, H. N., Cai, Q. Z.,Wang, D., Lin, J., Sun,M., Krahenbuhl, P., & Yu, F. (2019). *Joint monocular 3d vehicle detection and tracking*. Proceedings of the IEEE/CVF International Conference on Computer vision, 5390–5399.
5. Mo, X., Xing, Y., & Lv, C. (2020). *Interaction-aware trajectory prediction of connected vehicles using cnn-lstm networks*. IECON 2020 the 46th annual conference of the IEEE industrial electronics society, 5057–5062.https://doi.org/10.1109/IECON43393.2020.9255162.
6. Prabhakar, G., Kailath, B., Natarajan, S., & Kumar, R. (2017). *Obstacle detection and classification using deep learning for tracking in high-speed autonomous driving*. IEEE

region 10 symposium(tensymp), 1–https://doi.org/10.1109/TENCONSpring.2017.8069972.
7. Schulz, J., Hubmann, C., Lochner, J., & Burschka, D. (2019). *Multiple model unscented kalman filtering in dynamic bayesian networks for intention estimation and trajectory prediction*. 21st International Conference on Intelligent Transportation Systems (ITSC), 1467-1474.https://doi.org/10.1109/ITSC.2018.8569932.
8. Sun, L., Zhan, W., Wang, D., & Tomizuka, M. (2019). *Interactive prediction for multiple, heterogeneous traffic participants with multi-agent hybrid dynamic bayesian network*. 2019 IEEE Intelligent Transportation Systems Conference (ITSC), (pp. 1025-1031).https://doi.org/10.1109/ITSC.2019.8917031.
9. Weng, X.,Wang, J., Held, D., & Kitani, K. (2020). *3d Multi-object tracking: A baseline and new evaluation metrics*. 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems(iros), 10359–10366. https://doi.org/10.1109/IROS45743.2020.9341164.
10. Xie, G., Gao, H., Huang, B., Qian, L., &Wang, J. (2018). *A driving behavior awareness model based on a dynamic bayesian network and distributed genetic algorithm*. International Journal of Computational Intelligence Systems, 11(1), 469–482.
11. Zhang, Y., Qiu, Z., Yao, T., Liu, D., &Mei, T. (2018). *Fully convolutional adaptation networks for semantic segmentation*. Proceedings of the IEEE conference on computer vision and pattern recognition, 6810–6818.

Capítulo 2. Red Neuronal Convolucional para la Identificación Paramétrica de un robot cartesiano

Carlos Leopoldo Carreón Díaz de León[1], Sergio Vergara Limon[2], María Aurora D. Vargas Treviño[2], Juan Manuel González Calleros[1]
[1]Benemérita Universidad Autónoma de Puebla. Facultad de Ciencias de la Computación
[2]Benemérita Universidad Autónoma de Puebla. Facultad de Ciencias de la Electrónica
e-mail autor por correspondencia. carlos.carreond@alumno.buap.mx

Resumen. Los parámetros dinámicos de un robot son utilizados para emular complejos sistemas con el objetivo de diseñar algoritmos de control, diseño mecánico, y diseño de procesos. Sin embargo, las metodologías convencionales toman tiempo en hallar los parámetros de un robot real debido a los sensores y actuadores utilizados. En este trabajo se utiliza una red neuronal convolucional (RNC) para extraer los parámetros dinámicos de un robot cartesiano de dos grados de libertad, utilizando una trayectoria senoidal. Se utiliza una técnica que introduce las señales experimentales del robot y las convierte a una imagen para que la red devuelva el residuo paramétrico dado un conjunto inicial de parámetros, posteriormente él se adecua los resultados para devolver valores dentro y fuera del rango de -1 a 1. Se diseña una métrica de evaluación tempo-espectral que muestra una similitud de 91.96%.

Palabras Clave: Red Neuronal Convolucional, Parámetros dinámicos, Robot cartesiano.

1 Introducción

Los sistemas robóticos complejos utilizan mecanismos automatizados para efectuar una trayectoria deseada mediante el uso de actuadores y sensores en cada articulación. El control de movimiento es una tarea esencial ya que la precisión del robot depende de que tan acertada sea la acción de control sobre los actuadores. En la bibliografía se encuentra que los mejores controladores de movimiento utilizan compensación del modelo dinámico del robot y muestran un excelente desempeño (Kelly, & Santibáñez, 2003). También se encuentran trabajos donde se linealiza el modelo del robot para utilizar los parámetros dinámicos (Petko et. al., 2016). La interacción humano-robot es otra área de investigación donde los parámetros dinámicos son necesarios ya que con ellos se puede determinar cuándo un robot tiene un comportamiento anormal debido a una colisión, exceso de fuerza, o cambios en su estructura (Haddadin, De Luca, & Albu-Schaffer, 2017). Las constantes asociadas al robot pueden clasificarse en tres categorías principales; inercia, fricción, y gravedad (Urrea, & Pascal, 2021). Ya que la gran mayoría de modelos matemáticos de robots son linealmente independientes a sus parámetros, se utiliza comúnmente el algoritmo de mínimos cuadrados (MC) para hallar los parámetros dinámicos (Zhang, Wang, Jing, & Tan, 2019). Un requerimiento crítico para MC es que todas las señales involucradas debe ser directamente medidas: la posición, velocidad, y aceleración son continuas en tiempo y amplitud con ruido

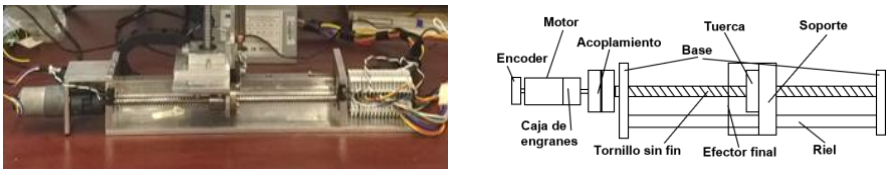
gaussiano aditivo. Los modelos dinámicos identificados con MC utilizan una matriz de observaciones y un vector de parámetros (Argin, &Bayraktaroglu, 2021). El problema con estos métodos recae en que la matriz de observaciones utiliza una señal de posición cuantificada, una estimación de velocidad y aceleración por lo que ya no son señales continuas.

La teoría de MC indica que las señales involucradas deben estar en el dominio de los números reales en su amplitud. Sin embargo, las señales del robot están en el conjunto de los números reales; por lo tanto, los métodos de identificación basados en MC no funcionan adecuadamente tal y como se predice en su teoría. En aplicaciones reales de métodos basados en MC, se utilizan distintas técnicas para evitar los efectos de la cuantificación de la señal de posición. La más utilizada es hallar la mejor trayectoria para permitir una rápida identificación de parámetros (Swever et. al., 1997). También se utiliza MC ponderado para mejorar los métodos basados en MC simple (Gautier, & Poignet, 2001). Para recuperar una señal submuestreada se utiliza un filtro pasa-bajas (Ogata, 1995), sin embargo, lo mismo no sucede para una señal cuantificada. Para estimar la velocidad se utilizan técnicas de medición de tiempo entre flancos un encoder (Hace, & Čurkovič, 2018). El sensor más comúnmente utilizado en articulaciones rotacionales son los encoders incrementales cuyas mediciones de posición pueden ser mejoradas para el caso de encoders ópticos (Ye et. al., 2015), encoders análogos (Benammar, & Gonzales, 2016), y encoders electromagnéticos (Zhang et. al., 2015). Los robots reales no pueden adquirir y usar la posición, velocidad, y aceleración en forma continua.

Las redes neuronales (RN) han demostrado ser de gran utilidad en casos donde la información es incompleta (Gu et. al., 2018). Las RN pueden estimar señales de robots cuando el sensor utilizado para esa medición no es utilizado, tal y como se muestra en Peng et. al. (2021) donde implementan un control de admitancia sin un sensor de fuerza. También las RN pueden compensar fenómenos de fricción en modelos de robots; en (Liu, Wang, & Wang, 2021) se ha implementado un estimador de fricción para cambios de temperatura. Las RNC tienen un mejor comportamiento para el procesamiento de señales. En Wu, & Jahanshahi (2019) se estima el movimiento de tres sistemas mecánicos y los autores concluyen que la RCN supera a una RC completamente conectada. La contribución de este artículo es la aplicación de una RNC para extraer desde una imagen los parámetros dinámicos de un robot cartesiano. La imagen utiliza las señales de torque, posición, y un conjunto inicial de parámetros dinámicos del robot. La RCN identifica cuando la imagen contiene parámetros dinámicos que no corresponden con el torque y el movimiento del robot. El robot cartesiano sigue una trayectoria predefinida con un control proporcional saturado. Una métrica tempo-espectral muestra la similitud entre las señales de torque experimental y el torque reconstruido con los parámetros dinámicos. Los resultados muestran que la propuesta puede identificar los parámetros en menos de 1 segundo. Este artículo muestra algunos de los resultados publicados en Carreon et. al. (2022). El artículo está organizado de la siguiente forma: En la sección 2 se encuentra la descripción del robot cartesiano, la sección 3 muestra la creación de la imagen, la sección 4 muestra la red neuronal y sus datos de entrenamiento. La sección 5 se muestran los resultados y en la sección 6 las conclusiones.

2 Descripción del robot

La RCN propuesta identifica los parámetros de la articulación horizontal del robot de la Fig. 1. El actuador de esta articulación está compuesto de un motor de corriente directa conectado a una caja de engranes. Un encoder magnético mide la posición del eje del motor: la reducción de la caja de engranes determina la posición angular mediante la división de la posición angular entre 131.125 unidades.



a) b)

Fig. 1. Articulación horizontal del robot cartesiano, a) fotografía, b) diagrama esquemático.

La articulación de la Fig. 1 está soportada por una placa de aluminio. El riel de acero otorga la rigidez necesaria para mover el efector final. Las conexiones mecánicas del robot r_{01}, r_{12}, y r_{23}, utilizan un resorte para modelar la rigidez y un amortiguador para la fricción de la unión mecánica. Las variables x_0, x_1, y x_2 denotan la posición angular del motor, la caja de engranes y el tornillo sin fin. La variable x_4 denota la posición del efector final. Los momentos de inercia y la masa del efector final están en los coeficientes I_o, I_1, I_2, y I_3. Las constantes v_0, v_1, v_2, y v_3 representan la viscosidad lineal. Nótese que el acoplamiento mecánico r_{01} representa todos los engranes dentro de la caja de reducción. Se incluye la fricción de Coulomb en el riel de acero en el contacto del efector final. El modelo dinámico está (1) donde el torque debido a las conexiones es $\tau_{01}, \tau_{12}, \tau_{23}$, g_r es la reducción de la caja de engranes, a_1, a_2, a_3 son los coeficientes de rigidez, c_1, c_2, c_3 son los coeficientes de los amortiguadores de las conexiones, y l_n es el coeficiente de transmisión de la tuerca fijado en $2\pi/4 \times 10^{-3}$.

$$\begin{aligned} \tau_{01} &= a_1(x_0 - g_r x_1) + c_1(\dot{x}_0 - g_r \dot{x}_1) & \ddot{x}_0 &= (\tau_0 - v_0 \dot{z}_0 - \tau_{01})/I_0 \\ \tau_{12} &= a_2(x_1 - x_2) + c_2(\dot{x}_1 - \dot{x}_2) & \ddot{x}_1 &= (g_r \tau_{01} - v_1 \dot{x}_1 - \tau_{12})/I_1 \\ \tau_{23} &= a_3(x_2 - l_n x_3) + c_3(\dot{x}_2 - l_n \dot{x}_3) & \ddot{x}_2 &= (\tau_{12} - v_2 \dot{x}_2 - \tau_{23})/I_2 \\ \ddot{x}_3 &= \left[l_n \tau_{23} - v_3 \dot{x}_3 - ksign(\dot{x}_3) - \tau_{of}\right]/I_3 \end{aligned} \quad \textbf{(1)}$$

donde τ_0 es el torque del motor y τ_{of} es el par de bias. Considerando que las deformaciones de las conexiones mecánicas están por debajo de $4 \times 10^{-3}/8400$ [m], que es la resolución del encoder, la ecuación (1) se simplifica dando como resultado la ecuación equivalente (2):

$$\tau = I_n \ddot{x} + v_n \dot{x} + k_n sign(\dot{x}) + \tau_g \quad \textbf{(2)}$$

donde $\tau = g_r\tau_0,\ I_n = g_r^2 I_0 + I_1 + I_2 + (1/l_n^2)I_3,\ v_n = g_r^2 v_0 + v_1 + v_2 + (1/l_n^2)v_3,$ $k_n = (1/l_n^2)k,\ \ \tau_g = (1/l_n^2)\tau_{of},\ \ x \approx g_r x_0,\ \ x \approx x_1,\ \ x \approx x_2,\ \ x \approx (1/l_n)x_3.$ Los coeficientes por extraer son $C = \left[I_n, v_n, k_n, \tau_g\right]^T$.

3 Creación de la imagen

La imagen necesita del movimiento del robot, el par, y un conjunto de parámetros dinámicos para su creación. Asumiendo que la articulación horizontal es modelada por (2), se considera factible reconstruir el torque con los parámetros dinámicos de C. La idea principal detrás de la imagen Z es que cuando los parámetros C reconstruyen adecuadamente el torque, entonces Z contiene aproximadamente el torque al cuadrado. En caso contrario, Z contiene otro tipo de señales diferentes al valor cuadrado del torque. La Fig. 2-a) muestra una imagen Z_p donde el subíndice p indica que los parámetros están muy cerca de los que caracterizan al torque. La Fig. 2-b) muestra una imagen $Z_n$donde los parámetros no son los adecuados. Obsérvese que la diferencia entre ambas imágenes muestra formas diferentes al par elevado al cuadrado, como muestra la Fig. 2-c). La imagen necesita ser tan pequeña como sea posible para reducir el tiempo de ejecución de la RNC; la imagen contiene la información condensada del robot. En (3) se muestra una técnica de submuestreo de una señal z utilizando la transformada discreta del coseno Q, donde z_s es la señal submuestreada. El espectro de frecuencia original es cortado a la longitud deseada en forma independiente a la longitud original.

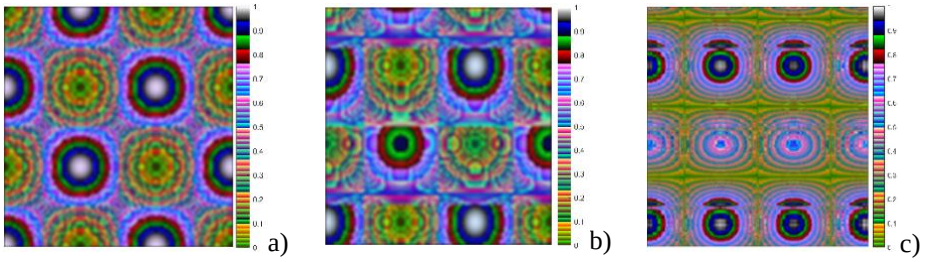

a) b) c)

Fig. 2. Imágenes creadas con las señales de un robot cartesiano para identificación paramétrica.

$$\{z_s = q, |q = Q^{-1}[Q_z(i)] \wedge 1 \le i \le 100\} \qquad a(i) = \left(\sqrt{2/N}\right)[1 + \delta(i-1)]^{-0.5}$$
$$Q_z(i) = \sum_{j=1}^{N} a(i)z(j) \cos\cos\left[(\pi/2N)(2j-1)(i-1)\right] \qquad \delta(0) = 1, \delta(x) = 0\ \ \forall|x| > 0 \qquad \textbf{(3)}$$

Las señales submuestreadas contienen solo 100 muestras: Las señales del robot contienen información repetitiva, por lo que los parámetros dinámicos pueden ser deducidos únicamente con la información más relevante. El método M estima las

derivadas de la posición y velocidad utilizando las señales del encoder (Hace, & Čurkovič, 2018). La ecuación (4) filtra la velocidad y la aceleración para tener señales suaves. El coeficiente u es fijado en 1×10^{-4}, σ es la frecuencia del espectro Q.

$$z_f = Q^{-1}\left[10^{([Q(z)] - u\sigma / lnln\,(10)}\right] \quad \textbf{(4)}$$

La ecuación (5) muestra la construcción de la imagen Z. Primero, un conjunto inicial de parámetros dinámicos denotado por C_s reconstruye el torque utilizando las señales filtradas identificadas por el subíndice $(.)_{fs}$. La matriz A guarda los valores del par actual τ_{fs} multiplicado por el error del par. Finalmente, Z es la normalización de A.

$$\begin{gathered} C_s = [I_s, v_s, k_s, \tau_s]^T \qquad \tau_r = C_{s1}\ddot{z}_{fs} + C_{s2}\dot{z}_{fs} + C_{s3} sign(\dot{z}_{fs}) + C_{s4} \\ A(i,j) = \tau_{fs}(j)\left[2\tau_{fs}(i) - \tau_r(i)\right], i, j \in \{0,1,\ldots,100\} \\ Z = [A - A(i,j)\,]/[A(i,j)\; - A(i,j)\,] \end{gathered} \quad \textbf{(5)}$$

Los datos de entrenamiento son creados con (5) utilizando 10,000 conjuntos de parámetros dinámicos. Todas las señales de torque de la ecuación (2) son construidas por la simulación para cada conjunto de parámetros. La trayectoria predefinida es $x_d = sin\, sin\,(t)$, $\forall t \in [0, \pi]$, donde el paso de integración es fijado en 2.5ms utilizando Runge-Kutta.

4 Diseño de la Red Neuronal Convolucional

La RCN extrae los residuos paramétricos de la imagen creada por (5); si los parámetros C_s no corresponden al torque τ, la RCN devuelve $C_a - C_s$ donde C_a es el conjunto real de parámetros del robot. La propuesta de diseño de la RCN consiste en tres capas convolucionales (Y_0, Y_1, Y_2), y tres capas completamente conectadas (RNCC), como se muestra en la Fig. 3. El diseño contiene dos capas de reducción por valor máximo (X_1, X_2) con una ventana de 2 por 2. La función de activación para las capas 1 a 5 es $f[X(i,j)] = [0, X(i,j)]$, y para la capa 6 es $f[X(i,j)] = [1 + exp\; exp\; [-X(i,j)]\,]^{-1}$. La Tabla 1 muestra el tamaño de la RNC. La ecuación (6) denota la operación de convolución utilizada, donde d y o son los índices de entrada de X y Y, respectivamente, se usa un bias por canal de mapa de salida de las capas convolucionales; se utiliza una matriz B cuyos elementos son los bias, $W_{d,o}$es el kernel convolucional de tamaño $S_w \times S_w$ (Gu et. al., 2018).

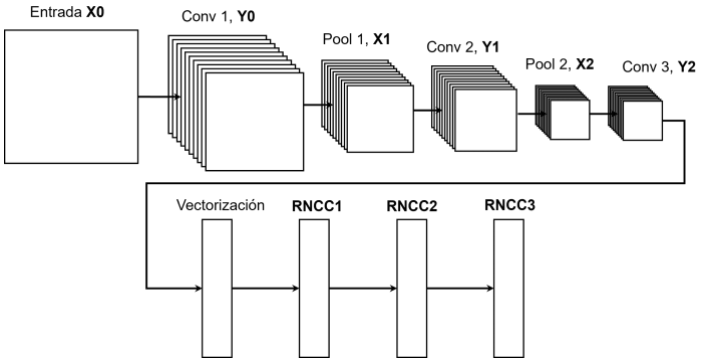


Fig. 3. Red Neuronal Convolucional diseñada.

Cada imagen del conjunto de datos de entrenamiento es etiquetada con el residuo paramétrico $C_a - C_s$. Debido a que la función de activación de la capa de salida está en el rango de 0 a 1, todas las etiquetas son igual a $L_t = [1 + exp\ exp\ (C_a - C_s)\ ]^{-1}$.

$$Y_o(i,j) = \sum_{d=0}^{D-1} \left[\sum_{u=0}^{S_w} \sum_{v=0}^{S_w} W_{d,o}(u,v) X_d(u+i, v+j) \right] + b_o B_0 \quad \textbf{(6)}$$

Tabla 1. Tamaño de la RCN propuesta. T: Capa de reducción, V: vectorización.

Capa	Tamaño de entrada	Kernel/Pesos	Tamaño de salida
1	100x100	9x9x1x10	$46x46x10^{T}$
2	46x46x10	5x5x10x10	$21x21x10^{T}$
3	21x21x10	3x3x10x10	$3610x1^{V}$
4	3610x1	100x3610	100
5	100	100x100	100
6	100	4x100	4

La RCN es entrenada con el algoritmo de propagación del error hacía atrás y los datos de entrenamiento. La función de costo del error es $E = 0.125 \sum_{i=1}^{4} (L_t - FFNN_3)^2$. Aplicando la función inversa de activación a la capa de salida, se recupera el residuo $C_r = C_a - C_s$. Agregando a C_s el residuo C_r, se obtiene finalmente C_a. La métrica de evaluación tempo-espectral $\phi(\tau, \tau_r)$ de la ecuación (7) devuelve la similitud entre el torque real τ y el reconstruido τ_r con los parámetros.

$$\phi(\tau,\tau_r) = \left[1 - \left(\frac{\left|\sum \quad \tau - \sum \quad \tau_r\right|}{\sum \quad (|\tau| + |\tau_r|)}\right)\right]^{0.4} \left[1 - \left(\frac{\sum \quad |Q(\tau) - Q(\tau_r)|}{\sum \quad (|Q(\tau)| + |Q(\tau_r)|)}\right)\right]^{0.4} \tag{7}$$

5 Resultados

Esta sección muestra los resultados de entrenamiento de la RCN, una prueba con una simulación de la ecuación (2), y los resultados experimentales. La RCN ha sido entrenada utilizando Matlab© con una GPU GTX1060 de Nvidia© y un procesador i7 de Intel©. El proceso de entrenamiento divide el conjunto de datos en dos partes iguales; un conjunto S_{tr} para entrenar, y otro conjunto S_{ts} para validar el entrenamiento. La Fig. 4 muestra la evolución del costo del error para $N_t = 4.5 \times 10^5$ iteraciones de entrenamiento. Se observa que la función de costo de ambos conjuntos sigue la misma tendencia, sin embargo, el costo del conjunto de validación está ligeramente por encima lo que indica que la RCN no está sobre ajustada.

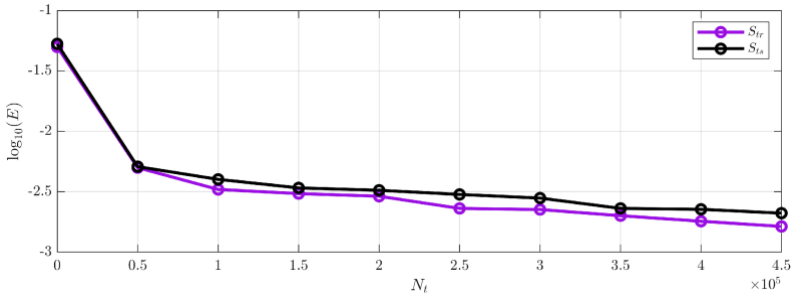


Fig. 4. Evolución de la función de costo.

Las señales demuestran que la RNC puede generalizar para nuevos datos que no han sido incluidos en el entrenamiento. Debido a que los parámetros de la ecuación (2) pueden ser ajustados a cualquier frecuencia y amplitud de la señal original, se puede utilizar una trayectoria diferente para la simulación y los datos experimentales igual a $x_d = 2\pi\ sin\ sin\ (0.2875t)$ (ver Fig. 5), donde los parámetros son escalados por la ecuación (8):

$$z_e = a\ sin\ sin\ (\omega t) \qquad I = C_{a1}/a\omega^2 \qquad v = C_{a2}/a\omega \tag{8}$$

donde z_e denota la trayectoria de cualquier frecuencia y amplitud. La RNC es probada con una simulación de la ecuación (2) donde los parámetros utilizados son mostrados junto con los parámetros experimentales. La métrica de similitud muestra que las señales de torque reconstruidas y las reales son muy similares. La Fig. 6 muestra el torque de simulación y el reconstruido. La articulación horizontal del robot de la Fig. 1 sigue la trayectoria predefinida como se muestra en la Fig. 8. Los parámetros dinámicos extraídos para los datos experimentales reflejan que el modelo de la ecuación (2) aproxima muy bien la realidad.

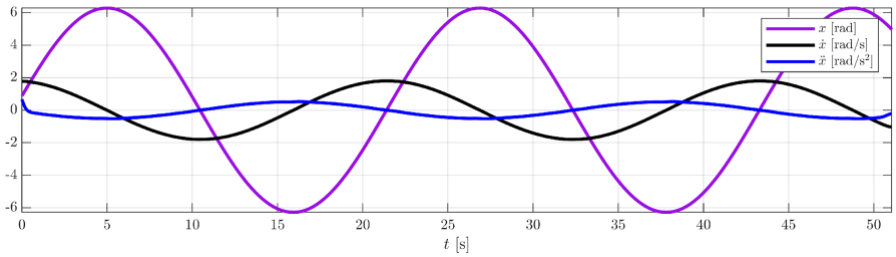


Fig. 5. Trayectoria experimental de la articulación horizontal.

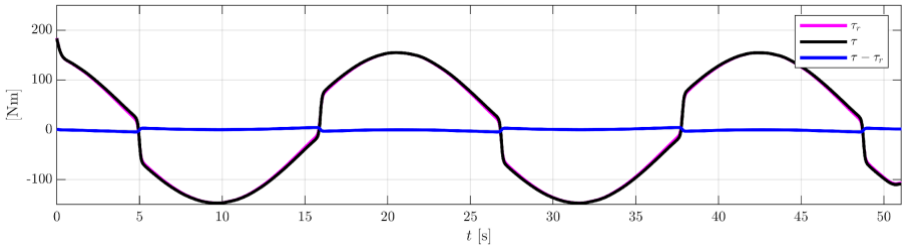


Fig. 6. Torque de simulación con su reconstrucción.

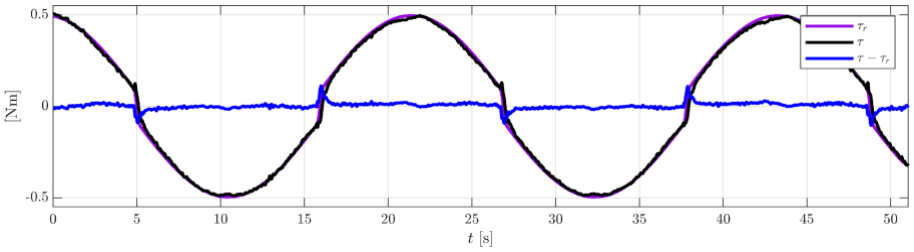


Fig. 7. Torque experimental y su reconstrucción.

Tabla 2. Resultados de identificación paramétrica.

Parámetros	Valores de simulación	Valores extraídos por la RNC para la simulación	Valores experimentales con la RCN
I kg·m^2	50	51.4989	0.0041
v kg·m^2·s	60	62.2023	0.2302
k N·m	40	35.85	0.0817
τ_g N·m	4	4.1202	0.0001
$\phi(\tau, \tau_r)$	-	99.16%	91.96%

El torque experimental y el reconstruido se muestra en la Fig. 7. El controlador proporcional utilizado es $\tau = 0.4405[100(x_d - x)]$.

6 Conclusiones

En este artículo se ha presentado una RNC que identifica los parámetros dinámicos de la articulación horizontal de un robot cartesiano. La RNC extrae los residuos paramétricos a partir de una imagen creada con las señales del robot para reconstruir el par con el modelo dinámico. Los datos de entrenamiento no requieren ningún tipo de trayectoria óptima para que la RNC funcione adecuadamente. La trayectoria utilizada es escalable en frecuencia y amplitud sin perder su forma esencial. Esta característica de la trayectoria hace el método de identificación propuesto atractivo ya que sin importar que frecuencia o amplitud se utilice, la RNC podrá extraer los parámetros dinámicos. La RNC funciona en forma diferencial; la red solo responde en función del residuo paramétrico, pero no en función del valor absoluto de los parámetros dinámicos. Este comportamiento permite obtener buenos resultados de entrenamiento. Los resultados de simulación muestran que la RNC devuelve los parámetros de simulación con una similitud cercana al 100%, todo en tan solo 0.047 segundos. Los datos experimentales de la articulación vertical crean una imagen que es introducida a la RNC y los parámetros extraídos aproximan el torque experimental en solo 0.8126 segundos. Esta contribución muestra que la RNC puede generalizar su aprendizaje para hallar parámetros dinámicos con datos no incluidos en el entrenamiento. Este artículo concluye con el hecho de que la RNC es adecuada para aplicaciones reales de señales de robots. Consecuentemente, el trabajo futuro de esta investigación es la aplicación de este método a otros tipos de robots.

Referencias

1. Argin, O. F., and Bayraktaroglu, Z. Y. (2021). *ConsistentdynamicmodelidentificationoftheStäubliRX*-160 industrial robot using convex optimization method", Journal of Mechanical Science and Technology, vol. 35, no. 5, pp. 2185-2195.
2. Benammar, M., and Gonzales, A. S. P. (2016). "*Position Measurement Using Sinusoidal Encoders and All-Analog PLL Converter With Improved Dynamic Performance*", IEEE Transactions on Industrial Electronics, vol. 63, no. 4, pp. 2414-2423.
3. Carreón Díaz de León, C. L., Vergara Limon, S., Vargas Treviño, M. A. D., and González Calleros, J. M. (2022). "*A novel methodology of parametric identification for robots based on a CNN*", Journal of Intelligent &Fuzzy Systems, vol. Preimpreso, no. Preimpreso, pp. 1-14, doi: 10.3233/JIFS-219246.
4. Gautier, M., and Poignet, P. (2001). *"P Extended Kalman filtering and weight edl east squares dynamic identification of robot"*, Control Engineering Practice, vol. 9, pp. 1361-1372.
5. Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., and Chen, T. (2018). *"Recent advances in convolutional neural networks",* Pattern Recognition, vol. 77, pp. 354-377.
6. Hace, A., and Čurkovič, M. (2018). "*Accurate FPGA-Based Velocity Measurement with an Incremental Encoder by a Fast Generalized Division less MT-Type Algorithm*", Sensors, vol. 18, no. 10, pp. 1-28.

7. Haddadin, S., De Luca, A., and Albu-Schaffer, A. (2017). "*Robot Collisions: A Survey on Detection, Isolation, and Identification*", IEEE Transactions on Robotics, vol. 33, no. 6, pp. 1292-1312.
8. Kelly, R., and Santibáñez, V. (2003). "*Control de Movimiento de Robots Manipuladores*", PEARSON Prentice Hall, Madrid.
9. Liu, S., Wang, L., and Wang, X. V. (2021). "*Sensor less force estimation for industrial robots using disturbance observer and neural learning of friction approximation*", Robotics and Computer-Integrated Manufacturing, vol. 71, pp. 1-11.
10. Ogata, K. (1995). "*Discrete-Time Control Systems*", Prentice Hall, New Jersey.
11. Peng, G., Chen, C. L. P., He, W., and Yang, C. (2021). "*Neural-Learning-Based Force Sensor less Admittance Control for Robots With Input Dead zone*", IEEE Transactions on Industrial Electronics, vol. 68, no. 6, pp. 5184-5196.
12. Petko, M., Gac, K., Góra, G., Karpiel, G., and Ochoński, J. (2016). "*CNC system of the 5-axis hybrid robot formilling*", Mechatronics, vol. 37, pp. 89-99.
13. Swevers, J., Ganseman, C., Tukel, D. B., De Schutter, J., and Van Brussel, H. (1997). "*Optimal Robot Excitation and Identification*", IEEE Transactions on Robotics and Automation, vol. 13, no. 5, pp. 730-740.
14. Urrea, C., and Pascal, J. (2021). "*Design and validation of a dynamic parameter identification model for industrial manipulator robots*", Archive of Applied Mechanics, vol. 91, no. 5, pp. 1981-2007.
15. Wu, R. T., and Jahanshahi, M. R. (2019). "*Deep Convolutional Neural Network for Structural Dynamic Response Estimation and System Identification*", Journal of Engineering Mechanics, vol. 145, no. 1, pp. 1-25.
16. Ye, G., Liu, H., Fan, S., Li, X., Yu, H., Lei, B., Shi, Y., Yin, L., and Lu, B. (2015). "*Precise and robust position estimation for optical incremental encoders using a linearization technique*", Sensors and Actuators A: Physical, vol. 232, pp. 30-38.
17. Zhang, S., Wang, S., Jing, F., and Tan, M. (2019). "*Parameter estimation survey formulti-joint robot dynamic calibration case study*", Science China Information Sciences, vol. 62, no. 10, pp. 1-15.
18. Zhang, Z., Dong, Y., Ni, F., Jin, M., and Liu, H. (2015). "*A Method for Measurement of Absolute Angular Position and Application in a Novel Electromagnetic Encoder System*", Journal of Sensors, vol. 2015, pp. 1-10.

Capítulo 3. Reconocimiento automático de Lengua de Señas mediante una red neuronal BiLSTM

Daniel Sánchez-Ruiz, J. Arturo Olvera-López, Ivan Olmos Pineda
Benemérita Universidad Autónoma de Puebla. Facultad de Ciencias de la Computación
e-mail autor por correspondencia. daniel.sanchez.4712@mail.com

Resumen. La población que padece de audición limitada o la pérdida total de la misma tiene como uno de sus grandes retos el de la comunicación con la población hablante. Los avances tecnológicos han permitido tener mejores interacciones; esto se ha logrado gracias a los avances en el área del procesamiento de lengua de señas, donde se definen tres grandes áreas: el reconocimiento, la generación y la traducción automática de las distintas lenguas de señas. En este trabajo se realiza el reconocimiento automático de lengua de señas empleando una red neuronal BiLSTM, se presentan algunos resultados iniciales y se discuten los siguientes pasos para el trabajo futuro.

Palabras Clave: Reconocimiento de lengua de señas, reconocimiento de patrones, visión por computadora.

1 Introducción

Actualmente se calcula que 430 millones de personas tienen alguna discapacidad de pérdida de audición en todo el mundo, lo cual representa alrededor del 5% de la población total, y se estima que para el año 2050 habrá 700 millones de personas con este tipo de discapacidad. Tener una discapacidad auditiva significa que una persona no es capaz de escuchar del mismo modo que lo hace una persona oyente (en umbrales de audición de 25 dB o más en ambas orejas) (WHO, 2022).

Existen dos categorías principales de discapacidad auditiva: problemas de audición, que se refiere a personas con pérdida de audición media a severa (que generalmente suelen comunicarse de forma hablada y que mejoran su capacidad auditiva a través de implantes o dispositivos auditivos), y la otra categoría es la de sordos, donde se tiene pérdida de audición de forma profunda, lo cual implica una audición extremadamente limitada o la supresión total de todo tipo de capacidad auditiva. Este último grupo de personas suele comunicarse a través de lengua de señas (WHO, 2022).

De acuerdo con la Federación Mundial de Sordos, existen alrededor de 300 lenguas de señas en el mundo y 70 millones de personas sordas que las usan (WDF, 2022). Las lenguas de señas hacen uso de señas gesticuladas a través de las manos que se complementan con expresiones faciales y lenguaje corporal. Este lenguaje es un medio de comunicación con el cual buscan poder transmitir sus sentimientos, ideas y necesidades. Al igual que los lenguajes hablados, las lenguas de señas se componen de

distintas reglas gramaticales y vocabulario dependiendo de la región, inclusive hay lenguas que cuentan con distintos componentes subregionales, por lo cual no se puede hablar de una lengua de señas universal.

Actualmente se busca mejorar la calidad de la vida de las personas a través de los distintos avances tecnológicos que se han desarrollado, y que se tienen planeados a futuro. No obstante, en el caso de las tecnologías de la comunicación, la gran mayoría sólo tienen soporte para lenguaje hablado o escrito, excluyendo a las lenguas de señas. Aunado a esto, existen pocas personas que dominen el conocimiento en alguna lengua de señas, por lo cual la comunidad sorda sufre de una barrera de comunicación considerable respecto a la mayoría de las personas (Bragg et al., 2019).

Un sistema de reconocimiento automático de lengua de señas puede ser clasificado en tres categorías con base a las señas que se busca reconocer: Deletreo con dedos, palabras aisladas y oraciones con señas continuas (Mohandes et al., 2014). El deletreo de palabras es usado en situaciones donde palabras nuevas, nombres de personas, lugares o palabras no tienen una seña definida, por lo cual tienen que ser "deletreadas" por movimientos de manos. En la categoría del reconocimiento de palabras aisladas, por cada dato de entrada sólo se analiza el significado de una única seña. Finalmente, en la última categoría de reconocimiento de oraciones (continuo) se busca reconocer el contenido de una conversación, la cual está compuesta de múltiples señas (Kamal et al., 2019).

El trabajo está organizado de la siguiente manera: en la Sección 2 se aborda un análisis del trabajo relacionado para el reconocimiento continuo de lengua de señas; en la Sección 3 se describe la metodología que se propone para la resolución del problema; la Sección 4 muestra algunos resultados obtenidos; finalmente, en la Sección 5 se listan las conclusiones y el trabajo futuro a realizar en la investigación.

2 Trabajo Relacionado

El objetivo del reconocimiento de lengua de signos es establecer diferentes métodos y algoritmos que puedan reconocer signos ya desarrollados y percibir su significado. A continuación, se muestran trabajos que se ha realizado tanto para el reconocimiento de lengua de señas continuo como para el aislado.

Koller et al. (2016) demostraron el uso de un algoritmo basado en la Esperanza-Maximización (EM) para el reconocimiento continuo del lenguaje de señas. El algoritmo basado en EM se diseñó para abordar el problema de alineación temporal asociado con las tareas del procesamiento continuo de videos. Otro experimento de Camgoz et al. (2017) desarrolló un sistema integral diseñado para la alineación y el reconocimiento continuos del lenguaje de señas; el modelo diseñado se basa en el modelado explícito de subunidades. Del mismo modo, Wang et al. (2018) sugirió un método de fusión temporal conexionista que tiene la capacidad de traducir lenguajes visuales continuos en videos en oraciones de lenguaje textual.

Al explorar los desafíos de la traducción continua, Pigou et al. (2017) observó que las redes residuales profundas se pueden usar para aprender patrones en videos continuos que contienen gestos y signos. El uso de redes residuales profundas puede minimizar la necesidad de preprocesamiento. Cui et al. (2017) también sugirió un

enfoque débilmente supervisado que podría reconocer el lenguaje de señas continuo con la ayuda de redes neuronales profundas. Este enfoque logró un resultado comparable a los enfoques más avanzados.

Hasta hace poco, la mayoría de los experimentos de reconocimiento de lenguaje de señas se habían llevado a cabo en muestras de señas aisladas. Un experimento de Konstantinidis et al. (2018) propuso un sistema de reconocimiento de lengua de señas aislado diseñado para extraer aspectos discriminativos de videos, donde cada video firmado correspondía a una palabra. Fang et al. (2017) sugirió el uso de un modelo jerárquico basado en redes neuronales recurrentes profundas. El modelo combinó con éxito las características aisladas de la lengua de señas estadounidense de bajo nivel en una representación organizada de alto nivel que podría usarse para la traducción.

Los desarrollos recientes en los experimentos con el lenguaje de señas también han sugerido que el uso de regiones de interés (ROI) para aislar los gestos de las manos y las características del lenguaje de señas puede mejorar la precisión del reconocimiento (Bantupalli et al., 2018). En Kumar et al. (2018), los autores utilizaron un sistema de reconocimiento de brillo aislado para facilitar la traducción del lenguaje de señas en tiempo real. El sistema de reconocimiento de brillo aislado incluía preprocesamiento de video, así como un módulo de red neuronal de serie temporal. Un estudio reciente de Huang et al. (2018) se centró en una tarea básica aislada del reconocimiento de lengua de señas. Se propuso el uso de una red 3D-CNN basada en la atención para reconocer un amplio vocabulario. El modelo fue ventajoso porque aprovechó las capacidades de aprendizaje de características espaciotemporales de este tipo de redes.

En el reconocimiento continuo de lengua de señas es necesario alinear cuidadosamente las secciones del video en orden cronológico y asegurarse de que cada oración esté etiquetada correctamente. Esto debe tenerse en cuenta durante la evaluación de metodologías, así como el proceso de selección de características. Si el etiquetado secuencial se realiza correctamente y se seleccionan las características más predictivas, el modelo resultante tiene una mayor probabilidad de ser preciso con análisis de video continuo.

3 Metodología

Para la solución del problema del reconocimiento continuo se propone la metodología presentada en la Figura 1, donde de manera inicial se requiere de un conjunto de datos, posteriormente se identifican y segmentan las regiones de interés, con las cuales se hará una etapa de extracción de características, las cuales servirán para la última fase donde a través de un método de reconocimiento se identificarán y evaluarán los significados de las distintas señas. Dentro de esta metodología en la etapa de extracción de características se analiza la utilidad de hacer la extracción de características que no han sido consideradas como la estimación de la mirada y la inclinación de la cabeza.

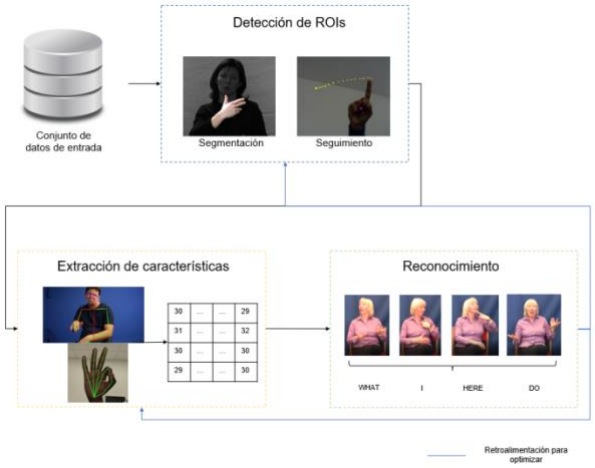


Fig. Metodología general propuesta para el reconocimiento continuo de lengua de señas.

Con base en la metodología descrita, se detallan las actividades realizadas para la detección y segmentación de las regiones de interés, apartado en el cual se tienen avances hasta el momento. Cabe hacer mención que las regiones de interés que se consideran son las manos, el rostro y también se consideran los puntos clave relacionados con la estimación de la postura del cuerpo, todo ello se justifica en el hecho de tomar en cuenta características manuales y no manuales.

Para los experimentos realizados para la detección de las regiones de interés y la estimación de los puntos clave referentes a la postura del cuerpo, se ocupó el conjunto de datos LIBRAS (Quadros et al., 2021), en particular el de la región de Florianópolis, el cual contiene 639 registros, cada uno compuesto de videos con una resolución de 640x414 pixeles, capturados con una tasa de refresco de 30 fotogramas por segundo y un archivo de anotación de las señas.

Para la tarea de detectar la región de las manos y del rostro se ocupó el sistema YOLOv5 (Jocher et al., 2021), el cual es el estado del arte en la tarea de detección de objetos. Sin embargo, dado que las características que están presentes en los datos son muy específicas (deformaciones y oclusiones en la región de las manos y del rostro), se realiza el entrenamiento para obtener un modelo de inferencia con un subconjunto de imágenes de LIBRAS previamente anotado.

En el apartado de la estimación de la postura del cuerpo se ocupó el sistema OpenPose (Cao et al., 2019), el cual además de estimar los puntos clave referentes a las extremidades del cuerpo, también tiene la posibilidad de estimar puntos clave referentes a la región de la mano y del rostro.

Una de las propuestas respecto al trabajo relacionado es hacer la extracción de características no manuales con base a la estimación de la mirada y la inclinación de la cabeza; dichos descriptores no han sido estudiados en profundidad como se aprecia en algunos trabajos (Koller, 2020; Rastgoo et al., 2021). La herramienta empleada para esta tarea es OpenFace (Baltrusaitis et al., 2018), con la cual se pueden obtener diversas características relacionadas a la región del rostro y cabeza.

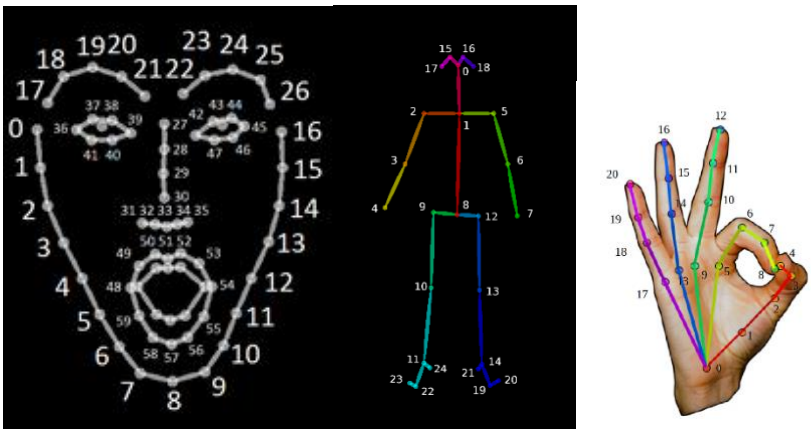
Una vez descritas las técnicas para hacer la detección, reconocimiento o hasta extracción de características de las diversas regiones de interés consideradas, a continuación, se establecen de forma puntual cada una de las características que serán empleadas en la etapa de reconocimiento y la forma en la que son calculadas.

- Coordenadas (x,y) de la mano dominante. Esto respecto al centroide del cuadro envolvente detectado con YOLOv5.
- Coordenadas (x,y) de la mano secundaria. Esto respecto al centroide del cuadro envolvente detectado con YOLOv5.
- Velocidad aproximada mano dominante y secundaria. Empleando la formula para estimar la velocidad entre dos puntos, siendo los puntos los centroides de los cuadros envolventes de la región de cada mano en dos instantes de tiempo distintos (cada 3 fotogramas), los cuales son inferidos con YOLOv5.
- Distancia euclidiana entre puntos clave referentes a expresiones faciales. Los puntos de OpenPose considerados se muestran en la Tabla 1 y se visualizan en la Figura 2 (a).
- Coordenadas (x,y) de puntos clave referentes a la región de los brazos. Los puntos que se consideran son (3, 4, 6, 7) y también se pueden visualizar en la Figura 2 (b).
- Distancia euclidiana entre puntos clave referentes a la región de las manos. Los puntos de OpenPose considerados se muestran en la Tabla 2 y se visualizan en la Figura 2 (c).
- Coordenadas angulares (x,y) de la dirección de la mirada. Coordenadas en radianes promediadas de ambos ojos y obtenidas con OpenFace.
- Rotación en radianes alrededor de los ejes X, Y, Z. Valores obtenidos con OpenFace que ayudan a estimar la postura de la cabeza.

Finalmente, para el reconocimiento de lengua de señas se hace uso de una red neuronal recurrente con memoria de largo y corto termino bidireccional (BiLSTM); este tipo de técnicas son ideales para problemas de reconocimiento de secuencias, es decir, de datos que se componen de diversos estados a lo largo de un periodo de tiempo. Este tipo de redes neuronales han demostrado ser útiles en trabajos relacionados de reconocimiento de lengua de señas (Koller, 2020; Rastgoo et al., 2021).

Tabla 1. Puntos clave considerados para expresiones faciales y obtenidos con OpenPose

Región	Puntos
Apertura vertical boca	51 – 57
Apertura horizontal boca	48 – 54
Labio superior – zona inferior nariz	51 – 33
Zona superior ojo izquierdo – zona central ceja izquierda	19 – 37
Zona superior ojo derecho – zona central ceja derecha	24 – 44



a) b) c)

Fig- 2. Visualización de puntos clave obtenidos con Open Pose para: a) la zona del rostro, b) extremidades del cuerpo y c) manos.

Tabla 2. Puntos clave considerados para las manos y obtenidos con OpenPose

	Puntos
Palma inferior – dedo medio inferior	0 – 9
Palma inferior – pulgar inferior	0 – 2
Palma inferior – meñique inferior	17 – 20
Anular inferior – anular superior	13 – 16
Dedo medio inferior – dedo medio superior	9 – 12
Índice inferior – índice superior	5 – 8
Pulgar inferior – pulgar superior	2 – 4

La topología de la red es la que se presenta en la Figura 3, la cual se compone de dos capas ocultas que componen la capa de la red BiLSTM, posteriormente se tiene una capa completamente conectada, que tiene como salida el número de etiquetas posibles a predecir; la capa que hace estas predicciones es una de tipo Softmax, que es la ocupada en problemas de clasificación.

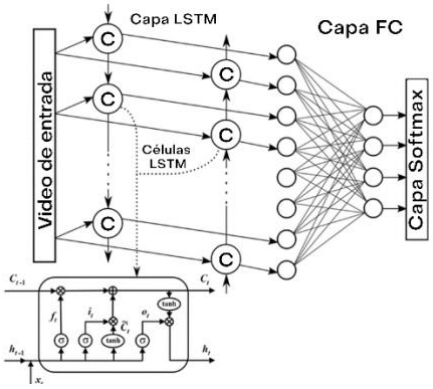


Fig. 3. Topología de la red BiLSTM empleada en la tarea de reconocimiento de lengua de señas.

4 Experimentos y Resultados

Para los experimentos realizados se ocupan videos del conjunto de datos LIBRAS (Quadros et al., 2021). Dado que la extracción de características es un proceso que recientemente se terminó y que por cada video se necesita un cierto tiempo, en este momento sólo se ocupan 4 registros de los 639, un video por cada registro.

Para el proceso del entrenamiento se decidió ocupar la herramienta en línea de Google Colab. Tanto para el proceso de entrenamiento como en el de evaluación, el total de los datos del conjunto de imágenes se dividió en tres grupos: uno de entrenamiento, uno de validación y uno de prueba; para esto se siguió la práctica en las investigaciones de detección de objetos de ocupar 70% para entrenamiento y 30% para pruebas.

Los parámetros que deben de definirse para el proceso del entrenamiento de la red BiLSTM son el número de épocas, el cual se estableció en 3y que se definió de forma empírica a través de los experimentos. Además, se define el tamaño del *batch,* en50. El número de células en cada capa oculta es 128; y el *learning rate*es de 0.003.

Adicionalmente, cabe resaltar que las clases para este problema de clasificación son las palabras que están relacionadas al significado de una gesticulación dada en un instante. La forma en la que se hace el etiquetado en el vector de características es mediante el archivo de anotaciones eaf asociado a cada video en cada registro.

La forma en la que se miden los resultados que se obtuvieron hasta este punto es a través de una gráfica que representa la función de perdida, la cual se muestra en la Figura 4; en ella se visualiza que el clasificador funciona con un porcentaje bajo de error. Sin embargo, resta explorar si esto puede deberse a que exista una o varias clases que tengan muchas instancias en comparación con el resto, es decir, que se tenga un conjunto de datos desbalanceado y estén generando predicciones de forma incorrecta.

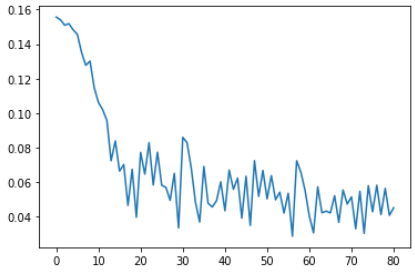


Fig. 4. Gráfica de función de perdida a través del proceso de entrenamiento.

Mediante una matriz de confusión se puede mostrar por cada una de las etiquetas el porcentaje de predicciones realizadas de forma correcta. En este tipo de graficas se tiene que visualizar en la diagonal de la matriz mediante un color (en este caso amarillo a verde) que por cada clase se tiene una alta precisión en las inferencias. En la Figura 5 se puede apreciar que hay varias predicciones hechas de forma incorrecta, por lo cual se debe de analizar si la topología empleada es la idónea, si las características aportan información relevante o redundante y si la inclusión de más videos en el proceso de entrenamiento ayuda.

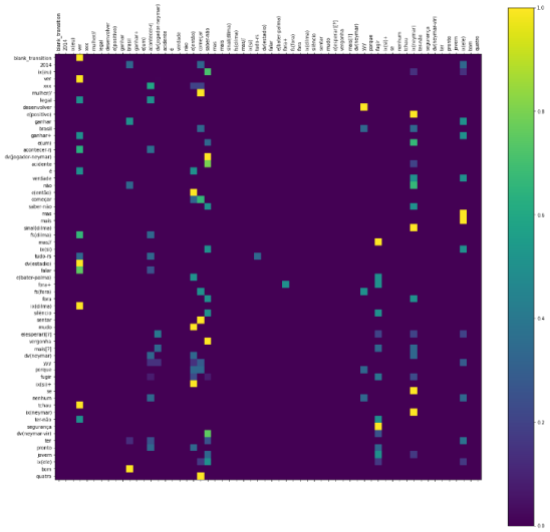


Fig. 5. Matriz de confusión de los resultados obtenidos.

5 Conclusiones y Trabajo Futuro

En el presente trabajo se destacó la utilidad y el estado actual del reconocimiento de lengua de señas. Posteriormente se propuso una metodología que aborda el reconocimiento de lengua de señas ocupando características basadas en componentes manuales y no manuales.

Una vez que se presentó el conjunto de datos a emplear, se definieron sus características, el número de datos y los videos que fueron considerados. Con dichos datos se realizaron los primeros experimentos que correspondieron a la extracción de características y reconocimiento de lengua de señas. Los resultados obtenidos mostraron que a pesar de tener un clasificador con un valor bajo en su función de perdida no tiene resultados altos por cada clase.

a la situación de los resultados obtenidos, como trabajo futuro es necesario hacer un análisis de la información que provee cada característica, la pertinencia para el problema de la topología de la red BiLSTM presentada y el uso de más datos en el proceso de entrenamiento. Para atacar estos puntos, se propone analizar la varianza de cada una de las características, emplear métodos de selección de características (ANOVA, Información mutua), de reducción de dimensionalidad de descriptores como PCA y también además de otro tipo de topología de red BiLSTM explorar otros métodos de clasificación como los enfoques de atención (Transformadores).

Referencias

1. Bantupalli, K., & Xie, Y. (2018). *American sign language recognition using deep learning and computer vision.* 2018 IEEE International Conference on Big Data (Big Data), pp. 4896-4899. IEEE.
2. Baltrusaitis, T., Zadeh, A., Lim, Y. C., y Morency, L. P. (2018). *Openface 2.0: Facial behavior analysis toolkit*. 2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018), pp. 59-66. IEEE.
3. Bragg, D., Koller, O., Bellard, M., Berke, L., Boudrealt, P., Braffort, A., Caselli, N., Huenerfauth, M., Kacorri, H., Verhoef, T., Vogler, C., y Morris, M. R. (2019). "*Sign Language Recognition, Generation, and Translation: An Interdisciplinary Perspective*". Proceedings of the 21st International ACM SIGACCESS Conference on Computers and Accessibility.
4. Camgoz, N. C., Hadfield, S., Koller, O., y Bowden, R. (2017). "*SubUNets: End-to-End Hand Shape and Continuous Sign Language Recognition*". Proceedings of the IEEE International Conference on Computer Vision, pp. 3075-3084.
5. Cao, Z., Hidalgo, G., Simon, T., Wei, S. E., y Sheikh, Y. (2019). "*OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields*". IEEE transactions on pattern analysis and machine intelligence, 43(1), 172-186.
6. Cui, R., Liu, H., y Zhang, C. (2017). "*Recurrent Convolutional Neural Networks for Continuous Sign Language Recognition by Staged Optimization*". 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1610-1618.
7. Fang, B., Co, J., y Zhang, M. (2017). *Deepasl: Enabling ubiquitous and non-intrusive word and sentence-level sign language translation.* Proceedings of the 15th ACM conference on embedded network sensor systems, pp. 1-13.

8. Huang, J., Zhou, W., Li, H., y Li, W. (2018). *Attention-based 3D-CNNs for large-vocabulary sign language recognition*. IEEE Transactions on Circuits and Systems for Video Technology, 29(9), 2822-2832.
9. Jocher, G., Stoken, A., Borovec, J., NanoCode012, ChristopherSTAN, Changyu, L. y Yu, L. (2021). "*ultralytics/yolov5: v4.0 - nn.SiLU() activations, Weights & Biases logging, PyTorch Hub integration*". Zenodo. doi:10.5281/zenodo.4418161.
10. Kamal, S. M., Chen, Y., Li, S., Shi, X., y Zheng, J. (2019). "*Technical Approaches to Chinese Sign Language Processing: A Review*". IEEE Access, vol. 7, pp. 96926-96935.
11. Koller, O. (2020). *Quantitative survey of the state of the art in sign language recognition*. arXiv preprint arXiv:2008.09918.
12. Koller, O., Ney, H., y Bowden, R. (2016). *Deep hand: How to train a cnn on 1 million hand images when your data is continuous and weakly labelled*. Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3793-3802.
13. Konstantinidis, D., Dimitropoulos, K., y Daras, P. (2018). *A deep learning approach for analyzing video and skeletal features in sign language recognition*. 2018 IEEE international conference on imaging systems and techniques (IST), pp. 1-6. IEEE.
14. Kumar, S. S., Wangyal, T., Saboo, V., y Srinath, R. (2018). *Time series neural networks for real time sign language translation*. 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 243-248. IEEE.
15. Mohandes, M., Deriche, M., y Liu, J. (2014). "*Image-Based and Sensor-Based Approaches to Arabic Sign Language Recognition*". IEEE Transactions on Human-Machine Systems, vol. 44, no. 4, pp. 551-557.
16. Pigou, L., Van Herreweghe, M., y Dambre, J. (2017). *Gesture and sign language recognition with temporal residual networks*. Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 3086-3093.
17. Quadros, R. M., Schmitt, D., Lohn, J., y de Arantes Leite, T. (2021). *Corpus de Libras*.
18. Rastgoo, R., Kiani, K., y Escalera, S. (2021). *Sign language recognition: A deep survey*. Expert Systems with Applications, 164, 113794.
19. Wang, S., Guo, D., Zhou, W. G., Zha, Z. J., y Wang, M. (2018). *Connectionist temporal fusion for sign language translation*. Proceedings of the 26th ACM international conference on Multimedia, pp. 1483-1491.
20. WDF, World Federation of the Deaf. (2022). *Our Work*. Recuperado de https://wfdeaf.org/our-work/.
21. WHO, World Health Organization. (2022). *Deafness and hearing loss*. Recuperado de https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss.

Capítulo 4. Preprocesamiento de imágenes para la detección multiclase de la Retinopatía Diabética

David Ferreiro Piñeiro, Ivan Olmos Pineda, José Arturo Olvera López
Benemérita Universidad Autónoma de Puebla. Facultad de Ciencias de la Computación
e-mail autor por correspondencia. david.ferreiro@viep.com.mx

Resumen. La disponibilidad de imágenes médicas para el desarrollo y posterior evaluación de sistemas automáticos de detección de enfermedades reviste hoy gran importancia. El desempeño de los modelos propuestos depende, en gran medida, de la calidad de los repositorios existentes. Estos repositorios públicos, son conformados bajo diferentes protocolos, por tal razón, existe una amplia variación de las características de las imágenes y eso se traduce en la necesidad de uniformar bajo ciertos parámetros los datos disponibles para garantizar las operaciones posteriores. Por estos motivos, en el presente trabajo se propone un método de preprocesamiento orientado a garantizar la homogeneización del tamaño de las imágenes determinando la región de interés experimental y realzando las estructuras vasculares aplicando una Ecualización Adaptativa de Histograma Limitado por Contraste (EAHLC) para facilitar las etapas posteriores de diagnóstico y clasificación.

Palabras Clave: Preprocesamiento, Ecualización Adaptativa de Histograma Limitado por Contraste, Retinopatía Diabética.

1 Introducción

El desarrollo de técnicas o métodos de análisis de imágenes médicas para el diagnóstico automático de enfermedades es un área de investigación activa que presenta resultados alentadores. Dentro de esta área, se investiga en el desarrollo de metodologías para la detección de la Retinopatía Diabética (RD) y posteriormente realizar su clasificación de acuerdo con su evolución clínica.

La RD es una microangiopatía que se desarrolla en pacientes diabéticos, como resultado de la obstrucción de los vasos sanguíneos que irrigan la retina debido a la acumulación de azúcar en la sangre. Esta enfermedad progresiva compromete el sistema visual humano, y se detecta a partir del análisis de lesiones que aparecen en el tejido de la retina: micro aneurismas, exudados duros y blandos, hemorragias, y nuevas vascularizaciones.

El diagnóstico clínico de la enfermedad desarrolla una revisión de las imágenes de la retina, resultando en un proceso lento, tedioso y subjetivo al depender de las habilidades de los oftalmólogos. Por estas razones, se han propuesto diferentes técnicas para realizar estos procesos de manera automática, buscando no remplazar a los especialistas humanos, sino más bien complementar y facilitar su trabajo.

Las técnicas propuestas, se basan generalmente en algoritmos de aprendizaje automático y su rendimiento depende, en gran medida, de la calidad de los datos disponibles para su desarrollo, por estos motivos se ha comenzado a recopilar imágenes en repositorios públicos orientados a garantizar los procesos de desarrollo-entrenamiento de los nuevos sistemas y su posterior evaluación de manera objetiva.

Las imágenes disponibles provienen de diferentes repositorios. Estos datos, presentan características disímiles derivadas de las condiciones en las que se realizaron las capturas, los dispositivos de procesamiento y su almacenamiento; razones que pueden dificultar el desarrollo de los procesos automatizados y por tanto resulta oportuno ejecutar las siguientes operaciones:

- Homogeneizar el tamaño de las imágenes de entrada.
- Determinar la región de interés experimental.
- Suavizar las imágenes para reducir el ruido aditivo presente.
- Aplicar alguna técnica para mejorar el contraste de las imágenes.

En la literatura se reportan algunas propuestas para abordar las problemáticas señaladas. Generalmente los trabajos se orientan al ajuste del contraste, brillo u otras propiedades de interés. En este sentido, es usual que realicen transformaciones en los espacios de color con el objetivo de limitar la influencia de los procedimientos sobre los valores de croma o facilitar la comprensión de los sistemas desarrollados al trabajar con modelos perceptualmente uniformes.

Otra de las problemáticas se relaciona con la limitada disponibilidad de datos, sobre todo en ambientes médicos, por lo que los autores desarrollan técnicas para aumentar artificialmente la cantidad de datos. Este aumento se logra aplicando transformaciones espaciales a las imágenes: rotaciones en diferentes ángulos, acercamientos o incluso, añadiendo ciertas cantidades de ruido. Como resultado se obtienen imágenes representativas de cada conjunto. Este tipo de abordaje es común cuando se pretenden emplear redes Neuronales Convolucionales (CNN).

En la Tabla 1 se resumen algunas de las técnicas de preprocesamiento frecuentemente utilizadas por otros autores.

Tabla 1. Técnicas de preprocesamiento para el diagnóstico de la RD.

Técnicas de preprocesamiento	Artículos
1- Modificación de las dimensiones de las imágenes, normalización de la iluminación y del contraste, separación de los canales RGB, transformaciones a escala de grises.	(Butt et al., 2019; Zhou et al., 2018)
2- Aumento artificial de la cantidad de datos disponibles, Ecualización Adaptativa de Histograma	(Hajabdollahi et al., 2019)
3- Ecualización Adaptativa de Histograma Limitado por Contraste (EAHLC) sobre el canal verde de la imagen RGB.	(Samanta et al., 2020)
4- Seguimiento ocular para determinar la región de interés	(Hsieh et al., 2021)

Técnicas de preprocesamiento		Artículos
5-	Detección de bordes de Canny interpolación, normalización, sobremuestreo y submuestreo de los datos.	(Saranya y Prabakaran, 2020)
6-	Ecualización del Histograma para la preservación del brillo basado en técnicas de estiramiento dinámico del histograma con cambios de espacio de color RGB-HSI-RGB.	(AbdelMaksoud et al., 2020)
7-	Ecualización Adaptativa del Histograma con apertura morfológica	(Das et al., 2021)

Es importante señalar que las técnicas o procedimientos propuestos durante el preprocesamiento de las imágenes dependen de la aplicación o de los objetivos específicos perseguidos. En las secciones siguientes se detallan los repositorios de imágenes públicos orientados específicamente al diagnóstico de la RD y el procedimiento propuesto para mejorar estas imágenes y garantizar los procesos subsecuentes.

2 Imágenes de la retina

La práctica ha demostrado que los resultados obtenidos por los modelos desarrollados dependen de la calidad de los datos utilizados para realizar el entrenamiento y su posterior validación. En este sentido, según Khan et al. (2021) y Mateen et al. (2020) la disponibilidad de datos, en este caso de imágenes, ha resultado esencial en el desarrollo acelerado de las investigaciones.

Las imágenes de la retina son fotografías de su superficie, los vasos sanguíneos, el disco óptico y la mácula (Dutta et al., 2017) que proveen información útil para permitir la identificación y el diagnóstico de diferentes enfermedades (Salahuddin y Qidwai, 2020). Estas imágenes pueden ser capturadas por diferentes especialistas bajo numerosas configuraciones y se almacenan por las instituciones médicas o centros de investigación.

Con el objetivo de facilitar las investigaciones médicas y la integración con las nuevas tecnologías, se han desarrollado diferentes repositorios públicos o privados que bajo diferentes condiciones brindan la oportunidad de evaluar los desarrollos en el área y comparar de forma objetiva los resultados obtenidos. Estos repositorios son mantenidos de forma independiente por diversas instituciones, laboratorios u organizaciones, lo que implica que las imágenes capturadas reúnan diferentes características que pueden dificultar las labores de detección y posterior clasificación de la enfermedad.

En la Tabla 2se observa un resumen de las principales bases de datos públicas identificadas en la literatura que se encuentran disponibles para el desarrollo y evaluación de modelos de diagnóstico de la RD.

Tabla 2. Principales repositorios públicos para el diagnóstico de la RD

Bases de datos	Características
Messidor-1 (Decencière et al., 2014)	● 1200 imágenes en colores. ● Dimensiones: 1440x960, 2240x1488, 2304x1536 pixeles. ● 800 imágenes fueron capturadas dilatando las pupilas
HRF (Budai et al., 2010)	● 45 imágenes en colores; (15 sanas, 15 con algún grado de RD, 15 con diagnóstico de glaucoma).
IDRiD (Porwal et al., 2018b, 2018a, 2020)	● Se encuentra dividida en tres componentes: segmentación, clasificación y localización. ● 516 imágenes cuentan con su diagnóstico, incluyendo la etapa proliferativa.
EyePACS	● 88702 imágenes en colores. ● Incluye el diagnóstico de la etapa proliferativa de la enfermedad.
Messidor-2 (Abràmoff et al., 2013; Decencière et al., 2014)	● 1748 imágenes en colores. ● Las imágenes se adquirieron sin dilatar las pupilas
APTOS	● 3648 imágenes en colores. ● Se encuentra disponible el diagnóstico para cada una de las imágenes, incluyendo la etapa proliferativa.

3 Preprocesamiento digital de las imágenes

Las imágenes disponibles en los repositorios públicos señalados en la Tabla 2 están conformadas por una región de interés que corresponde al tejido de la retina e incluye a la red vascular, la mácula, el disco óptico, entre otras estructuras. Esta región, generalmente, se encuentra rodeada por un área que no aporta información útil y que dificulta las labores de procesamiento de las imágenes, como se observa en la Figura 1a. Por estas razones resulta necesario delimitar el área de interés. Para ello se realiza una transformación del espacio de color al modelo HSV, para separar los valores de croma de la iluminación durante el procesamiento, como se observa en la Figura 1 b-d.

Al analizar las imágenes resultantes, se evidencia que en el canal de "Brillo" (ver Figura 1d) proporciona información relevante para separar la región de interés, particularmente aquellas regiones con un valor de brillo inferior a 0.1 para una imagen con rango dinámico normalizado en el intervalo 0-1.

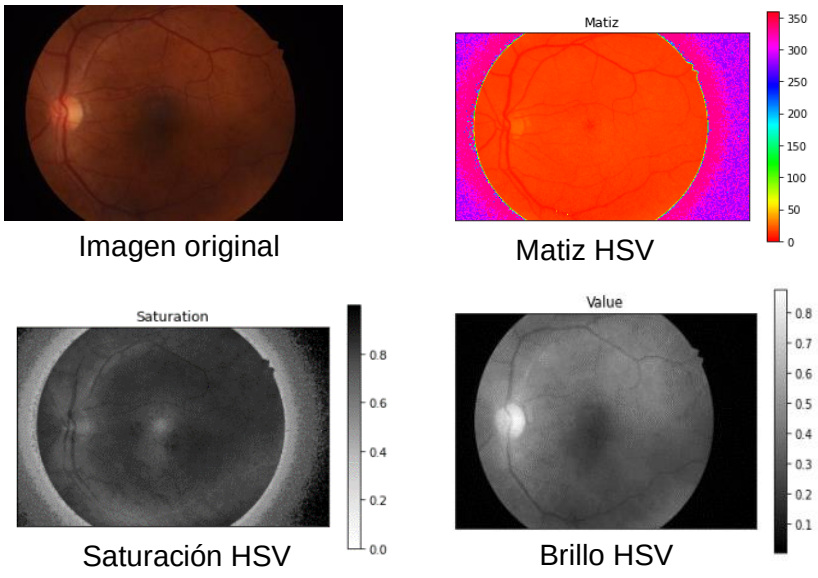


Fig. 1. Imágenes de la retina en el espacio HSV.

Al calcular el histograma de este canal, como se observa en la Figura 2, se evidencia un comportamiento bimodal, la distribución de píxeles que corresponden al fondo, contenido no útil, y los píxeles que brindan la información de interés, se encuentran separados por un valle fácilmente identificable que permite proponer alguno de los métodos de umbralado dinámico conocidos.

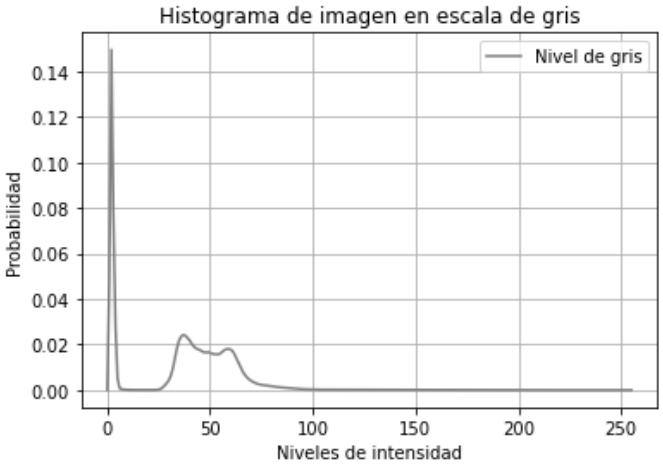


Fig. 2. Histograma del canal Brillo HSV.

A partir de esto, se procede a binarizar la imagen utilizando el método de Otsu (Otsu, 1979), Figura 3b, para con posterioridad determinar la caja envolvente que incluye a la región de interés experimental. Los límites que fijan la ubicación de la caja envolvente fueron calculados determinando, en los cuatro puntos cardinales, las ubicaciones en las que existían transiciones de fondo a contenido útil. Una vez determinada estas ubicaciones, se procede a modificar las dimensiones de la imagen original y seleccionar únicamente a la región de interés, Figura 3c.

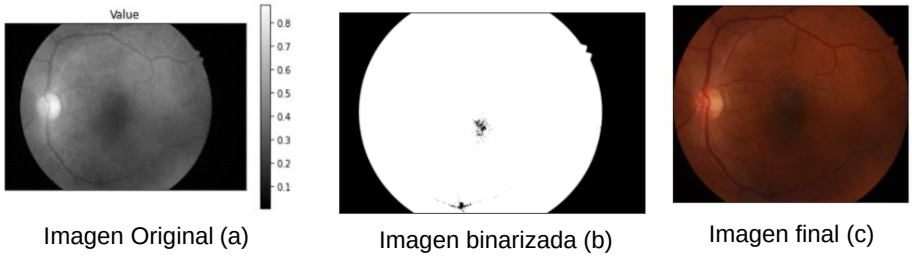


Fig. 3. Determinación de la región de interés experimental.

3.1. Análisis y mejora del contraste en las imágenes

Una vez se ha determinado la región de interés, se procede a analizar la imagen resultante para aplicar alguna técnica de procesamiento que garantice el realce de las estructuras: red vascular, lesiones amarillas, lesiones rojas, entre otras; del resto de los elementos de la imagen para de esta forma facilitar las labores de detección de la enfermedad y su posterior clasificación.

El análisis de la imagen recortada (ver Figura 4a) indica un bajo contraste que dificulta las labores de discriminación de los elementos relevantes de la retina. Esto se aprecia con mayor claridad en su histograma (ver Figura 4b), el cual demuestra que una parte considerable de los píxeles de la imagen corresponden a los valores inferiores del rango dinámico. Esto motiva la necesidad de aplicar alguna técnica que mejore esta distribución.

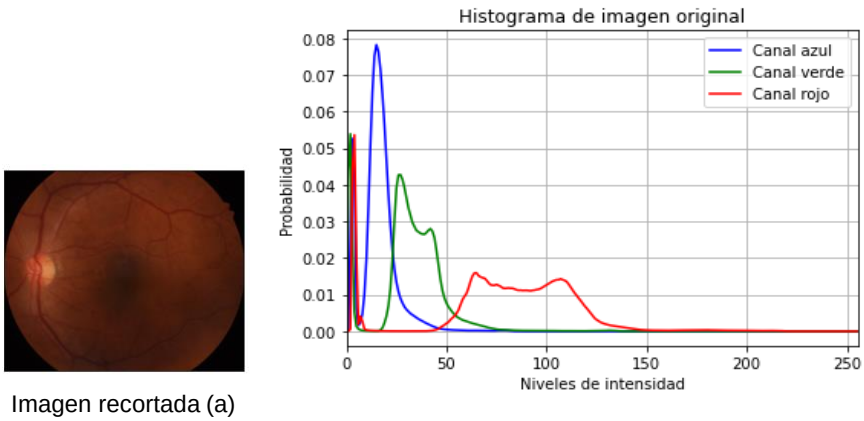


Imagen recortada (a)

Histograma de a) (b)

Fig. 4. Imagen antes del preprocesamiento.

El modelo de color RGB es un modelo aditivo que representa las imágenes digitales en tres canales. Cada canal aporta información diferente y complementaria. En este caso el canal verde aporta la mayor cantidad de información útil. En este canal, el contraste entre las diferentes estructuras de la imagen y su fondo es mayor, facilitando las labores de segmentación y extracción de características. De manera similar, el canal rojo puede ser utilizado con estos objetivos, aunque se aprecia una disminución del contraste entre las estructuras.

El canal azul, por otra parte, no aporta información relevante en relación con la morfología de las diferentes estructuras y es el responsable, en gran medida, del ruido existente en la imagen. A partir de este análisis, resulta deseable que los procesos de "mejoramiento" maximicen el contraste en los canales verde y rojo, al mismo tiempo que disminuyan los efectos del ruido presente en el canal azul.

Con esta meta, a la imagen recortada, se le aplica un filtro gaussiano con una desviación estándar de 1 y un tamaño del núcleo de $3x3$para disminuir el ruido presente y suavizar la imagen; preparándola para el resto de los procesos. Seguidamente se realiza una transformación del espacio de color y se representa la imagen analizada en el modelo LAB aplicando una Ecualización Adaptativa de Histograma Limitado por Contraste (EAHLC) sobre el canal de Luminancia (L), de manera que se mejore el contraste, pero se respeten los valores de croma originales.

En la Figura 5a, se puede observar el resultado de aplicar la metodología descrita. Como se aprecia, ha ocurrido un realce del contraste entre los diferentes elementos de interés de la retina y el fondo que presenta poca información útil; de esta manera, resultan fácilmente identificables la red bascular y las diferentes lesiones que pueden estar presentes. Si analizamos el histograma de la imagen de salida, Figura 5b, se observa cómo se han distribuido de forma más regular en el rango dinámico los pixeles de la imagen, lo que se traduce en un aumento del contraste, tal como se había señalado.

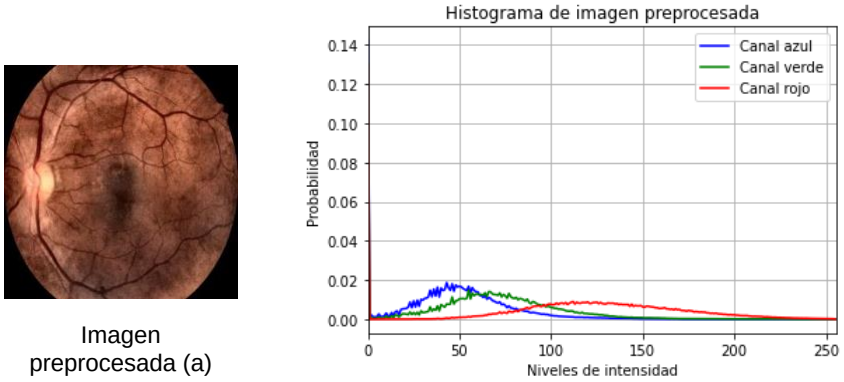


Imagen preprocesada (a)

Histograma de a) (b)

Fig. 5. Resultados del preprocesamiento.

4 Discusión

Durante la extracción de la región de interés, la presencia de pixeles con valores inadecuados derivados del ruido, pueden ser interpretados falsamente como transiciones fondo-región de interés. Esto repercute en una extracción incorrecta del tejido de la retina. Para mitigar este efecto, se suaviza la imagen de entrada aplicando el filtro gaussiano y se considera una determinada longitud en pixeles para determinar la presencia de una transición efectiva, mitigando de esta forma la problemática.

Para algunas imágenes disponibles en los repositorios: imágenes saturadas, con presencia de artefactos y con muy bajo contraste, el método no resulta efectivo porque no es posible determinar las regiones de interés o se determinan de forma errónea y la mejora de contraste no repercute de forma significativa en estas imágenes. Por estas razones, es necesario el desarrollo de algoritmos complementarios que permitan descartar este tipo de imágenes no útiles y que pueden ser clasificadas como no graduables, porque la información que aportan no permite determinar el grado de la enfermedad; de esta manera se mejora el desempeño de los procesos propuestos.

5 Conclusiones

En este trabajo se presentan resultados de un método de preprocesamiento que garantiza la homogeneización de las imágenes disponibles y una mejora de su contraste, de manera que se faciliten las labores de segmentación y extracción de características que garanticen los procesos de diagnóstico y clasificación. Si bien la metodología propuesta, garantiza las condiciones necesarias para procesar las imágenes de la retina, resulta oportuno señalar que no es posible su generalización a otro tipo de problemas.

Para el análisis de imágenes, cada fase del preprocesamiento depende de las características necesarias o deseables para las etapas posteriores. Además, como resultado de la propia composición de los repositorios, pueden existir imágenes no graduables porque se presenten saturadas o con artefactos que imposibiliten realizar los procesos de diagnóstico, en este sentido, resulta necesario considerar el desarrollo de un sistema complementario que permita detectar y descartar este tipo de imágenes.

Referencias

1. Abdel Maksoud, E., Barakat, S., y Elmogy, M. (2020). "*A comprehensive diagnosis system for early signs and different diabetic retinopathy grades using fundus retinal images based on pathological changes detection*". Computers in Biology and Medicine, vol. 126, pp. 104039.
2. Abràmoff, M. D., Folk, J. C., Han, D. P., Walker, J. D., Williams, D. F., Russell, S. R., Massin, P., Cochener, B., Gain, P., Tang, L., Lamard, M., Moga, D. C., Quellec, G., y Niemeijer, M. (2013). "*Automated Analysis of Retinal Images for Detection of Referable Diabetic Retinopathy*". JAMA Ophthalmology, vol. 131, pp. 351–357.
3. Budai, A., Michelson, G., y Hornegger, J. (2010). "*Multiscale blood vessel segmentation in retinal fundus images*". CEUR Workshop Proceedings.
4. Butt, M. M., Latif, G., Iskandar, D. N. F. A., Alghazo, J., y Khan, A. H. (2019). "*Multi-channel Convolutions Neural Network Based Diabetic Retinopathy Detection from Fundus Images*". Procedia Computer Science, vol. 163, pp. 283–291.
5. Das, S., Kharbanda, K., M, S., Raman, R., y D, E. D. (2021). "*Deep learning architecture based on segmented fundus image features for classification of diabetic retinopathy*". Biomedical Signal Processing and Control, vol. 68, pp. 102600.
6. Decencière, E., Zhang, X., Cazuguel, G., Lay, B., Cochener, B., Trone, C., Gain, P., Ordonez, R., Massin, P., Erginay, A., Charton, B., y Klein, J.-C. (2014). "*Feedback on a publicly distributed image database: The Messidor Database*". Image Analysis & Stereology, vol. 33, pp. 231.
7. Dutta, M. K., Parthasarathi, M., Ganguly, S., Ganguly, S., y Srivastava, K. (2017). "*An efficient image processing based technique for comprehensive detection and grading of nonproliferative diabetic retinopathy from fundus images*". Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization, vol. 5, pp. 195–207.
8. Hajabdollahi, M., Esfandiarpoor, R., Najarian, K., Karimi, N., Samavi, S., y Reza Soroushmehr, S. M. (2019). "*Hierarchical Pruning for Simplification of Convolutional Neural Networks in Diabetic Retinopathy Classification*". 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 970–973.
9. Hsieh, Y.-T., Chuang, L.-M., Jiang, Y.-D., Chang, T.-J., Yang, C.-M., Yang, C.-H., Chan, L.-W., Kao, T.-Y., Chen, T.-C., Lin, H.-C., Tsai, C.-H., y Chen, M. (2021). "*Application of deep learning image assessment software VeriSeeTM for diabetic retinopathy screening*". Journal of the Formosan Medical Association, vol. 120, pp. 165–171.
10. Khan, S. M., Liu, X., Nath, S., Korot, E., Faes, L., Wagner, S. K., Keane, P. A., Sebire, N. J., Burton, M. J., y Denniston, A. K. (2021). "*A global review of publicly available datasets for ophthalmological imaging: barriers to access, usability, and generalisability*". The Lancet Digital Health, vol. 3, pp. e51–e66.

11. Mateen, M., Wen, J., Hassan, M., Nasrullah, N., Sun, S., y Hayat, S. (2020). "*Automatic Detection of Diabetic Retinopathy: A Review on Datasets, Methods and Evaluation Metrics*". IEEE Access, vol. 8, pp. 48784–48811.
12. Otsu, N. (1979). "*A Threshold Selection Method from Gray-Level Histograms*". IEEE Transactions on Systems, Man, and Cybernetics, vol. 9, pp. 62–66.
13. Porwal, P., Pachade, S., Kamble, R., Kokare, M., Deshmukh, G., Sahasrabuddhe, V., y Meriaudeau, F. (2018a). "*Indian Diabetic Retinopathy Image Dataset (IDRiD): A Database for Diabetic Retinopathy Screening Research*". Data, vol. 3, pp. 25.
14. Porwal, P., Pachade, S., Kamble, R., Kokare, M., Deshmukh, G., Sahasrabuddhe, V., y Meriaudeau, F. (2018b). "*Indian Diabetic Retinopathy Image Dataset (IDRiD)*". IEEE Dataport.
15. Porwal, P., Pachade, S., Kokare, M., y Mériaudeau, F. (2020). "*IDRiD: Diabetic Retinopathy – Segmentation and Grading Challenge*". Medical Image Analysis, vol. 59, pp. 101561.
16. Salahuddin, T., y Qidwai, U. (2020). "*Computational methods for automated analysis of corneal nerve images: Lessons learned from retinal fundus image analysis*". Computers in Biology and Medicine, vol. 119, pp. 103666.
17. Samanta, A., Saha, A., Satapathy, S. C., Fernandes, S. L., y Zhang, Y.-D. (2020). "*Automated detection of diabetic retinopathy using convolutional neural networks on a small dataset*". Pattern Recognition Letters, vol. 135, pp. 293–298.
18. Saranya, P., y Prabakaran, S. (2020). "*Automatic detection of non-proliferative diabetic retinopathy in retinal fundus images using convolution neural network*". Journal of Ambient Intelligence and Humanized Computing.
19. Zhou, L., Zhao, Y., Yang, J., Yu, Q., y Xu, X. (2018). "*Deep multiple instance learning for automatic detection of diabetic retinopathy in retinal images*". IET Image Processing, vol. 12, pp. 563–571.

Capítulo 5. Autenticación de personas mediante la extracción de rasgos faciales

Aida Anai Aparicio-Arroyo, Ivan Olmos-Pineda, José Arturo Olvera-López
Doctorado en Ingeniería del Lenguaje y del Conocimiento, Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla, Puebla, México
e-mail autor por correspondencia. aida.aparicio@alumno.buap.mx

Resumen. En este trabajo, se presenta una estrategia para la autenticación de personas, analizando cada una de las etapas que conforman este proceso: detección de rostros, preprocesamiento, extracción de rasgos faciales y etapa de clasificación. De igual manera, se exponen los resultados visuales de las primeras etapas y los resultados numéricos de la etapa de clasificación, en la cual se utiliza una Máquina de Vectores de Soporte (SVM, por sus siglas en inglés) con diferentes kernels. Por último, se hace un análisis de los resultados obtenidos y el posible trabajo a futuro.

Palabras Clave: Extracción de Rasgos Faciales, Máquinas de Vectores de Soporte, Autenticación de Personas.

1 Introducción

Actualmente, la autenticación de personas ha tomado mucho auge, ya que existen diferentes sistemas que son capaces de realizar la tarea con éxito. Algunos ejemplos los podemos observar en: bancos (cuando se va a realizar algún trámite y es necesario verificar que la persona presente es la misma que abrió la cuenta), aplicaciones móviles (para acceder a ciertas aplicaciones, se necesita verificar a través de correos electrónicos o contraseñas), teléfonos móviles (como parte de su sistema de seguridad cuentan con el reconocimiento facial), aeropuertos (en la zona de embarque, se está empleando el reconocimiento facial como sistema checador en el abordaje), entre otros.

En los ejemplos anteriormente mencionados, los sistemas de autenticación utilizan diferentes características como huella dactilar, comandos de voz, palabras claves o rasgos faciales. Pero en los últimos años, utilizar los rasgos faciales para el proceso de autenticación de personas ha incrementado y por tal motivo, este trabajo se enfocará en los rasgos faciales.

Queda claro que, para autenticar a una persona, el proceso debe de estar constituido por diferentes etapas como: detección de rostros, preprocesamiento, extracción de rasgos faciales, clasificación y, finalmente, la etapa de pruebas para tener como resultado la autenticación.

2 Trabajos relacionados

Dentro del estado del arte, existen diferentes trabajos relacionados con la autenticación de personas. Estos implementan diferentes técnicas para la etapa de detección de rostros, el preprocesamiento y la etapa de extracción de rasgos faciales, esto con el fin de respaldar el proceso de autenticación. Las técnicas más aplicadas para la extracción de rasgos faciales son: Principal Components Analysis (PCA), Local Binary Patterns (LBP), Local Discriminant Analysis (LDA), entre otras.

Mientras que, los clasificadores más utilizados en la literatura son: Máquinas de vectores de soporte (SVM, por sus siglas en inglés), Redes neuronales (NN, por sus siglas en inglés), K-Vecinos más cercanos (K-NN, por sus siglas en inglés) y AdaBoost (Adaptive Boost). Enfocándonos en la etapa de extracción de rasgos faciales y la etapa de clasificación, a continuación, se analizan algunos trabajos relacionados.

Con respecto a las diferentes técnicas para la extracción de rasgos faciales, dentro de las más usadas se encuentra la de PCA como se reporta en los trabajos de (Fujisawa et a., 2021) y (Uddin et al., 2021).Estos autores, utilizan esta técnica con el objetivo de extraer los rasgos faciales más relevantes del rostro y después de aplicar técnicas de preprocesamiento digital en conjunto con un clasificador, el algoritmo resultante pueda ser implementado en un sistema de vigilancia y así reconocer personas o incluso poder rastrearlas dentro de un video.

Otra de las técnicas que se utiliza con frecuencia para la extracción de rasgos faciales es LBP como se muestra en Barburiceanu et al. (2021) y Cheng et al. (2021). Esta técnica se usa para el reconocimiento de expresiones faciales y también para el reconocimiento facial, ya que proporciona un buen análisis de textura. Esta técnica analiza las imágenes de manera local, es decir, se divide la imagen en pequeñas regiones y a cada región se le analiza su textura.

Otro trabajo en el que analizan la textura es el que presentan Liu et al. (2017). Aquí utilizan la técnica LBP, al cual se obtiene el histograma. Los valores del histograma son introducidos a un clasificador de vecinos más cercanos (k-NN) y se reporta que se obtuvo un 95.94% de exactitud al momento del reconocimiento facial.

En el trabajo presentado por Lekdioui et al. (2017) se hace un análisis de la imagen de manera global, se analiza la textura aplicando las técnicas de LBP, CLBP y LTP. Luego, se obtiene el HOG (Histograma de Gradientes Orientados) de los canales RGB. Los HOG y los valores obtenidos de las técnicas de análisis de textura forman en conjunto un vector, el cual se introduce a un clasificador SVM. Los autores reportan una exactitud de reconocimiento de 93.34 %.

Por otra parte, hay artículos que miden otros rasgos del rostro basándose en la simetría, como puede ser el contorno del rostro o medir el ancho o alto de la cara, entre otras cosas. Jiang& Chen. (2021) y Wei et al. (2021) son ejemplos de trabajos en los que determinan los puntos de referencia para obtener métricas del rostro. Estos algoritmos en su mayoría están basados en Modelos de Apariencia Activa (AAM, por sus siglas en inglés), los cuales detectan alrededor de 68 o más puntos de referencia en todo el rostro.

3 Metodología propuesta

Revisado el estado del arte, se concluye que el proceso general de la autenticación de personas está conformado por una serie de etapas. En esta sección, se explicarán a grandes rasgos cada una de estas etapas, al igual que, se mostrarán los resultados visuales de cada una de ellas, concluyendo con el enlistado de los diferentes rasgos faciales que se extraerán y serán utilizados en la etapa de clasificación. En la Figura 1, se muestra el diagrama general del proceso de autenticación.

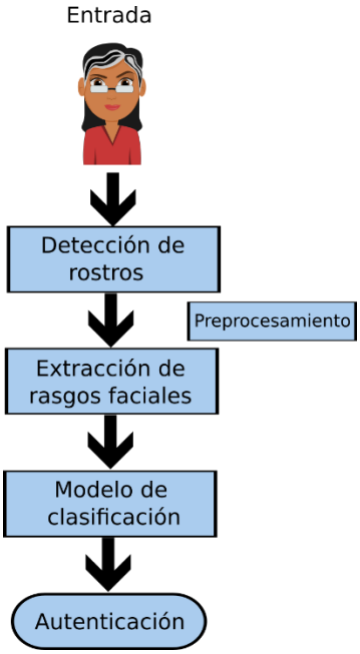


Fig. 1. Diagrama general del proceso de autenticación de personas.

El proceso comienza con una imagen de entrada, en la cual se detecta dónde hay un rostro o diferentes rostros de personas. Ya detectados los rostros, prosigue una etapa de preprocesamiento, en la mayoría de los trabajos relacionados, esta etapa está constituida por correcciones a las imágenes, tales como: corrección de iluminación, ajuste de contraste, corrección de pose, escalamiento, entre otras. Esta etapa se realiza, con el fin de tener las imágenes de los rostros bajos las mismas condiciones. Posteriormente, sigue la etapa de extracción de rasgos faciales, el conjunto de rasgos que se extraen debe de abarcar diferentes aspectos del rostro. En este trabajo de investigación, se propone extraer rasgos faciales que incluyan los basados en color, textura y simetría (en la sección 4, se mencionan los diferentes rasgos faciales que se extraen). Extraídos los rasgos faciales, son almacenados en un descriptor, el cual se utilizará para entrenar un clasificador y obtener como resultado un modelo. Este modelo se utilizará en la etapa de autenticación de personas.

Ya analizadas las diferentes etapas que conforman el proceso general de la autenticación de personas, a continuación, se presentan las diferentes técnicas utilizadas para cada una de estas etapas, al igual que, los resultados visuales obtenidos.

Para la etapa de experimentación, en este trabajo se recolectó un conjunto de imágenes de 10 personas (5 mujeres y 5 hombres). Por cada persona, se obtuvieron 30 imágenes, teniendo como total de la base de datos 300 imágenes. Cabe mencionar que, cada una de las imágenes fueron tomadas por cada una de las 10 personas que conforman la base de datos, por lo tanto, fueron tomadas a diferentes distancias y en diferentes circunstancias (lugares abiertos, lugares cerrados, mucha iluminación, poca iluminación, etc.). En la Figura 2, se muestran algunos ejemplos de la base de datos generada.

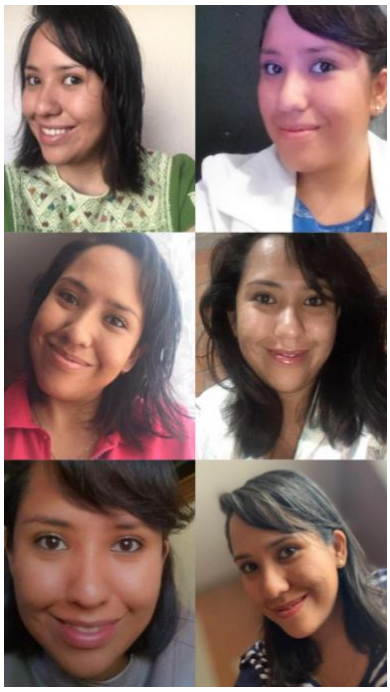


Fig. 2. Ejemplos de la base de datos generada.

La primera etapa del proceso de autenticación de personas es la detección de rostros. Para este proceso, se utilizó una Red Neuronal Convolucional ya entrenada (CNN, por sus siglas en inglés). Esta CNN fue entrenada con la base de datos COCO (Common Objects in Context), la cual contiene 91 clases y más de 300000 imágenes (Lin et al., 2014)., esta CNN puede detectar personas, perros, cosas, entre otras. En la Figura 3, se muestra el diagrama general de la CNN que se utiliza para esta etapa de detección de rostros. Como entrada se tiene una imagen que será procesada por la CNN ya entrenada, se construye un blob, el cual contiene las coordenadas de donde se encuentra el objeto dentro de la imagen. A partir de ese blob, se crea una máscara, que será utilizada para la segmentación del objeto y como resultado final, se visualiza la región de interés (ROI, por sus siglas en inglés) y la segmentación de la ROI.

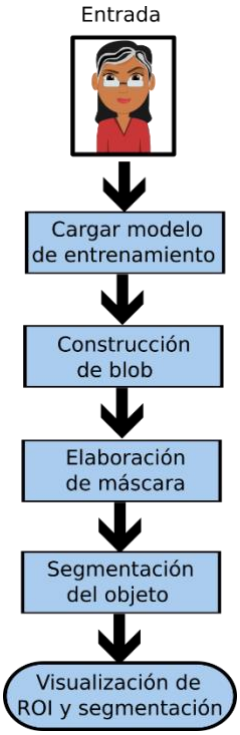


Fig. 3. Diagrama general del proceso de detección de rostros.

Mientras que, en la Figura 4 se muestran algunos de los resultados obtenidos de la detección de rostros, en la columna de la izquierda, se muestran las imágenes originales, mientras que, en la columna de la derecha, se muestra a la persona detectada (por ende, el rostro detectado).

Fig. 4. Ejemplos de la etapa de detección de rostros. Imagen original (columna izquierda) y rostro detectado (columna derecha).

Después de haber detectado el rostro, se prosigue con la etapa de preprocesamiento. En esta etapa, se realiza una serie de tareas, con el objetivo de tener todas las imágenes con las mismas condiciones. Para este objetivo, se realiza una corrección de pose y escalamiento, donde se utiliza el algoritmo de Landmarks, el cual consiste en utilizar una plantilla que detecta 68 punto de referencias, estos puntos abarcan diferentes zonas de interés como: cejas, ojos, nariz, boca y mentón.

Ya detectadas las zonas de interés, se genera una matriz afín, la cual servirá para poder rotar, trasladar y escalar una imagen en la posición y tamaño deseado. Cabe aclarar que, el escalado es el mismo para todos los rostros, para que todas las imágenes estén en las mismas condiciones. En la Figura 5, se muestran algunos ejemplos de esta etapa de preprocesamiento, en la columna de la izquierda se muestran las imágenes originales, mientras que, en la columna de la derecha, se muestran las correcciones de pose y escalamiento.

Fig. 5. Ejemplos de la etapa de preprocesamiento. Imagen original (columna de la izquierda), rostro corregido y escalado (columna de la derecha).

Con los rostros corregidos y escalados, se debe de realizar un último preproceso, el cual es segmentar el rostro, esto se hace con el objetivo de analizar las ROIs, ya que habrá regiones del rostro que no aportan mucha información. Para este proceso, se volvió a utilizar el algoritmo de Landmarks, el cual delimita las ROIs, proporcionando las coordenadas en donde se encuentran y donde empiezan y terminan. En la Figura 6, se muestran algunos de los resultados obtenidos de la segmentación de rostros, se muestran las imágenes corregidas (columna de la izquierda) y los rostros segmentados (columna de la derecha).

Fig. 6. Ejemplos de la etapa de segmentación. Rostro completo (columna izquierda) y rostro segmentado (columna derecha).

La etapa que sigue es la extracción de rasgos faciales, con los rostros ya segmentados, se extraen los rasgos faciales que se presentan en la tabla 1.

Tabla 1. Rasgos faciales extraídos de cada una de las diferentes categorías (basadas en color, basadas en textura y basadas en simetría).

Categoría	Rasgos faciales
Rasgos faciales basados en color de piel	Desviación estándar (R,G,B), Varianza (R,G,B), Valor mínimo (H,S,V), Valor máximo (H,S,V), Desviación estándar (H,S,V), Varianza (H,S,V) y Entropía (H,S,V)
Rasgos faciales basados en textura de la piel	Matriz de Co-ocurrencia, Transformada de Fourier y Filtro de Gabor
Rasgos faciales basados en simetría	Ancho del rostro, ancho de los ROIs y distancia entre los ROIs

4 Resultados

Tomando como base el conjunto de rasgos faciales que se presentan en la Tabla 1 de la sección anterior, se prosigue con la etapa de experimentación. El clasificador que es utilizado para esta etapa es una Máquina de Vector de Soporte (SVM). Estos modelos trabajan a través de kernels, los cuales pueden ser: “rbf”, “linear”, “poly”, “sigmoid” y “precomputed”. Para los resultados que se presentan a continuación, se utilizaron los kernels “linear” y “poly-6”. En las siguientes Tablas, se muestran los experimentos que se realizaron (de 2 a 10 personas) y el porcentaje global que se obtuvo para cada uno de los experimentos.

Cabe mencionar que, antes de comenzar con la etapa de clasificación, la base de datos es estratificada, esto con el objetivo de que por lo menos una clase (persona) esté en el submuestreo y, además, se hace una validación cruzada (cv, por sus siglas en inglés) como métrica de evaluación.

Tabla 2. Resultados de la etapa de clasificación, utilizando un clasificador SVM, una cv=5 y el kernel linear.

Experimento	Precisión global
2 personas	100%
4 personas	94.27%
6 personas	90.07%
8 personas	75.81%
10 personas	73.92%

Como se puede observar, con forme se va aumentando el número de personas (de 2 a 10), el porcentaje global va disminuyendo, esto debido a que habrá imágenes de una persona que se pueden parecer a otra persona y esto provoca que el clasificador se confunda al momento de la clasificación.

Por tal motivo, para últimos experimentos, se realiza una modificación en la cv pasando de 5 a 10, al igual que, se cambia el kernel que se utilizade linear a poly-6. En la Tabla 3, se muestran los resultados obtenidos después de hacer las modificaciones mencionadas.

Tabla 3. Resultados de la etapa de clasificación, utilizando un clasificador SVM, una cv=10 y el kernel poly-6.

Experimento	Precisión global
2 personas	100%
4 personas	97.82%
6 personas	96.12%
8 personas	98.30%
10 personas	99.11%

5 Conclusiones

Como parte del análisis de los resultados obtenidos, se puede notar que, cuando va aumentando el número de personas el porcentaje de precisión global va disminuyendo, como se mencionó previamente, una de las razones es debido a que, habrá imágenes parecidas entre varias personas y es por eso por lo que el modelo entrenado se puede confundir al momento de la clasificación.

Por tal motivo, se hizo una serie de modificaciones al algoritmo, con el objetivo de enriquecer la parte de la clasificación. Se puede observar que, cuando se aumentó el número de la validación cruzada y se hizo el cambio de kernel, los porcentajes de precisión global aumentaron considerablemente y, además, los resultados obtenidos pasaron a ser comparables con el estado del arte.

Como trabajo a futuro, se propone realizar experimentos con otras bases de datos, con el fin de probar el algoritmo y medir el porcentaje de precisión. Al igual que, aumentar el número de personas con el que se realizaron los experimentos en este artículo.

Referencias

1. Barburiceanu, S., Terebes, R., & Meza, S. (2021). *3D texture feature extraction and classification using GLCM and LBP-based descriptors*. Applied Sciences, 11(5), 2332.
2. Cheng, J., Xu, Y., & Kong, L. (2021). *Hyperspectral imaging classification based on LBP feature extraction and multimodel ensemble learning*. Computers & Electrical Engineering, 92, 107199.
3. Fujisawa, K., Shimo, M., Taguchi, Y. H., Ikematsu, S., & Miyata, R. (2021). *PCA-based unsupervised feature extraction for gene expression analysis of COVID-19 patients*. Scientific reports, 11(1), 1-11.
4. Jiang, M., & Chen, Z. (2021). *Symmetry detection algorithm to classify the tea grades using artificial intelligence*. Microprocessors and Microsystems, 81, 103738.
5. Lekdioui, K., Messoussi, R., Ruichek, Y., Chaabi, Y., & Touahni, R. (2017). *Facial de composition for expression recognition using texture/shape descriptors and svm classifier*. Signal Processing: Image Communication, 58, 300–312.
6. Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick,C. L. (2014). *Microsoft coco: Common objects in context*. In European conference oncomputer vision, (pp. 740–755). Springer.
7. Liu, Z., Wu, M., Cao, W., Chen, L., Xu, J., Zhang, R., Zhou, M., & Mao, J. (2017). *A facial expression emotion recognition based human-robot interaction system*. IEEE/CAA Journalof Automatica Sinica, 4(4), 668–676.
8. Uddin, M. P., Mamun, M. A., & Hossain, M. A. (2021). *PCA-based feature reduction for hyperspectral remote sensing image classification*. IETE Technical Review, 38(4), 377-396.
9. Wei, W., Ho, E. S., McCay, K. D., Damaševičius, R., Maskeliūnas, R., & Esposito, A. (2021). *Assessing facial symmetry and attractiveness using augmented reality*. Pattern Analysis and Applications, 1-17.

Capítulo 6. Un modelo de recomendación basado en recuperación de información para textos turísticos mexicanos en español

Victor Giovanni Morales Murillo[1], David Eduardo Pinto Avendaño[1], Franco Rojas López[2]
[1]Benemérita Universidad Autónoma de Puebla. Facultad de Ciencias de la Computación
[2]Universidad Politécnica Metropolitana de Puebla. Ingeniería en Sistemas Computacionales
e-mail autor por correspondencia. vg055@hotmail.com

Resumen. El turismo es una de las principales actividades económicas en el mundo, en México esta actividad representa el 8.7% del PIB y genera aproximadamente 4.5 millones de empleos directos. En el presente trabajo se generó un modelo de recomendación basado en recuperación de información para textos turísticos mexicanos en español para la tarea de sistema de recomendación del Rest-Mex 2022, esta tarea consiste en dado un turista de TripAdvisor y un lugar turístico mexicano se debe estimar automáticamente el grado de satisfacción (entre 1 a 5) que tendrá el turista al visitar el lugar. El modelo de recomendación se evaluó con la métrica MAE por cada clase (grado de satisfacción) y de forma general obtuvo un MAE de 0.74 y, finalmente estos resultados se compararon con los del Rest-Mex 2021.

Palabras Clave: Sistema de recomendación, recuperación de información, turismo, procesamiento del lenguaje natural.

1 Introducción

"El turismo es un fenómeno social, cultural y económico relacionado con el movimiento de las personas a lugares que se encuentran fuera de su lugar de residencia habitual, normalmente por motivos de ocio" (INEGI, 2022). Además, el turismo es una de las principales actividades económicas en el mundo, en México esta actividad representa el 8.7% del producto interno bruto (PIB) y genera aproximadamente 4.5 millones de empleos directos, sin embargo, la pandemia generada por el virus SARS-COV-2 en marzo del 2020 está perjudicando a este sector gravemente, por lo tanto, los servicios y productos turísticos deben mejorar su calidad y seguridad para restablecer el turismo nacional (Álvarez et al., 2021).

Por lo anteriormente expuesto, se organiza el evento *Recommendation System, Sentiment Analysis and Covid Semaphore Prediction for Mexican Tourist Texts* (Rest-Mex) que está formado por tres sub tareas para textos turísticos mexicanos: (1) sistema de recomendación (SR), (2) análisis de sentimientos y (3) predicción del semáforo COVID (Rest-Mex, 2022). Este evento fomenta la aplicación del procesamiento del lenguaje natural (PLN) que es un área de la inteligencia artificial que fortalecerá el turismo nacional. Además, PLN puede generar mecanismos que identifiquen problemas con base en las polaridades de las opiniones de los turistas, mecanismos que generen

recomendaciones precisas sobre lugares turísticos a visitar para los turistas y mecanismos que predigan el color del semáforo epidemiológico con base en noticias de la enfermedad COVID.

En el presente trabajo se propone un modelo de recomendación basado en recuperación de información para participar en la tarea de sistema de recomendación del Rest-Mex 2022, esta tarea consiste en dado un turista de TripAdvisor y un lugar turístico mexicano, se debe predecir automáticamente el grado de satisfacción (entre 1 a 5) que tendrá el turista al visitar el lugar. La tarea surge en el Rest-Mex porque un número limitado de sistemas de recomendación turísticos se basan en la relación de un perfil del usuario con la descripción de un lugar, los conjuntos de datos generalmente son en el idioma inglés y es fundamental generar recursos en español para desarrollar sistemas inteligentes para el turismo porque los países iberoamericanos son fundamentales para esta actividad (Rest-Mex, 2022).

Un sistema de recomendación es una herramienta de software que realiza sugerencias de productos llamados ítems como son sitios web, amigos, empleos, películas, productos comerciales, canciones, lugares turísticos, hoteles, restaurantes y otros artículos relevantes con base en las preferencias de los usuarios (Çano & Morisio, 2017). Actualmente, estos sistemas representan un alto impacto económico, social y tecnológico a nivel internacional porque múltiples compañías mundiales de diversos giros como Google, Amazon, Netflix, Spotify, Facebook y muchas más han utilizado los sistemas de recomendación dentro de sus principales servicios (Jannach et al., 2016). Además, estos sistemas benefician económicamente a estas compañías porque abordan el problema de la sobrecarga de información en el comercio electrónico y mejoran la experiencia de los usuarios con la ayuda en su toma de decisiones (Malekpour Alamdari et al., 2020).

La recuperación de información (RI) consiste en buscar documentos de texto que satisfacen una necesidad de información en grandes colecciones de documentos (Manning et al., 2009). Las técnicas de RI han sido utilizadas ampliamente con la técnica de recomendación del filtrado basado en contenido que consiste en sugerir ítems similares (Kaššák et al., 2016), en donde se genera un perfil del usuario con los atributos y las descripciones textuales de los ítems previamente evaluados por el usuario y se buscan los ítem más similares al perfil del usuario con base en los metadatos textuales de los ítems (Idrissi et al., 2019). A pesar de que las técnicas de RI se han utilizado tradicionalmente en el filtrado basado en contenido, también es posible utilizarlas en otra técnica de recomendación que es el filtrado colaborativo que consiste en buscar usuarios (filtrado colaborativo basado en el usuario) o ítems (filtrado colaborativo basado en el ítem) similares llamados vecinos cercanos con base en las calificaciones (entre 1 a 5) que han otorgado los usuarios hacia los ítems que se encuentran almacenadas en una matriz de calificaciones que representa a los usuarios como sus filas y a los ítems como sus columnas, con base en los vecinos cercanos identificados se predicen las calificaciones que un usuario otorgaría a un ítem que no ha calificado (Tahmasebi et al., 2021). En este trabajo se utilizan técnicas de RI para aplicar la técnica de recomendación del filtrado colaborativo basado en el ítem.

El resto del artículo está estructurado de la siguiente manera, en la sección 2 se presentan los trabajos más relevantes de los sistemas de recomendación del Rest-Mex, en la sección 3 se presenta la metodología para desarrollar el modelo de recomendación

basado en RI, en la sección 4 se presenta el experimento realizado, así como los resultados obtenidos y, finalmente en la sección 5 se presentan las conclusiones.

2 Trabajos relacionados

En la edición del Rest-Mex 2021 (Álvarez et al., 2021) la tarea de SR consiste en predecir el grado de satisfacción que tendrá un turista al recomendar un destino de Nayarit, México, a partir del historial de los lugares visitados por el turista y las opiniones que se le dan a cada uno de ellos. Se desarrolla una colección de datos con 2,263 instancias de 2,011 usuarios que visitaron 18 lugares turísticos en Nayarit, México. El conjunto de datos se divide en 70/30, es decir 1,582 instancias para entrenamiento y en 681 instancias para pruebas. También, se utiliza la métrica de evaluación llamada en inglés Mean Average Error (MAE). En esa edición del Rest-Mex participaron dos equipos en la tarea de sistema de recomendación el equipo Alumni-MCE 2GEN y el equipo Labsemco-UAEM, los resultados obtenidos por estos dos equipos se presentan en la Tabla 1, en donde se observa que el mejor resultado lo obtuvo el equipo Alumni-MCE 2GEN (Ejecución 1) con un MAE de 0.31, seguido del sistema de recomendación del equipo Baseline que es utilizado por los organizadores del evento con un MAE de 0.73 y finalmente el equipo Labsemco-UAEM con un MAE de 1.65.

Tabla 1. Resultados de la tarea de sistema de recomendación del Rest-Mex 2021.

Equipo	**MAE**
Alumni-MCE 2GEN (Ejecución 1)	0.31
Alumni-MCE 2GEN (Ejecución 2)	0.32
Baseline	0.73
Labsemco-UAEM	1.65

El equipo Alumni-MCE 2GEN participó en el Rest-Mex 2021 con el trabajo titulado en inglés *An Embeddings Based Recommendation System for Mexican Tourism. Submission to the REST-MEX Shared Task at Iber LEF 2021*(Arreola et al., 2021) que propone dos variantes de un mismo modelo, en donde el principal desafío fue la representación de un vector de palabras. En la primera variante, un modelo de Doc2Vec fue aplicado a la información de los usuarios y los lugares que contiene el conjunto de datos, se diseñó una matriz con los embeddings obtenidos y se utilizó una red neuronal con una capa oculta. En la segunda variante, se propone un sistema basado en representaciones distribuidas de textos utilizando el enfoque BERT.

El segundo equipo Labsemco-UAEM participó en el Rest-Mex 2021 con el trabajo titulado en inglés *A Recommendation System for Tourism Based on Semantic Representations and Statistical Relational Learning* (Morales et al., 2021) que propone un método diferente a los métodos de coocurrencia léxica en texto, este método extrae características lingüísticas en el texto, específicamente las señales léxicas y semánticas de la sinonimia y la antonimia. Además, se utiliza un modelo ComplEx para generar las recomendaciones con base en la relación entre un usuario y un lugar.

En ambos trabajos previos se aborda el problema de exactitud porque es el principal objetivo de la tarea sobre sistema de recomendación del Rest-Mext, sin embargo, en la era digital es importante generar motores de recomendación escalables que procesen altos volúmenes de datos y que impacten en el tiempo de ejecución de sus recomendaciones (Nilashi et al., 2018), por tal motivo, en este trabajo se utilizan técnicas de recuperación de información y un índice invertido como estructura de datos para minimizar el costo de cómputo de las recomendaciones.

3 Metodología

En esta sección se describe el conjunto de datos utilizado del Rest-Mex 2022 y las técnicas de recuperación de información y de sistemas de recomendación usadas para desarrollar un modelo de recomendación basado en RI para textos turísticos mexicanos en español. En la Figura 1, se presenta el esquema general de la metodología aplicada que está formado por 4 componentes principales. (1) En el preprocesamiento se representan documentos de texto, de los cuales se eliminan signos de puntuación y se convierten letras mayúsculas en minúsculas. (2) En la indización se utiliza un índice invertido con un diccionario como estructura de datos, se tokenizan los documentos para obtener sus términos y se indizan cada uno de los términos de los documentos usando el esquema de ponderación llamado en inglés term frequency-inverse document frequency (tf-idf). (3) En la recuperación de información se implementa el algoritmo llamado en inglés Cosine Score que utiliza la similitud del coseno y la ponderación tf-idf para buscar los *k* documentos más similares a una consulta representada como documento de texto que es previamente preprocesada. Finalmente, (4) en el sistema de recomendación se aplica la técnica del filtrado colaborativo basado en el ítem a través del algoritmo de RI y se evalúan los resultados con la métrica MAE.

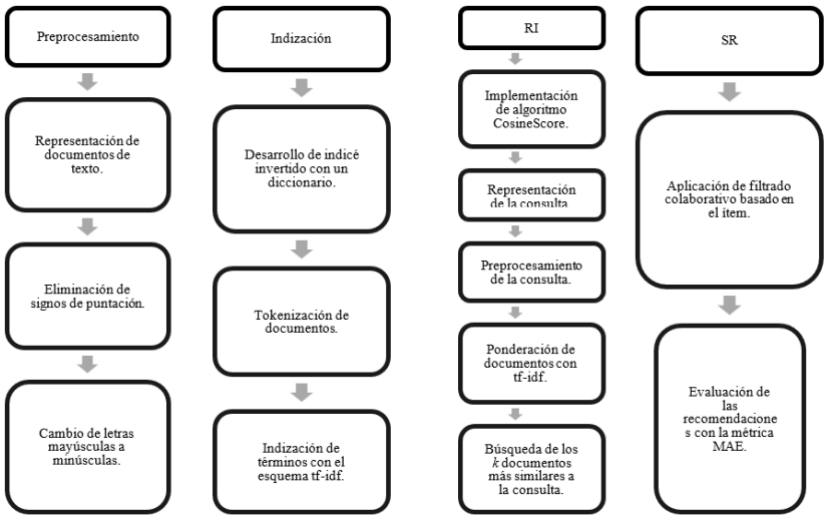


Fig. 1. Esquema general de la metodología aplicada.

3.1 Conjunto de datos

El conjunto de datos del Rest-Mext 2022 contiene 2,263 instancias de 2,011 usuarios que visitaron 18 lugares turísticos en Nayarit, México. Esta colección de datos fue obtenida de turistas que compartieron su grado de satisfacción en TripAdvisor entre 2010 y 2020. El grado de satisfacción representa a cinco clases que están etiquetadas como (1) muy malo, (2) malo, (3) neutral, (4) Bueno y (5) muy bueno. El objetivo de la tarea de sistemas de recomendación del Rest-Mex 2022 es predecir el valor de estas clases que asignaría un turista a un lugar turístico de Nayarit, México. La distribución de estas clases en el conjunto de entrenamiento es la siguiente, la clase 1 tiene 45 instancias, la clase 2 tiene 53 instancias, la clase 3 tiene 167 instancias, la clase 4 tiene 457 instancias y la clase 5 tiene 860 instancias, en total son 1,582 instancias. El número de instancias por cada clase no está balanceado lo que representa un desafío importante para la tarea de SR.

La información que contiene cada instancia se divide en dos partes: (1) la información del usuario que contiene su género, el lugar que el turista recomienda visitar, el lugar de origen del turista, la fecha en que se registró la recomendación, el tipo de viaje [familia, amigos, solo, en pareja, negocios] y el historial de los lugares que el turista a visitado que contiene una reseña textual (opinión del turista) y una calificación (grado de satisfacción) sobre cada uno de esos lugares, y (2) la información del lugar que contiene una breve descripción textual del lugar y una serie de características como el tipo de turismo que se puede realizar (aventura, playa, relajación y otros más), si tiene un ambiente familiar, privado o público, si es gratis o se paga por el acceso y otras características más. En esta investigación se utilizó el conjunto de entrenamiento del Rest-Mex 2022 que contiene 1,582instancias (70%) de las 2,263 instancias totales porque aún no está disponible el conjunto de pruebas. Para el conjunto de pruebas se seleccionaron292 instancias que contienen exclusivamente textos turísticos en español.

3.1.1 Preprocesamiento de conjunto de datos

En esta etapa con Pythonse removieron los usuarios que contengan en su historial de lugares visitados reseñas textuales en otro idioma diferente al español y que no contengan ninguna reseña textual en español registrada, de las 1,582 instancias se seleccionaron 292 instancias para las pruebas que hacen referencia únicamente a textos turísticos en español. Además, en las descripciones textuales de los lugares turísticos y en las reseñas textuales de los usuarios se sustituyeron letras mayúsculas por minúsculas, se removieron acentos y se eliminaron signos de puntuación como los siguientes[,.;:#$!¡?%\n\f=*@\|]. El resultado del preprocesamiento permite tokenizar los documentos del conjunto de datos de una forma más eficiente para seleccionar los términos de cada documento de una forma estandarizada para desarrollar un índice invertido como estructura de datos para el algoritmo de RI.

3.2 Técnicas de recuperación de información

En primera instancia se representaron como documentos de texto los lugares turísticos a recomendar y las reseñas textuales de los turistas hacia los lugares que han visitado en el pasado para utilizar técnicas de recuperación de información. En segunda instancia se genera un modelo espacio vectorial de RI que consiste en representar un conjunto de documentos como vectores que comparten un mismo espacio vectorial y cada término de los documentos representa una dimensión del espacio vectorial (Manning et al., 2009). Este modelo es fundamental para operaciones de RI como son la ponderación de documentos en una consulta, la clasificación de documentos y la agrupación de documentos. También, es importante representar una consulta textual como un vector en el mismo espacio vectorial que comparten los vectores del conjunto de documentos para buscar los vectores más cercanos al vector de la consulta utilizando el ángulo del coseno. La similitud del coseno se denota por la siguiente formula:

$$sim(d_1, d_2) = \frac{\vec{V}(d_1) \cdot \vec{V}(d_2)}{(\vec{V}(d_1)||\vec{V}(d_2))}$$

En donde se calcula la similitud del coseno entre el documento d_1 y el documento d_2, la representación vectorial de estos documentos es a través de $\vec{V}(d_1)$ y $\vec{V}(d_2)$, el numerador de la formula significa el producto punto de los vectores y el denominador significa su distancia euclidiana. Asimismo, se pude calcular la similitud de un vector de una consulta con el vector de un documento para obtener los *k* documentos más similares a la consulta. Se utiliza un esquema de ponderación tf-idf que calcula el peso de cada componente de un documento que representa el término de un vector, asigna un peso al término *t* en el documento d de la siguiente forma: (1) es alto cuando *t* ocurre muchas veces en un bajo número de documentos, (2) es bajo cuando el término ocurre pocas veces en un documento u ocurre en muchos documentos y (3) es más bajo cuando ocurre el término en prácticamente en todos los documentos. La fórmula de tf-idf es la siguiente:

$$tf - idf_{t,d} = tf_{t,d} \times idf_t$$

En donde, la frecuencia del término en un documento denotada por tf y la frecuencia inversa del documento denotada por $idf_{t,d}$ se utilizan con la normalización del logaritmo y del coseno de la siguiente manera.

$$tf_{t,d} = 1 + log\ log\ (tf_{t,d})$$
$$idf_t = log\ (\frac{N}{df_t})$$

En Python, se realiza la implementación del modelo espacio vectorial con un esquema de ponderación tf-idfy la similitud del coseno utilizando el algoritmo llamado en inglés Cosine Score de Manning (Manning et al., 2009).Adicionalmente, se construyó un índice invertido de forma automática por cada usuario, el índice invertido contiene como llaves los términos de los documentos del historial de lugares visitados

del turista y como valores contiene los documentos en que aparecen los términos y la frecuencia de los términos, es fundamental el índice invertido por cada usuario para poder usar el algoritmo de Cosine Score.

3.3 Modelo de recomendación basado en RI

El modelo de recomendación se genera utilizando la técnica del filtrado colaborativo basado en ítem que consiste en buscar ítems similares para predecir la calificación de un ítem que no ha visto un usuario. Esta técnica de recomendación se implementa en Python con las técnicas de RI previamente descritas en la sección anterior. En la Figura 2, se muestra este modelo de recomendación basado en RI, en primera instancia, se preprocesan e indizan los documentos de las reseñas textuales de un turista en un índice invertido para poder utilizar el algoritmo de RI. En segunda instancia, la consulta para el algoritmo de RI es representada por un documento preprocesado de un lugar turístico para recomendar. En tercera instancia, el algoritmo de RI obtiene los *k* reseñas más similares al lugar turístico con *k*=3. En cuarta instancia, se promedian las calificaciones de los lugares turísticos relacionados a las reseñas para estimar el grado de satisfacción del usuario para el lugar turístico de la consulta. Finalmente, se evalúa el valor de la recomendación estimado con la métrica de MAE.

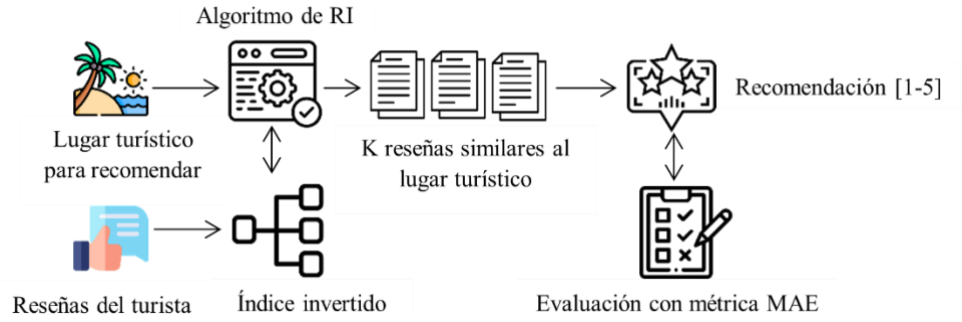


Figura 2. Modelo de recomendación basado en RI (Flaticon, 2022).

4 Experimento y resultados

El experimento se realiza con el conjunto de entrenamiento del Rest-Mex 2022 que contiene 1,582 instancias, de las cuales se seleccionaron 292 instancias para formar el conjunto de pruebas que contiene usuarios con reseñas turísticas exclusivamente en el idioma español. Por cada instancia, se utilizó el modelo de recomendación basado en RI y se evaluó el resultado de la recomendación con la métrica principal del Rest-Mext 2022 que es el MAE. La fórmula para calcular el MAE es la siguiente.

$$MAE = \frac{\sum_{u \in U} \quad \sum_{i \in testset_u} \quad |rec(u,i) - r_{u,i}|}{\sum_{u \in U} \quad |testset_u|}$$

En donde se calcula el promedio de desviación entre la recomendación estimada $rec(u,i)$ y el valor real de la calificación $r_{u,i}$ para todos los usuarios evaluados $u \in U$ y todos los ítems en el conjunto de pruebas$testset_u$(Jannach et al., 2011). Los resultados obtenidos se presentan en la Tabla 2, en la cual se muestra por cada clase el nombre de la clase, el número de instancias evaluadas y el MAE obtenido. En la clase 1 se obtuvo un MAE de 3.5 con 4 instancias evaluadas, en la clase 2 se obtuvo un MAE de 2.2 con 10 instancias evaluadas, en la clase 3 se obtuvo un MAE de 1.11 con 26 instancias evaluadas, en la clase 4 se obtuvo un MAE de 0.61 con 68 instancias evaluadas, en la clase 5 se obtuvo un MAE de 0.59 con 184 instancias evaluadas y finalmente se obtuvo un MAE general de 0.74 con 292 instancias evaluadas.

Tabla 2. Resultados del experimento con la métrica de evaluación MAE.

Clase	No. Instancias	MAE
1	4	3.5
2	10	2.2
3	26	1.11
4	68	0.61
5	184	0.59
MAE GENERAL	**292**	**0.74**

Los mejores resultados con la métrica MAE los presentan las clases que tienen un mayor número de instancias en el conjunto de pruebas que son las clases 5 y 4. Los resultados con mayor error los presentan las clases que tienen un menor número de instancias en el conjunto de pruebas que son las clases 3, 2 y 1.Sin embargo, el MAE general es de 0.74 porque esta métrica tiene la ventaja de penalizar los valores de error a típicos o muy elevados, además esta métrica es un excelente indicador para buscar la precisión en las predicciones (Rackaitis, 2019).

5 Conclusiones

En este trabajo se generó un modelo de recomendación con técnicas de recuperación de información para la tarea de sistema de recomendación del Rest-Mex 2022, se desarrolló un modelo espacio vectorial con un esquema de ponderación tf-idf que utiliza la similitud del coseno y un índice invertido por cada usuario para implementar el algoritmo de Cosine Score. Además, se aplicó la técnica de recomendación del filtrado colaborativo basado en el ítem utilizando las técnicas de RI. El modelo de recomendación se evaluó con la métrica de MAE y obtuvo un valor de 0.74 general, este resultado es aproximado al MAE de 0.73 del Baseline y supera al equipo Labsemco-UAEM que obtuvo un MAE de 1.65 en el Rest-Mex 2021, por tal motivo, este modelo de recomendación se tomará como base en trabajos futuros para abordar otros desafíos de la tarea de sistema de recomendación del Rest-Mex 2022 como son los textos turísticos en idioma español e inglés y el arranque enfrío (usuarios sin reseñas textuales).

Referencias

1. Álvarez, M. Á., Aranda, R., Arce, S., Fajardo, D., Guerrero, R., López, A. P., Martínez, J., Pérez, H., & Rodríguez, A. Y. (2021). *Overview of Rest-Mex at IberLEF 2021: Recommendation System for Text Mexican Tourism*. Procesamiento Del Lenguaje Natural, 67, 163–172.
2. Arreola, J., Garcia, L., Ramos, J., & Rodríguez, A. (2021). *An Embeddings Based Recommendation System for Mexican Tourism . Submission to the REST-MEX Shared Task at IberLEF 2021*. Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2021), 2943, 110–117.
3. Çano, E., & Morisio, M. (2017). *Hybrid Recommender Systems: A Systematic Literature Review*. Intelligent Data Analysis, 21(6), 1487–1524.
4. Flaticon. (2022). *Iconos vectoriales y stickers - PNG, SVG, EPS, PSD y CSS.* https://www.flaticon.es/
5. Idrissi, N., Zellou, A., Hourrane, O., Bakkoury, Z., & Benlahmar, E. H. (2019). *A New Hybrid-Enhanced Recommender System for Mitigating Cold Start Issues*. ICIME 2019: Proceedings of the 2019 11th International Conference on Information Management and Engineering, 10–14.
6. INEGI. (2022). *Turismo*. https://www.inegi.org.mx/temas/turismo
7. Jannach, D., Resnick, P., Tuzhilin, A., & Zanker, M. (2016). *Recommender Systems — Beyond Matrix Completion*. COMMUNICATIONS OF THE ACM, 59(11), 94–102.
8. Jannach, D., Zanker, M., Felfernig, A., & Friedrich, G. (2011). *Recommender Systems An Introduction (1st ed.)*. Cambridge University Press.
9. Kaššák, O., Kompan, M., & Bieliková, M. (2016). *Personalized Hybrid Recommendation for Group of Users: Top-N Multimedia Recommender*. Information Processing & Management, 52(3), 459–477.
10. Malekpour Alamdari, P., Jafari Navimipour, N., Hosseinzadeh, M., Ali Safaei, A., & Darwesh, A. (2020). *A Systematic Study on the Recommender Systems in the E-Commerce*. IEEE Access, 8, 115694–115716.
11. Manning, C., Raghavan, P., & Schütze, H. (2009). *An Introduction to Information Retrieval*. Cambridge University Press.
12. Morales, E., Torres, D., Ehrlich, A., Toledo, M., Martínez, B., & Hermosillo, J. (2021). *A Recommendation System for Tourism Based on Semantic Representations and Statistical Relational Learning*. Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2021), 134–148.
13. Nilashi, M., Ibrahim, O., & Bagherifard, K. (2018). *A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques*. Expert Systems With Applications, 92, 507–520.
14. Rackaitis, T. (2019). *Evaluating Recommender Systems: Root Means Squared Error or Mean Absolute Error? Towards Data Scienc*e. https://towardsdatascience.com/evaluating-recommender-systems-root-means-squared-error-or-mean-absolute-error-1744abc2beac
15. Rest-Mex. (2022). *Rest-Mex 2022: Recommendation System, Sentiment Analysis and Covid Semaphore Prediction for Mexican Tourist Texts*. https://sites.google.com/cicese.edu.mx/rest-mex-2022
16. Tahmasebi, F., Meghdadi, M., Ahmadian, S., & Valiallahi, K. (2021). *A hybrid recommendation system based on profile expansion technique to alleviate cold start problem*. Multimedia Tools and Applications, 80(2), 2339–2354.

Capítulo 7. Reinforcement learning como generador de analíticas prescriptivas en el dominio de tratamientos dinámicos para cáncer de mama

Gustavo Emilio Mendoza Olguín[1], María Josefa Somodevilla García[1], María de la Concepción Pérez de Celis Herrero[1], Yanin Chavarri Guerra[2]
[1]Benemérita Universidad Autónoma de Puebla. Facultad de Ciencias de la Computación
[2]Instituto Nacional de Ciencias Médicas y Nutrición Dr. Salvador Zubirán. Departamento de Hematología y Oncología
e-mail autor por correspondencia. gustavo.mendozao@alumno.buap.mx

Resumen. En el presente trabajo se hace una revisión de las aplicaciones del *reinforcement learning* y sus variantes en el ámbito del cuidado de la salud. La justificación de la técnica se debe a que de forma natural puede utilizarse para proveer analíticas prescriptivas, las cuales buscan responder a las preguntas ¿Qué debo hacer y por qué debo hacerlo? dentro del proceso de Ciencia de Datos. Se propone el uso de enfoques basados en *reinforcement learning* para la elaboración de herramientas auxiliares que faciliten la elección de tratamientos para cáncer de mama que su vez maximicen los resultados esperados (entendiendo como resultado esperado la disminución o remisión del crecimiento tumoral) por los profesionales de salud en los pacientes y minimicen los efectos secundarios y la toxicidad asociados a la medicación coadyuvante o al tratamiento por quimioterapia o radioterapia en estos últimos.

Palabras Clave: Reinforcement learning, análisis prescriptivo, ciencia de datos.

1 Introducción

La Ciencia de Datos (*Data Science*) es un enfoque multidisciplinario para obtener perspectivas (*insights*) a partir de *datasets* creados, almacenados y que crecen día con día en forma estructurada y no estructurada. Este enfoque envuelve tres etapas principales: preparación, análisis y presentación. El análisis de datos puede ser destinado a describir, predecir o prescribir comportamientos a partir de los datos. El análisis prescriptivo combina las herramientas y técnicas de los otros dos análisis e integra algoritmos, reglas de negocio, modelos de computadora y procedimientos de *Machine Learning* sobre diferentes *datasets*: transaccionales e históricos, de manera que combina *Big Data (BD)* y datos en tiempo real y ayuda a resolver problemas que involucran BD, investigación de operaciones, sistemas de soporte de decisiones y optimización (Poornima y Pushpalatha, 2020).

Actualmente, la producción del conocimiento está completamente ligado a la tecnología y sus aplicaciones en cada campo de la ciencia. El uso de técnicas de AI, específicamente ML en el campo médico y biomédico está aumentando, prometiendo diagnósticos, herramientas y pronósticos terapéuticos mejorados (Starke et al., 2021). Por lo tanto, los avances en esta área tienen dos principales problemáticas:

- Recolectar datos médicos es complicado debido a las implicaciones éticas y de privacidad que conllevan; aunado a la diversidad de áreas biomédicas y sus propios requerimientos.
- El análisis de la información es una tarea compleja como consecuencia natural de la diversidad de la información y de la ausencia de información relacionada.

El análisis prescriptivo es sugerido como el paso siguiente en camino a la madurez del análisis de datos y a la toma de decisiones óptimas. Este apunta a sugerir (prescribir) las mejores opciones de decisión para tomar ventaja de las predicciones obtenidas de grandes conjuntos de datos. Este análisis involucra diferentes herramientas en un contexto probabilístico para proveer decisiones óptimas, adaptativas, automáticas, restringidas y dependientes del tiempo (Lepenioti et al., 2020). Los sistemas prescriptivos entregan al usuario recomendaciones óptimas para la acción, y la calidad de estas recomendaciones es medida por su exactitud (Singh et al., 2020).

El objetivo de este trabajo es el de justificar a las analíticas prescriptivas como la base de las nuevas herramientas que deben ser diseñadas como auxiliares de los profesionales de salud; y específicamente, del potencial que tiene el *reinforcement learning* (RL) para la obtención de estas analíticas dentro del campo de la salud. El documento se encuentra organizado como sigue: en la segunda sección se presenta el RL como herramienta para generar analíticas prescriptivas. La tercera sección analiza las aplicaciones actuales de las analíticas prescriptivas obtenidas mediante RL en el proceso de prescripción de tratamientos médicos y, finalmente, las conclusiones y el trabajo a futuro.

2 Las analíticas prescriptivas para la toma de decisiones y el reinforcement learning

El concepto de *analytics* es definido como la combinación de arte y ciencia para descubrir *insights* significativos y novedosos a partir de un volumen de datos variados mediante la aplicación de técnicas como ML y algoritmos matemáticos y estadísticos, para apoyar a la toma de decisiones oportunas. Las analíticas son obtenidas mediante técnicas de minería de datos, minería de texto, minería de web y utilizados para producir análisis descriptivos, diagnósticos, predictivos y prescriptivos que describen un fenómeno (Mosavi y Santos, 2020). La fig. 1 presenta este proceso.

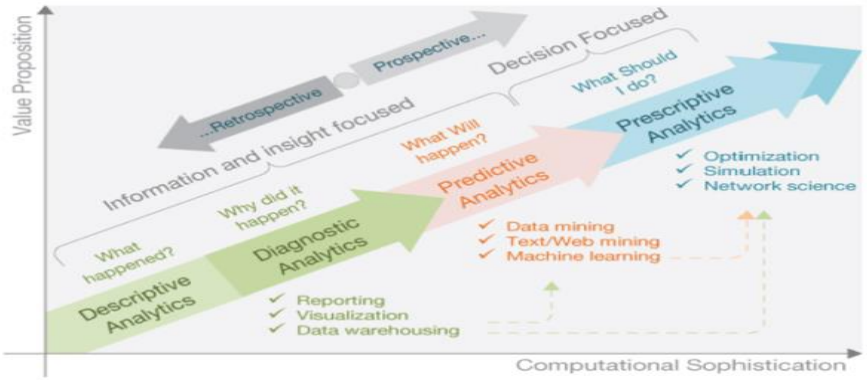


Fig. 1. Proceso de construcción de *analytics*. Fuente: (Mosavi y Santos, 2020)

El análisis prescriptivo usa la salida de todos los procesos previos para presentar la mejor acción posible utilizando técnicas como optimización, simulación y ciencia de redes. Este tipo de análisis responde a la pregunta ¿Qué debo hacer? De acuerdo a los autores, existe una falla en los *frameworks* de toma de decisiones médicas tradicionales debido a que están basados completamente en información limitada sobre el paciente y las experiencias de los médicos en sus tratamientos. Estos (*frameworks*) han causado grandes costos de tratamientos, calidad pobre de los servicios de salud y resultados débiles. Por esto, debe revisarse un nuevo protocolo basado en los datos completos de los pacientes y esta visión es posible mediante el análisis prescriptivo (Mosavi y Santos, 2020). Si bien la investigación en este tipo de técnicas no está tan madura como los otros tipos de análisis, actualmente existe un interés en la investigación de estas técnicas debido a que se consideran el siguiente paso hacia una toma de decisiones óptimas basadas en *data analytics* (Lepenioti et al., 2020).

El análisis prescriptivo tiene dos fases de intervención humana: el soporte de decisión y las decisiones automáticas. La efectividad de las acciones depende en qué tan bien estos modelos incorporen una combinación de datos estructurados y no estructurados que representen el dominio de estudio y capturen el impacto de las decisiones tomadas (Lepenioti et al., 2020).

Dentro de las técnicas de *Machine Learning* y *Data Mining* utilizadas para generar analíticas prescriptivas se encuentra el *reinforcement learning*; este consiste en aprender que hacer – en contexto de situaciones a acciones – de manera que se maximice una recompensa numérica. A un agente inteligente (*learner*) no se le dice cuáles acciones debe tomar; en cambio, debe descubrir cuales acciones conducen a la mayor recompensa al probarlas. En los casos más interesantes y retadores, las acciones no solo afectan a las recompensas inmediatas sino a todas las subsecuentes recompensas. Estas dos características: prueba y error y recompensas postergadas son sus principales diferenciadores (Sutton y Barto, 2018).

En esta técnica, el agente trabaja en un ambiente conocido o desconocido adaptando y aprendiendo constantemente basándose en *feedback* que puede ser positivo (*reward*) o negativo (*punishment*). Sus principales diferencias con el ML tradicional radican en que no se requieren datos de entrenamiento; la interacción ocurre con los ambientes en vez de los datos, por esto se requiere un mayor número de parámetros que pueden entrar

en juego; los escenarios usualmente son mundos simulados en 2D o 3D o escenarios de juego y que el objetivo del RL es alcanzar una meta (Nandy y Biswas, 2018).

La aplicación del RL se basa en un conjunto de reglas simples iniciales, las cuales son aprendidas por el agente para posteriormente poder tomar decisiones complejas basadas en un conjunto de reglas más amplio. El agente debe ser capaz de identificar el estado de su entorno en cierta forma y debe tomar decisiones que afecten el estado; además debe tener una meta o metas relacionadas con el estado del entorno (Sutton y Barto, 2018). Al tomar una decisión, deben generarse nuevos escenarios basados en el nuevo estado del entorno, y el agente debe cambiar entre los roles de aprendiz y de tomador de decisiones.

Un reto del agente radica en que no solo se debe explotar el conocimiento obtenido previamente, sino que también debe explorarse activamente el escenario actual; pues probablemente hacer las cosas de diferente manera podría mejorar los resultados. El problema es que demasiada observación puede decrementar la recompensa o causar que se olvide lo que se ha aprendido; por lo que debe encontrarse un balance entre esas actividades de alguna manera. El dilema de la explotación / exploración es una de las preguntas activas del RL. Un segundo reto radica en que la recompensa puede ser alcanzada después de las varias acciones realizadas. El agente debe descubrir causalidades que pueden engañar al flujo del tiempo y de las acciones (Lapan, 2020).

Además del agente y el entorno, existen cuatro elementos importantes involucrados en el proceso de aprendizaje. Estos son: las políticas, la señal de recompensa, la función de evaluación y, opcionalmente, un modelo del entorno.

Una política define la forma en que el agente se comporta en un determinado momento. Las políticas definen las acciones que el agente debe tomar dependiendo de los estados que han sido percibidos en el entorno, definiendo su comportamiento. En general, las políticas son estocásticas, especificando probabilidades para cada acción (Sutton y Barto, 2018).

La señal de recompensa define la meta de un problema de RL. En cada paso, el entorno envía al agente un número llamado *reward*. El único objetivo del agente es maximizar la recompensa total que recibe a través de los pasos realizados. Esta señal también define cuáles son los eventos "buenos" y "malos" para el agente, de manera que definan la forma en que se altera la política: si una acción seleccionada por la política es seguida de una recompensa baja, entonces la política debe ser cambiada para seleccionar otra acción en esa situación en el futuro. En general, estas recompensas pueden ser funciones estocásticas sobre el estado del entorno y las acciones seleccionadas (Sutton y Barto, 2018).

Mientras que la señal de recompensa indica que es "bueno" en un sentido inmediato, la función de valor especifica que es bueno durante todo el proceso. Es decir, el valor de un estado es la cantidad total de recompensas que un agente puede acumular en el futuro comenzando desde un estado específico (Sutton y Barto, 2018). Desafortunadamente, es más difícil determinar la función de valor que las recompensas, pues las primeras deben ser tomadas a través de juicios de valor, mientras que las segundas pueden ser tomadas de forma voraz. Las recompensas son directamente ofrecidas por el entorno, mientras que la función de valor debe ser estimada iterativamente a partir de las secuencias de observaciones que el agente ha realizado a lo largo del recorrido. Este es el componente más importante del RL y se ha mantenido como una pregunta abierta por las últimas seis décadas (Sutton y Barto, 2018). El último

elemento es un modelo del entorno, el cual es una función que imita el comportamiento de éste y permite hacer inferencias sobre los cambios a partir de las acciones que el agente realice. Los modelos son usados para planear el estado final del entorno ante cualquier acción futura incluso antes de realizarlas. Este elemento es opcional, pues puede haber agentes basados en modelos y agentes *free-model*. Los primeros utilizan métodos que involucran al modelo en la toma de decisiones y los segundos son específicamente modelos de prueba y error. Una de las ventajas del RL es que un agente puede comportarse en cualquier momento como basado en modelo, agente libre y construir un modelo a partir del entorno.

3 Estado del arte

El uso de analíticas prescriptivas en la medicina ha sido abordado desde distintas perspectivas y técnicas y existen ya algunos algoritmos como el propuesto por Kolcu y Murat; el cual parte de un enfoque predictivo y, con el entrenamiento de variables, lo convierte en un resultado prescriptivo usando técnicas de optimización de dos niveles y comparan sus resultados con los obtenidos mediante optimización estocástica, estimación de puntos y optimización por ML. En su trabajo proponen un problema de decisión con parámetros desconocidos y tratan de estimar la relación predictiva entre las respuestas y un conjunto de entradas. Los autores consideran que un problema de decisión puede ser analizado como un modelo de optimización y que las relaciones estadísticas entre parámetros desconocidos pueden ser aproximadas mediante un modelo de regresión paramétrica (Kolcu y Murat, 2021) y proponen un *framework* que fue comparado con varios métodos mediante experimentos numéricos sobre problemas conocidos. Los autores concluyen que su propuesta puede ser utilizada en cualquier tarea prescriptiva con parámetros desconocidos. Desafortunadamente, este método está restringido a casos donde la tarea prescriptiva tenga una forma lineal; pero puede ser utilizado para generar otros métodos que involucren métodos predictivos y prescriptivos.

Existe un incremento en datos que detallan la complejidad y heterogeneidad del cáncer, sin embargo, esto no se ha traducido en regímenes de tratamiento más actualizados (Nagy et al., 2020). De acuerdo al autor, las diferentes técnicas de ML aplicadas a los datos prometen avances en este sentido. Los autores realizan una investigación sobre la literatura de las técnicas de ML utilizadas en oncología y muestran como representativos las técnicas de redes neuronales, redes neuronales convolucionales, *randomforest*, sistemas de recomendación y redes neuronales profundas. Los autores terminan haciendo recomendaciones de cómo deben ser consideradas las herramientas por los profesionales de la salud. Sin embargo, no hacen mención al RL dentro de las técnicas representativas.

Son Yu y su equipo, en una revisión sistemática de aplicaciones del RL dentro del contexto del área de la salud, quienes presentan al RL como una técnica utilizable. En su trabajo, analizan el RL desde el punto de vista de los procesos de Markov (MDP) encontrando aplicaciones desde esa perspectiva. Como justificación, mencionan que uno de los objetivos de la toma de decisiones médicas es desarrollar regímenes efectivos de tratamientos que puedan ser adaptados dinámicamente a la variedad de los

estados clínicos de los pacientes y optimizar los beneficios a largo plazo (Yu et al., 2021). Los autores llaman a esto "regímenes de tratamiento dinámicos" (DTR, por sus siglas en inglés), los cuales están compuestos de una secuencia de reglas de decisión para determinar el curso de acción en un punto específico del tiempo de acuerdo al estado actual de salud y el historial de tratamientos previos del paciente. En este sentido, los estudios mencionados relacionados a la generación de DTR tienen las limitaciones en el establecimiento de las recompensas, además, de que los datos utilizados fueron generados por simulación.

Es en el trabajo de Meyer et al, en el que se presentan las primeras menciones al RL como técnicas representativas en el área de la radioterapia. El autor menciona la utilización un algoritmo de optimización a priori que genera directa y automáticamente un plan óptimo para cada paciente utilizando una "*wishlist*" con dosis clínicas predefinidas o restricciones basadas en protocolos. Su modelo de optimización pasa por dos etapas: en la primera el algoritmo optimiza el valor objetivo individual para cumplir con las diferentes metas requeridas mientras respeta el orden de prioridad de las restricciones definidas. Si la meta no puede lograrse, se restringe el valor alcanzado y se excluye de una siguiente ejecución para una siguiente dosis. Si el valor obtenido es menor que la dosis meta, entonces la dosis meta es restringida, dando espacio a la optimización para los demás objetivos. Durante la segunda etapa, el optimizador reduce la dosis obtenida durante la primera pasada tanto como sea posible. Sin embargo, solo se tienen algunas pruebas de laboratorio siempre evaluadas por un profesional humano (Meyer et al., 2021).

El trabajo de Liu et al., presenta la aplicación del RL en la localización de tumores de cáncer de pulmón (Liu et al., 2019). En su trabajo, utilizan *Deep RL* para la localización de tumores basados en imagenología. Si bien su trabajo no se relaciona con el tratamiento, dejan en claro en las conclusiones que el modelo puede utilizarse también para la generación de tratamientos e intervenciones quirúrgicas menos invasivas.

Murphy presenta una propuesta de simulación y RL para generar tratamientos de pacientes con cáncer de ovario (Murphy et al., 2021). En este trabajo, utilizan los datos de 225 pacientes obtenidos del *dataset* de *The Cancer Genome Atlas* ("TGCA") y obtienen los resultados de las respuestas de los pacientes a las terapias del trabajo de (Villalobos et al., 2018) y mediante simulación, utilizan RL para obtener tratamientos que igualen la supervivencia de las pacientes de acuerdo con los resultados médicos. La función de valor en este caso fue una red convolucional profunda, la cual se encarga de escoger el tratamiento de acuerdo a los resultados obtenidos previamente y los pares de estado – acción fueron alimentados al agente para tomar la decisión de forma estocástica. La recompensa es el número de meses de supervivencia mientras la acción no resultara en muerte. Las pruebas obtuvieron una media de supervivencia de 45.5 meses sobre los 43.4 meses obtenidos por los médicos tras 20000 simulaciones.

Otra aplicación de *Deep RL*, la cual es utilizada para la creación de tratamientos personalizados de cáncer de próstata se presenta en (Shen et al., 2021). Este trabajo es una actualización de un trabajo anterior donde se cambia la función de valor de un modelo voraz hacia un modelo basado en conocimiento. Los autores realizaron pruebas con 74 pacientes de los cuales 15 fueron utilizados para validación y los demás para pruebas. Los resultados obtenidos demuestran que los tratamientos obtenidos basados en conocimiento son mejores que los obtenidos en base a la función voraz. Dentro de

sus conclusiones establecen que los requerimientos computacionales pueden ser altos y que la portabilidad hacia otro tipo de cáncer requiere un reentrenamiento completo del sistema. La técnica utilizada para la toma de decisiones es una red neuronal profunda, por lo que se requiere de datos de entrenamiento, además de que, de acuerdo a los autores, el trabajar con tipos de cáncer más complejos aumentarán la complejidad de la red neuronal lo que se traduce en mayores requerimientos.

El RL ha sido utilizado también para optimizar los parámetros para la radioterapia como se presenta en (Hrinivich y Lee, 2020). Los autores utilizan una red neuronal convolucional para controlar la dosis con los parámetros de entrada. La función de valor está definida por el costo acumulativo basado en dosis, y la política del sistema es minimizar dicha función. Se utiliza una red de dos dimensiones que optimiza cada hoja de forma independiente mientras monitorea las dosis correspondientes. Este enfoque fue probado en 40 pacientes de cáncer de próstata, con 15 pacientes para entrenamiento y 5 para validación, y el resto para prueba. Los resultados obtenidos les permiten concluir que si se amplía la red a tres dimensiones podrán realizar la calibración más efectiva sin necesidad de tener datos de entrenamiento.

El uso del RL en la optimización de drogas anticáncer que han fallado las pruebas de fase III debido a su baja eficacia y/o alta toxicidad es presentado en (Park et al., 2020). Para esto, los autores utilizan un modelo de RL que utiliza una red neuronal convolucional para predecir la vinculación entre el *affinity score* (AS) y la absorción, distribución, metabolismo, excreción y toxicidad (ADMET). Para la recompensa, utilizan una Red neuronal molecular de manera que el nuevo modelo pueda considerar múltiples propiedades. Para sus pruebas, utilizaron tres medicamentos que fallaron en las pruebas de fase III (Iniparib, brivanib y rebimastat) obteniendo que dos de ellas pueden ser optimizadas.

Dentro de los algoritmos más utilizados, se encuentran *Q-Learning, Inverse Reinforcement Learning, Contextual bandits, Sarsa, Actor critic, UCB, Deep Reinforcement Learning, DQN* entre otros (den Hengst et al., 2020). La elección del algoritmo depende de factores como la eficiencia de las muestras, la convergencia de los resultados, el horizonte finito o infinito del problema y la visión parcial o total del proceso de decisión de Markov (Liu et al., 2019). Además, deben tomarse en cuenta los retos a los que aún se enfrenta el RL dentro del campo del cuidado de la salud: el aprendizaje a partir de datos limitados, la definición de un performance óptimo, la calidad de los datos y la decisión entre explotación y exploración, así como la complejidad de representación del entorno del agente (Liu et al., 2019; Yu et al., 2021; Nagy et al., 2020). Una síntesis de los trabajos realizados se presenta en la Tabla 1.

Tabla 1. Síntesis del estado del arte. Fuente: elaboración propia

Aplicación	Referencia	Método	*Dataset*	Ventajas	Desventajas
Dosis óptima de tratamiento de quimioterapia en cáncer	(Liu et al., 2019)	*Deep RL* basado en Q-network	114 pacientes	Utiliza *Deep learning* para la elaboración de la función de transición	Implementan una GANN para simular pacientes
	(Murphy et al., 2021)	*Deep RL* basado en MDP + CNN	TCGA	Determinan la función de transición	Evalúan la esperanza de supervivencia

				mediante MDP + CNN	contra los reportados en el *dataset*.
	(Shen et al., 2021)	*Deep RL* + Knowledge Base	74 pacientes 15 para evaluación	*Treatment Planner System* basado en conocimiento	Sin modelo del contexto. No probado con objetivos clínicos reales.
Aplicación	Referencia	Método	*Dataset*	Ventajas	Desventajas
Dosis óptima de tratamiento de radioterapia en cáncer	(Meyer et al., 2021)	*Automated Rule Implementation and reasoning*	NA	No requiere datos de entrenamiento	Tiempo de obtención de resultados
		Atlas-Based Knowledge Base	2000 pacientes	Rapidez, planes homogéneos, calidad de las propuestas	Requiere de una base de conocimientos de 100 a 150 casos.
		Random Forest	2500 pacientes.	Rapidez, homogeneidad en los resultados	Requiere de al menos 100 planes para entrenar.
		A priori MCO	En desarrollo	No requiere entrenamiento, única solución óptima.	Tiempo requerido para elaborar la *whishlist*.
	(Bakx et al., 2021)	U-Net	90 pacientes para entrenamiento 15 para evaluación	Utilizan CNNS para dosis óptimas y doble proceso de optimización	No presenta resultados comparativos.
		Contextual Atlas Regression Forest cARF			Modelo bidimensional con pacientes similares
	(Hrinivich y Lee, 2020)	RL + CNN	40 pacientes 15 para evaluación	Calibración de equipo de radioterapia a partir los resultados de imagenología.	Usan un modelo bidimensional. Requiere datos de entrenamiento para la CNN.
Otras aplicaciones de RL	(den Hengst et al., 2020)	RL	NA	Aborda algoritmos para la toma de decisiones que involucran personalización.	Visión general de aplicación. Cada algoritmo es útil en contextos específicos.

4 Conclusiones

ElRL ha mostrado ser útil dentro del campo de la medicina (Yu et al., 2021). Sin embargo, se encuentra en un estado inicial de desarrollo de acuerdo a la opinión de varios autores (Poortmans et al., 2020; Liu et al., 2019; Yu et al., 2021). Los principales retos que la mayoría de los autores mencionan abarcan desde la dispersión y limitación de los datos a utilizar, la complejidad de la elección de la función de valor, la técnica utilizada para la toma de decisiones, hasta la evaluación de los resultados obtenidos, entre otros. Estos retos ofrecen una oportunidad importante de investigación en la propuesta de diferentes funciones de valor que a su vez reduzcan los tiempos de respuesta o mejoren la efectividad de las prescripciones obtenidas. La mayoría de los autores coinciden en que es necesario un trabajo multidisciplinario exhaustivo que considere todas las perspectivas del problema. Además, en ninguno de los trabajos se considera el valor de un enfoque prescriptivo, pues los resultados obtenidos en todos se reducen a la selección de tratamientos basados en la predicción. Un enfoque concebido inicialmente desde el punto de vista prescriptivo, utilizando una herramienta como el RL que es intrínsecamente multidisciplinario, representa un campo abierto de investigación que aporte mayor claridad a los resultados obtenidos por los trabajos relacionados basados en técnicas de IA. Desde el punto de vista computacional, el *expertise* en ciencia de datos representa la aportación que las ciencias computacionales ofrecen en el logro de estas herramientas y otras en todos los campos de la ciencia.

Referencias

1. Bakx, N., Bluemink, H., Hagelaar, E., van der Sangen, M., Theuws, J., y Hurkmans, C. (2021). *Development and evaluation of radiotherapy deep learning dose prediction models for breast cancer*. Physics and Imaging in Radiation Oncology, 17, 65–70. https://doi.org/10.1016/j.phro.2021.01.006
2. den Hengst, F., Grua, E. M., el Hassouni, A., y Hoogendoorn, M. (2020). *Reinforcement learning for personalization: A systematic literature review*. Data Science, 3(2), 107–147. https://doi.org/10.3233/DS-200028
3. Hrinivich, W. T., y Lee, J. (2020). *Artificial intelligence-based radiotherapy machine parameter optimization using reinforcement learning*. Medical Physics, 47(12), 6140–6150. https://doi.org/10.1002/mp.14544
4. Kelleher, J. D., y Tierney, B. (2018). *Data Science*. the MIT Press.
5. Kolcu, M., y Murat, A. E. (2021). *Integrated Optimization of Predictive and Prescriptive Tasks*. arXiv:2101.00354 [cs, stat]. http://arxiv.org/abs/2101.00354
6. Lapan, M. (2020). *Deep Reinforcement Learning Hands-On*. Packt.
7. Lepenioti, K., Bousdekis, A., Apostolou, D., y Mentzas, G. (2020). *Prescriptive analytics: Literature review and research challenges*. International Journal of Information Management, 50, 57–70. https://doi.org/10.1016/j.ijinfomgt.2019.04.003
8. Liu, Z., Yao, C., Yu, H., y Wu, T. (2019). *Deep reinforcement learning with its application for lung cancer detection in medical Internet of Things*. Future Generation Computer Systems, 97, 1–9. https://doi.org/10.1016/j.future.2019.02.068
9. Meyer, P., Biston, M.-C., Khamphan, C., Marghani, T., Mazurier, J., Bodez, V., Fezzani, L., Rigaud, P. A., Sidorski, G., Simon, L., y Robert, C. (2021). *Automation in radiotherapy*

treatment planning: Examples of use in clinical practice and future trends for a complete automated workflow. Cancer/Radiothérapie, 25(6), 617–622. https://doi.org/10.1016/j.canrad.2021.06.006

10. Mosavi, N., y Santos, M. (2020). *How Prescriptive Analytics Influences Decision Making in Precision Medicine*. Procedia Computer Science, 177, 528–533. https://doi.org/10.1016/j.procs.2020.10.073
11. Murphy, B., Nasir-Moin, M., von Oiste, G., Chen, V., Riina, H. A., Kondziolka, D., y Oermann, E. K. (2021). *Patient level simulation and reinforcement learning to discover novel strategies for treating ovarian cancer*. https://doi.org/10.48550/arXiv.2110.11872
12. Nagy, M., Radakovich, N., y Nazha, A. (2020). *Machine Learning in Oncology: What Should Clinicians Know?* JCO Clinical Cancer Informatics, 4, 799–810. https://doi.org/10.1200/CCI.20.00049
13. Nandy, A., y Biswas, M. (2018). Reinforcement Learning Basics. En A. Nandy y M. Biswas (Eds.), *Reinforcement Learning: With Open AI, TensorFlow and Keras Using Python* (pp. 1–18). Apress. https://doi.org/10.1007/978-1-4842-3285-9_1
14. Poornima, S., y Pushpalatha, M. (2020). *A survey on various applications of prescriptive analytics*. International Journal of Intelligent Networks, 1, 76–84. https://doi.org/10.1016/j.ijin.2020.07.001
15. Poortmans, P. M. P., Takanen, S., Marta, G. N., Meattini, I., y Kaidar-Person, O. (2020). Winter is over: *The use of Artificial Intelligence to individualise radiation therapy for breast cancer*. The Breast, 49, 194–200. https://doi.org/10.1016/j.breast.2019.11.011
16. Shen, C., Chen, L., Gonzalez, Y., y Jia, X. (2021). *Improving efficiency of training a virtual treatment planner network via knowledge-guided deep reinforcement learning for intelligent automatic treatment planning of radiotherapy*. Medical Physics, 48(4), 1909–1920. https://doi.org/10.1002/mp.14712
17. Singh, K., Li, S., Jahnke, I., Pandey, A., Lyu, Z., Joshi, T., y Calyam, P. (2020). *A Formative Usability Study to Improve Prescriptive Systems for Bioinformatics Big Data*. 2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 735–742. https://doi.org/10.1109/BIBM49941.2020.9313097
18. Starke, G., De Clercq, E., y Elger, B. S. (2021). *Towards a pragmatist dealing with algorithmic bias in medical machine learning*. Medicine, Health Care, and Philosophy, 24(3), 341–349. https://doi.org/10.1007/s11019-021-10008-5
19. Sutton, R. S., y Barto, A. (2018). *Reinforcement Learning: An Introduction (2nd ed.)*. Bradford Books.
20. Villalobos, V. M., Wang, Y. C., y Sikic, B. I. (2018). R*eannotation and Analysis of Clinical and Chemotherapy Outcomes in the Ovarian Data Set From The Cancer Genome Atlas*. JCO Clinical Cancer Informatics, 2, 1–16. https://doi.org/10.1200/CCI.17.00096
21. Yu, C., Liu, J., Nemati, S., y Yin, G. (2021). *Reinforcement Learning in Healthcare: A Survey*. ACM Computing Surveys, 55(1), 5:1-5:36. https://doi.org/10.1145/3477600

Capítulo 8. Localización de una Cámara Monocular Utilizando Métodos de Visión y Aprendizaje Profundo: Una Descripción General

Aldrich Alfredo Cabrera-Ponce[1] Manuel Martín-Ortiz[1] , José Martínez-Carranza[2]
[1]Benemérita Universidad Autónoma de Puebla (BUAP). Facultad de Ciencias de la Computación
[2]Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE). Coordinación de Ciencias de la Computacionales.
e-mail autor por correspondencia. aldrich.cabrera@alumno.buap.mx

Resumen. La localización es una herramienta utilizada para establecer la ubicación de un sistema u objeto en un ambiente por medio de sensores, láseres, y métodos de inspección. Inspirados en ello, diversos trabajos se han estado desarrollando para crear aplicaciones utilizando cámaras monoculares, presentando nuevas soluciones basados en métodos de aprendizaje para obtener las posiciones de la cámara y localizarse dentro de un escenario. Debido a ello, hoy en día los sistemas de localización con cámaras han presentado soluciones novedosas utilizando imágenes simples como medio de información en conjunto de técnicas de visión por computadora, sistemas de localización simultáneo y mapeo (SLAM), y aprendizaje profundo (Deep Learning). Motivados por la amplia variedad de metodologías y soluciones, el objetivo de este trabajo es presentar al lector una breve descripción de los métodos utilizados para obtener la localización, así como sus ventajas y desventajas. Se espera concluir que los sistemas de visualización basados en aprendizaje profundo destacan como uno de los mejores enfoques debido al uso de aprendizaje profundo con imágenes que extrae mayor información, permitiendo así cumplir la tarea de estimación de posición y la relocalización de una cámara en un escenario. Finalmente comparamos y analizamos 5 arquitecturas presentando sus resultados para el desarrollo de una tarea de localización.

Palabras Clave: Localización, Aprendizaje Profundo, Cámara Monocular, CNN

1 Introducción

Hoy en día la localización se ha convertido en una herramienta útil para múltiples sistemas con la capacidad de conocer su ubicación dentro de un escenario. Esta capacidad ha sido llevada a través de sensores tales como Giroscopios, Barómetros, IMU (*Inertial Measurement Unit)*, y GPS (*Global Positioning System*), provocando que se utilice en varios aspectos de la vida diaria en dispositivos móviles. Esto nos permite adquirir nuestra posición y localización de tal manera que nos pueda brindar la información con respecto a dónde nos encontramos. Por ello, diversas aplicaciones han sido desarrolladas empleando un sistema de localización en áreas civiles como

académicos, especialmente dentro de las áreas como la robótica, visión por computadora e inteligencia artificial.

Motivados por esto, la comunidad científica ha desarrollado sistemas de posicionamiento en múltiples tareas tales como marítimas, terrestres y aquellas para el uso civil, así como el industrial. También se han utilizado ampliamente para tareas de navegación en vehículos personales y sistemas robóticos, permitiendo adquirir la ubicación exacta mientras se realiza una tarea durante su trayectoria. Sin embargo, el uso de sistemas convencionales por medio de sensores e información geográfica ha llevado al manejo de altos costos computacionales, así como de hardware para adquirir la posición. Del mismo modo, la fiabilidad de esta puede verse afectada por agentes externos que interrumpen la transferencia de información entre los sensores y el sistema de estimación de posición.

De esta manera, estrategias basadas en información visual utilizando cámaras monoculares e imágenes han proporcionado un nuevo panorama para obtener la localización a través de la perspectiva. Este procedimiento consiste en extraer información para adquirir aquellas características correspondientes a señales dentro del escenario y con ello obtener una posición. En base a ello, múltiples trabajos han sido llevados a cabo utilizando múltiples métodos de visión, así como el uso de aprendizaje profundo. Además de los métodos tradicionales basados en características y algoritmos de coincidencias, existen aquellos con el uso de redes neuronales convolucionales (CNN) basado en el entrenamiento de un modelo a partir de un conjunto de datos.

Esto ha llevado a desarrollar sistemas de localización a partir de una cámara usando las imágenes de lo que observa como entrada. Así un modelo de aprendizaje obtiene un resultado esperado a partir de una secuencia de datos, las imágenes para estos trabajos han proporcionado mejores características para la estimación de la pose sin el uso de otros sensores externos o métodos de procesamiento habituales. Sin embargo, la selección de una correcta red neuronal para la localización ha llevado a la comunidad científica a crear sus propias arquitecturas modificando y mejorando las capas para obtener un mejor resultado y cumplir con el objetivo.

Motivados por lo anterior, el propósito de este trabajo de revisión es proveer al lector una descripción general acerca de los métodos tradicionales existentes y las redes neuronales utilizadas para la estimación de posición de un cámara a partir de imágenes. Esta estimación de la posición implica encontrar la coordenada que localice a una cámara con los datos de entrenamiento. Además, presentamos una breve comparación de algunas de las arquitecturas populares para cumplir este objetivo. Finalmente, la organización de este documento es la siguiente: En la sección 2 se hará una breve descripción de los sistemas de localización utilizando métodos de visión tradicional. En la sección 3 se presenta una breve descripción de los trabajos basados en aprendizaje profundo y las técnicas que utilizan para obtenerla. En la sección 4 presentamos una comparación de algunas de las características de las redes neuronales que existen, el conjunto de datos que utilizan, los resultados que obtienen, así como las ventajas y desventajas. Finalmente, en la sección 5 se dan nuestras conclusiones sobre el trabajo presentado.

2 Localización utilizando métodos de visión

La localización de una cámara dentro de un escenario ha sido un reto para las áreas de la inteligencia artificial y robótica, siendo un problema habitual para tareas de navegación, inspección, y misiones donde un sistema necesite su ubicación. Esto ha llevado a cabo diversos métodos utilizando visión con el objetivo de resolver el problema y obtener la localización de la cámara. Sin embargo, es un reto que hoy en día sigue estando vigente debido a la fiabilidad de las estimaciones de la posición por medio de imágenes correspondientes a un escenario. Varios investigadores utilizan métodos tradicionales tales como coincidencia de características utilizando algoritmos de vecinos más cercanos (Caselitz et al., 2016), recuperación de la imagen a través de descriptores visuales e información visual (Wong et al., 2017) (Figura 1), imágenes georeferenciadas (Costea y Leordeanu, 2016), e imágenes de consulta (Jabborov y Cho, 2020). No obstante, estos trabajos requieren un fino procesamiento y manipulación de la información visual para poder corresponder a una posición externa (Meng et al., 2016). Por un lado, utilizar imágenes satelitales conlleva a un procesamiento costoso debido al amplio tamaño de la imagen y el tiempo que tarda para su análisis (Conte y Doherty, 2008).

Por otro lado, existen sistemas de localización a través de cámaras a bordo de vehículos aéreos en entornos marítimos cuyo sistema es capaz de localizarse utilizando imágenes térmicas georreferenciadas (Helgesen et al., 2019). No obstante, estos enfoques aún dependen de un alto costo para adquirir la información de una imagen y su referencia con coordenadas correspondiente a las zonas donde se requiere localizar. Por lo anterior, los métodos tradicionales basados en coincidencias de características y descriptores visuales contenientes en una imagen presentan una mejora para la estimación de las posiciones y así obtener la ubicación de la cámara (Zamir y Shah, 2014). Del mismo modo las técnicas basadas en puntos clave con vectores de gravedad permiten estimar aquellas posiciones a través de un sistema de coordenadas. En el trabajo de (Jin et al., 2019) presenta un sistema de localización para una cámara a través de la detección de puntos clave, gráficas de redes y algoritmos PnP (*perspective-n-point)*. De manera similar, la posición y localización de una cámara es obtenida por la explotación de la información semántica en el campo (Chebrolu et al., 2019).

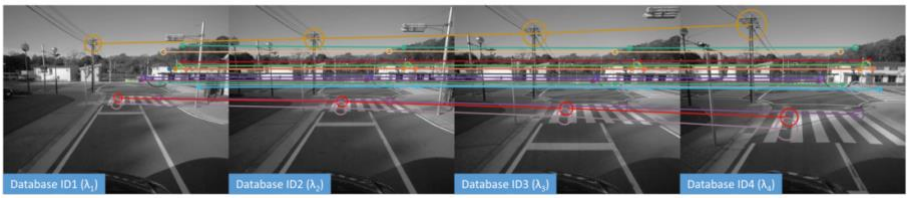


Fig. 1. Características extraídas para la obtención de la posición de la cámara.

En adición a los métodos visuales, existen aquellos que combinan la odometría visual (VO) y sistemas de localización simultáneo y mapeo (SLAM), siendo este último el más utilizado en tareas de mapeo con cámaras (Milford y Wyeth, 2008). Algunos de ellos se han llevado a cabo utilizando cámaras monoculares y la información contenida

en un mapa, permitiendo a un robot con un cámara a bordo, localizarse dentro del escenario. ORB-SLAM3 (Campos et al., 2021) es una versión actualizada del sistema ORB-SLAM2 utilizando descriptores ORB e información local siendo para la creación de un mapa que permita obtener la posición de la cámara. Esto se realiza por medio de coincidencia de características y bolsa de palabras (BoW) dando como resultado una ubicación dentro del mapa generado. Otra alternativa es con transformaciones de distribución normal (NDT) utilizando un método de coincidencia para obtener la localización (Le et al., 2019). Asimismo, la rapidez que proporcionan estos métodos para relocalizar al sistema se han vuelto ideales para tareas de navegación debido a la robustez en diversos escenarios (Mur-Artal y Tardós, 2017; Yang et al., 2019), la Figura 2 presenta el mapeo.

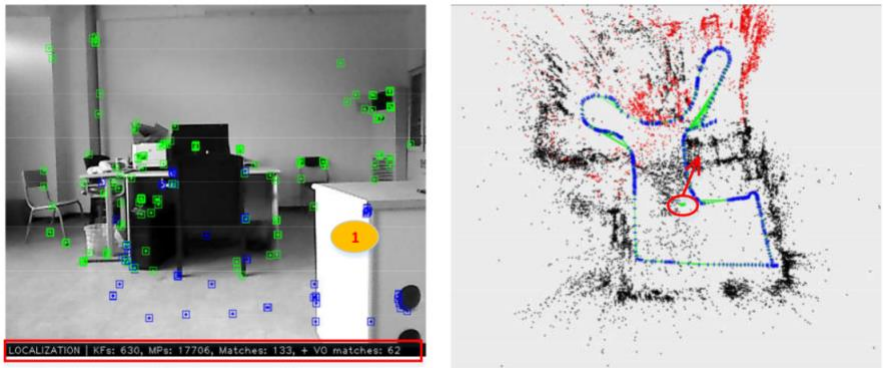


Fig. 2. Mapeo de un escenario utilizando el sistema de ORB-SLAM2 a partir de descriptores visuales e información local y global.

Finalmente existen trabajos donde combinan los sistemas de mapeo sensores externos en robots móviles, esto para obtener posibles posiciones y con ella la localización del robot en escenarios tanto interiores como exteriores (González et al., 2012). Esto ha llevado a que los autores usen técnicas de visión con otros métodos para mejorar la estimación de posición de la cámara, uno de ellos lo combina un proceso de aprendizaje siendo capaz de estimar las poses con respecto a su escenario (Ott et al., 2020). Para este último, hoy en día, existen trabajos que utilizan redes neuronales creando un modelo de aprendizaje que permite obtener la ubicación de una cámara a partir de una única imagen. En resumen, las técnicas basadas en descriptores visuales, coincidencia de características, imágenes satelitales y sistemas de odometría visual y SLAM, han permitido la creación de métodos de localización dentro de la inteligencia artificial y robótica. Además, las soluciones mencionadas lograron la localización de una cámara utilizando imágenes consistentes a un escenario. En la siguiente sección se discuten los trabajos realizados utilizando aprendizaje profundo.

3 Redes neuronales para localización

En esta sección presentamos algunas de las redes neuronales dentro del estado del arte desarrolladas para tareas de localización con una cámara monocular. En el campo de la inteligencia artificial múltiples soluciones han dirigido la localización y estimación de posición con el uso de redes neuronales convolucionales (CNN) teniendo un gran impacto dentro de la sociedad robótica. Una primera arquitectura es presentada en Sarlin et al. (2021) donde se realiza una estimación de posiciones utilizando una red que aprende los píxeles de una imagen. Otro desarrollo de una CNN utiliza un modelo de regresión con GPR (Proceso Gausiana de Regresión) para la estimación de las posiciones de una cámara con una única inferencia (Cai et al., 2018). En Melekhov et al. (2017) utilizan una CNN con un proceso piramidal en la capa de agrupamiento para predecir las posiciones entre dos cámaras. Del mismo modo el uso de redes neuronales siamesas con una arquitectura de AlexNet ha sido empleado para la estimación de la posición y la rotación de una cámara (Charco et al., 2018).

No obstante, existen arquitecturas diseñadas para obtener la localización de un cámara teniendo como base el reconocimiento de una zona, esto puede ser llevado a cabo con información local y geográfica. Por ejemplo, NetVLAD (Arandjelovic et al., 2016) y SPED (Chen et al., 2017) presentan una red basada en descriptores para la localización de una cámara e imágenes capturadas de Google Street View (Figura 3). Del mismo modo, PlaNet (Weyand et al., 2016) crea subdivisiones en base a la localización obtenida a partir de una imagen. Por último, LieNet (Do et al., 2018) detecta y segmenta múltiples objetos dentro de un escenario para estimar los 6 grados de libertad de una cámara empleando una subred dentro de la arquitectura central.

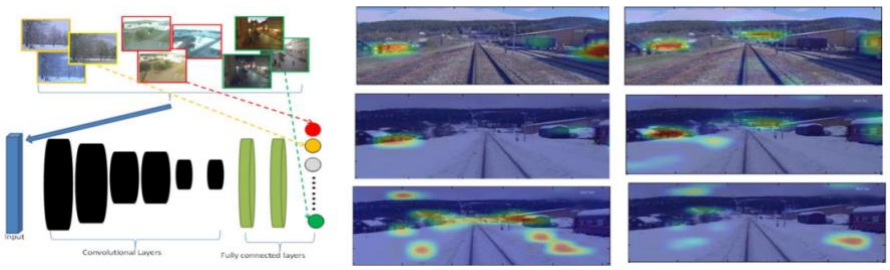


Fig. 3. Localización y reconocimiento de lugar utilizando escala de características.

Las arquitecturas más recientes para la localización de una cámara a partir de una imagen se han popularizado en base al trabajo presentado en Kendall et al. (2015) donde presenta una red de estimación llamado PoseNet (Figura 4) lo cual es construida bajo la arquitectura de GoogleNet. A raíz de ese trabajo, diversas soluciones han salido para mejorar las estimaciones de las posiciones, presentando resultados en relación a su precisión y el error obtenido en metros. Por ejemplo, Cabrera-Ponce y Martinez-Carranza (2019), eliminan 2 capas de estimación dejando solo 1 con el argumento de que las estimaciones de la posición en imágenes aéreas pueden ser aprovechadas para la relocalización de manera más rápida. Otras redes basadas en PoseNet cuya arquitectura tiene implementando módulos Inception capaces de llevar a cabo un

entrenamiento eficaz de la red ha sido cuestionada por investigadores, debido al proceso largo y tedioso en estas capas en (Wang et al., 2020) debatieron la posibilidad de reducir la dimensionalidad de los parámetros entrenados, presentando así una mejora en la arquitectura con un módulo de memoria de término largo y corto (LSTM).

Fig. 4. PoseNet: Red Neuronal Convolucional para relocalización de cámara.

Por otro lado, en Cabrera-Ponce y Martinez-Carranza (2022) se presenta un estudio de las redes basadas en PoseNet y una arquitectura compacta creada para comparar los resultados obtenidos con las redes del estado del arte. El resultado presentado ha demostrado que la estimación de posición de una cámara se desarrolla de una manera más rápida para tareas de navegación al remover algunas de las capas dentro de su arquitectura. Un ejemplo de la rápida estimación es presentado en Cocoma-Ortega y Martinez-Carranza (2021) donde aprovechan una arquitectura compacta basada en PoseNet para localizar a un drone utilizando imágenes en escala de grises. PoseNet++ (Zhang et al., 2018), es otra red de localización para una cámara monocular estimando los 6 grados de libertad con un mejoramiento en la precisión y un rápido rendimiento en tareas de robótica. Finalmente, en Blanton et al. (2020) crean una red llamada Multi-Scene PoseNet (MSPN) para aprender un conjunto de datos específico conectados a las capas finales de la red y compartiendo sus características a través de todas la escenas.

De acuerdo a estos trabajos en la siguiente sección se discuten las características, ventajas y desventajas de 5 redes del estado del arte debido a su proceso de localización, velocidad y rendimiento en las estimaciones de la posición de la cámara.

4 Análisis de las redes neuronales

En esta sección presentamos una breve comparación con 5 arquitecturas del estado del arte para la relocalización de un cámara utilizando PoseNet. Estas redes tienen diferentes características e implementaciones con la cual se compara los resultados obtenidos de cada uno de ellos. Al mismo tiempo se discuten las ventajas y desventajas exponiendo los puntos de vista del autor en base a las soluciones presentadas. Las redes seleccionadas son: PoseNet, PoseNet + LSTM, Compact PoseNet, PoseNet ++, y Multi-Scene PoseNet (MSPN). Los principales módulos para esta comparación y análisis son: 1) Arquitectura de la red; 2) Conjunto de datos utilizado; 3) Tiempo de procesamiento; 4) Error medio obtenido.

4.1 Diseño de Arquitectura

Las arquitecturas presentadas para el análisis de comparación tienen las siguientes especificaciones tales como: el número de capas convolucionales, si cuenta con alguna capa extra, módulo o modificación para mejorar el resultado, tamaño de entrada, y el resultado de salida. Los datos presentados en la Tabla 1 nos da una breve información de la composición de las arquitecturas, esto con el fin de reducir el tiempo de entrenamiento o acelerar el resultado final. Además, modificar una arquitectura puede reducir los parámetros de entrenamiento sin afectar al resultado final siendo útil para esta tarea.

Tabla 1. Especificaciones de las arquitecturas basadas en PoseNet.

CNN	Capas	Inception	Capa Extra	Entrada	Salida
PoseNet	23 Convnet	9 módulos	-	455 X 256	Posición y Cuaternión
PoseNet + LSTM	23 Convnet	9 módulos	Unidades LSTM	224 X 224	Posición y Cuaternión
CompactPN	4 Convnet	3 módulos	División de la FC	224 X 224	Posición
PoseNet ++	16 Convnet	-	2 FC	224 X 224	Posición y Cuaternión
MSPN	23 Convnet	9 módulos	2 Ramas en la FC	224 X 224	Posición y Cuaternión

PoseNet la arquitectura original basada en GoogleNet realiza un corte de la imagen para obtener una dimensión de 224 x 224. Sin embargo, múltiples autores han indicado que este corte demora al procesamiento de la red y por ello a su entrenamiento. Por lo anterior esta característica es el primer paso que los autores realizan antes de entrenar. Este hallazgo encontrado en los 4 trabajos basados en PoseNet, tienen como objetivo reducir el tiempo de procesamiento y acelerar la estimación de la posición. Así el problema de obtener un resultado con múltiples parámetros puede ser reducido para un llevarse a cabo en un ambiente en tiempo real. No obstante, las capas principales donde los resultados son obtenidos es en la Fully Connected Layer (FC) o capa completamente conectada. En esta capa es donde el modelo de aprendizaje en base a los parámetros aprendidos realiza la regresión para adquirir la posición y orientación de la cámara.

PoseNet + LSTM y MSPN realizan un cambio importante antes de la salida de la red. La primera arquitectura agrega una unidad LSTM el cual reduce los parámetros dentro de la FC quedándose con los parámetros más importantes que definirán el resultado de la estimación. La MSPN divide la última capa en dos ramas, la primera división utiliza la imagen para calcular la distribución de las posibles escenas y la segunda rama usa los pesos de las estimaciones para construir la información de salida que pasará a la FC. Por otro lado, PoseNet++ es la única red diseñada a partir de una arquitectura diferente, esta arquitectura es VGG16 donde argumentan que se puede aprovechar la transferencia de aprendizaje en las siguientes capas y regresarlas con un método de mapeo llamado *Georgia Tech Smoothing and Mapping* (GTSAM) utilizando descriptores SIFT.

Finalmente, en la red CompactPN se realiza un estudio sobre las arquitecturas y su diseño para eliminar aquellas capas que no aportan al entrenamiento, quedando de esta manera con 4 capas convolucionales y 3 módulos Inception. Este módulo Inception es una combinación de varias capas convolucionales el cuál mejora el entrenamiento de la red sin caer en un sobreajuste. Al final de la arquitectura se realiza una división en 3 neuronas, una para cada posición en x, y, z, esto con el argumento de que la red dará un resultado más aproximado al real. Este primer análisis de las arquitecturas nos ofrece una breve información de las modificaciones que se han llevado a cabo para obtener los resultados esperados. Además, se plantea el uso de redes neuronales más sofisticadas y rápidas que puedan otorgar un resultado en tiempo real sin demorar tanto en el entrenamiento, siendo una meta a lograr para resolver el problema de localización utilizando imágenes.

4.2 Resultados Obtenidos

Como resultados obtenidos tomamos en cuenta los presentados por los autores. Cabe aclarar que los conjuntos de datos al menos para CompactPN es diferente a los otros trabajos. Sin embargo, se toma en cuenta los resultados en términos de velocidad, y error medio de las posiciones obtenidas. En la Tabla 2 se presenta de manera breve la información donde podemos comparar la eficiencia de las arquitecturas desarrolladas. Como un primer análisis se puede observar en la Tabla 2 que la red original PoseNet sigue presentando un resultado admirable, no obstante, los otros trabajos no se quedan atrás al exponer su mejoría con respecto a los resultados de velocidad contra la red original.

De este modo, la velocidad de estimación es importante para establecer la localización de una cámara en tiempo real, sobre todo en tareas de navegación y robótica. Argumentamos también en base a los resultados presentados en CompactPN que es viable modificar una red de estimación al eliminar algunas capas para acelerar la velocidad de rendimiento. Otro factor importante es que la computadora que se maneja en cada uno de estos trabajos tiene un poder medio alto con el cual se espera que sea posible obtener las posiciones y la relocalización de la cámara en tiempo record.

Tabla 2. Comparación de los resultados de las arquitecturas basadas en PoseNet para la localización de una cámara monocular.

CNN	Error Medio	Velocidad
PoseNet	0.47m, 6.93◦	200 fps
PoseNet + LSTM	1.31 m, 2.79◦	-
CompactPN	2.82 m	102.44 fps
PoseNet ++	12.64738 m, 0.555°	22 fps
MSPN	2.67 m, 6.18°	-

Finalmente, a vista de este autor, se presentó un breve análisis de las redes neuronales presentes en el estado del arte, encontrando que las arquitecturas basadas en PoseNet son las más populares para lidiar con el problema de localización y estimación de posición. Sin embargo, aún es un reto dentro de la robótica debido a que depende de

un extenso conjunto de datos para poder entrenar el modelo de estimación. Esto último nos ha motivado a seguir investigando redes neuronales con métodos de entrenamiento continuo para poder llevar a cabo la relocalización de una cámara en tiempo real durante la misma misión de inspección. Además, con los trabajos relacionados presentados en este artículo se ha demostrado que se sigue avanzando en mejorar la velocidad de una red para obtener un entrenamiento y resultado de manera rápida sin afectar a la precisión de la estimación También a opinión de este autor, el trabajo es presentado como una breve introducción para que el lector conozca los trabajos más importantes hasta la fecha de hoy y se continuará con la investigación de la misma. De tal manera que se logre el objetivo para sistemas sofisticados, así como sistemas robóticos que puedan emplear estos métodos para tareas de localización utilizando una cámara monocular, así como la navegación dentro de escenarios tanto internos como externos.

5 Conclusión

Hemos presentado una revisión de la literatura sobre los métodos más avanzados para la estimación de posición y localización de una cámara a través de redes neuronales y técnicas de visión. Primero, hemos proporcionado una breve explicación y descripción de los métodos de visión tradicional que se encuentran en el estado del arte para la localización utilizando imágenes. Después se procedió a comparar las redes neuronales utilizadas para lograr dicho objetivo comparando las características que los autores utilizan, así como sus ventajas y desventajas contra otras redes. También presentamos un breve análisis sobre estos métodos y los resultados obtenidos al utilizarlos con imágenes monoculares. Finalmente hemos incluido una discusión sobre la información obtenida dando nuestro punto de vista en base a los resultados de cada trabajo.

En general hemos presentado una revisión del estado del arte sobre las técnicas de visión que existen para la localización, así como el uso de redes neuronales. En esta revisión optamos por elegir 5 redes neuronales basadas en PoseNet de las cuales son utilizadas mayormente para la localización de una cámara. Estas redes han sido diseñadas en base a la arquitectura de GoogleNet y módulos Inception para una mayor extracción de características de la imagen sin caer en un sobre ajuste. Del mismo modo, las redes basadas en PoseNet fueron diseñadas para presentar resultados mejorados con un error medio menor con respecto al Ground Truth. Sin embargo, no todos los resultados establecen una solución para realizar una localización en tiempo real.

Los resultados presentados y la comparación realizada muestra que PoseNet y Compact PoseNet presenta una mejor estimación con respecto a los otros enfoques. Al mismo tiempo ambos indican un error entre 0.47 m y 2,82 m eficiente para la localización de una cámara en tiempo real. Por otro lado, las otras arquitecturas han mostrado que la manipulación de PoseNet puede proporcionar información útil para diversas tareas que el autor requiera, empleando la localización de una cámara dentro de un escenario. La velocidad de estimación y el procesamiento ha demostrado que los métodos basados en entrenamiento pueden ser utilizados para tareas dentro de la robótica y navegación, proporcionando resultados favorables dentro de estos escenarios.

Así esta revisión ha logrado mostrar los actuales métodos existentes que otorgan la localización de una cámara dentro de un sistema, así como las que están a bordo de los robots. Además, estos métodos y diseños han sido presentados para dar un panorama al lector de las características principales que requiere una red para llevar a cabo una tarea de localización. Finalmente, el trabajo muestra una descripción general de aquellas arquitecturas y métodos que puedan servir a la comunidad e implementarla para tareas de navegación y localización utilizando imágenes.

Referencias

1. Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., & Sivic, J. (2016). *Netvlad: Cnn architecture for weakly supervised place recognition*. In Proceedings of the ieee conference on computer vision and pattern recognition (pp. 5297–5307).
2. Blanton, H., Greenwell, C., Workman, S., & Jacobs, N. (2020). Extending absolute pose regression to multiple scenes. In Proceedings of the ieee/cvf conference on computer vision and pattern recognition workshops (pp. 38–39).
3. Cabrera-Ponce, A. A. & Martínez-Carranza, J. (2019). *Aerial Geo-Localisation for MAVs using PoseNet*. 2019 Workshop on Research, Education and Development of Unmanned Aerial Systems (RED UAS), 192-198. https://doi.org/10.1109/REDUAS47371.2019.8999713
4. Cabrera-Ponce, A. A. & Martínez-Carranza, J. (2022). *Convolutional Neural Networks for Geo-Localisation with a Single Aerial Image*. Journal of Real-Time Image Processing, 1-11.
5. Cai, M., Shen, C., & Reid, I. D. (2018). A hybrid probabilistic model for camera relocalization. In Bmvc (Vol. 1, p. 8).
6. Campos, C., Elvira, R., Rodríguez, J. J. G., Montiel, J. M., & Tardós, J. D. (2021). *Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam*. IEEE Transactions on Robotics, 37(6), 1874-1890.
7. Caselitz, T., Steder, B., Ruhnke, M., & Burgard, W. (2016, October). *Monocular camera localization in 3d lidar maps*. In 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 1926-1931). IEEE.
8. Charco, J. L., Vintimilla, B. X., & Sappa, A. D. (2018). *Deep learning based camera pose estimation in a multi-view environment*. In 2018 14th international conference on signal-image technology & internet-based systems (sitis) (pp. 224–228).
9. Chebrolu, N., Lottes, P., Läbe, T., & Stachniss, C. (2019). *Robot localization based on aerial images for precision agriculture tasks in crop fields*. In 2019 international conference on robotics and automation (icra) (pp. 1787–1793).
10. Chen, Z., Jacobson, A., Sünderhauf, N., Upcroft, B., Liu, L., Shen, C., . . . Milford, M. (2017). *Deep learning features at scale for visual place recognition*. In 2017 ieee international conference on robotics and automation (icra) (pp. 3223–3230).
11. Cocoma-Ortega, J. A., & Martínez-Carranza, J. (2022). *A compact CNN approach for drone localisation in autonomous drone racing*. Journal of Real-Time Image Processing, 19(1), 73-86.
12. Conte, G., & Doherty, P. (2008). *An integrated uav navigation system based on aerial image matching*. In 2008 ieee aerospace conference (pp. 1–10).
13. Costea, D., & Leordeanu, M. (2016). *Aerial image geolocalization from recognition and matching of roads and intersections*. arXiv preprint arXiv:1605.08323

14. Do, T.-T., Pham, T., Cai, M., & Reid, I. (2018). *Real-time monocular object instance 6d pose estimation*. In British machine vision conference (bmvc) (Vol. 1, p. 6).
15. González, R., Rodríguez, F., Guzman, J. L., Pradalier, C. & Siegwart, R. (2012). *Combined visual odometry and visual compass for off-road mobile robots localization*. Robotica, 30(6), 865-878.
16. Helgesen, H. H., Leira, F. S., Bryne, T. H., Albrektsen, S. M., & Johansen, T. A. (2019). *Real-time georeferencing of thermal images using small fixed-wing uavs in maritime environments*. ISPRS Journal of Photogrammetry and Remote Sensing, 154, 84–97.
17. Jabborov, F., & Cho, J. (2020). *Image-Based Camera Localization Algorithm for Smartphone Cameras Based on Reference Objects*. Wireless Personal Communications, 114(3), 2511-2527.
18. Jin, R., Jiang, J., Qi, Y., Lin, D., & Song, T. (2019). *Drone detection and pose estimation using relational graph networks*. Sensors, 19 (6), 1479.
19. Kendall, A., Grimes, M., & Cipolla, R. (2015). *Posenet: A convolutional network for real-time 6-dof camera relocalization*. In Proceedings of the ieee international conference on computer vision (pp. 2938–2946).
20. Le, T., Gjevestad, J. G. O., & From, P. J. (2019). *Online 3d mapping and localization system for agricultural robots*. IFAC-PapersOnLine, 52 (30), 167–172.
21. Melekhov, I., Ylioinas, J., Kannala, J., & Rahtu, E. (2017). *Relative camera pose estimation using convolutional neural networks*. In International conference on advanced concepts for intelligent vision systems (pp. 675–687).
22. Meng, L., Chen, J., Tung, F., Little, J. J., & de Silva, C. W. (2016). *Exploiting random rgb and sparse features for camera pose estimation*. In Bmvc.
23. Milford, M. J., & Wyeth, G. F. (2008). *Mapping a suburb with a single camera using a biologically inspired SLAM system*. IEEE Transactions on Robotics, 24(5), 1038-1053.
24. Mur-Artal, R., & Tardós, J. D. (2017). *Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras*. IEEE Transactions on Robotics, 33 (5), 1255–1262.
25. Ott, F., Feigl, T., Loffler, C. & Mutschler, C. (2020). *ViPR: visual-odometry-aided pose regression for 6DoF camera localization*. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 42-43.
26. Sarlin, P.-E., Unagar, A., Larsson, M., Germain, H., Toft, C., Larsson, V., Pollefeys, M., Lepetit, V., Hammarstrand, L., Kahl, F. y col. (2021). *Back to the feature: Learning robust camera localization from pixels to pose*. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 3247-3257.
27. Wang, Y., Liu, E. & Wang, R. (2020). *Camera Re-localization by Training Multi-dataset Simultaneously via Convolutional Neural Network*. Proceedings of the 2020 3rd International Conference on Signal Processing and Machine Learning, 35-39.
28. Weyand, T., Kostrikov, I., & Philbin, J. (2016). *Planet-photo geolocation with convolutional neural networks*. In the European conference on computer vision (pp. 37–55).
29. Wong, D., Deguchi, D., Ide, I. & Murase, H. (2017). *Single camera vehicle localization using feature scale tracklets*. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, 100(2), 702-713.
30. Yang, G., Chen, Z., Li, Y. & Su, Z. (2019*). Rapid relocation method for mobile robots based on improved ORB-SLAM2 algorithm*. Remote Sensing, 11(2), 149.
31. Zamir, A. R., & Shah, M. (2014). *Image geo-localization based on multiple nearest neighbour feature matching using generalized graphs*. IEEE transactions on pattern analysis and machine intelligence, 36 (8), 1546–1558.
32. Zhang, R., Luo, Z., Dhanjal, S., Schmotzer, C., & Hasija, S. (2018). *Posenet++: A CNN Framework for Online Pose Regression and Robot Re-Localization*.

Capítulo 9. Razonamiento de sentido común computacional para la resolución de pronombres

Mustafa Ali-Saba[1], Darnes Vilariño-Ayala[1], María Somodevilla-García[1], Helena Gómez-Adorno[2]
[1]Benemérita Universidad Autónoma de Puebla. Facultad de Ciencias de la Computación
[2]Universidad Nacional Autónoma de México. Instituto de Investigaciones en Matemáticas Aplicadas y Sistemas
e-mail autor por correspondencia. mustafa.ali@alumno.buap.mx

Resumen. En este artículo se presenta un estudio sobre el problema de ambigüedad referencial en pronombres presente en el lenguaje natural. Se describen algunos de los marcos de referencia conocidos como desafío de los esquemas de Winograd (WSC) propuesto originalmente por Levesque, Davis y Morgenstern (2012), se describen otras variantes de conjuntos de datos como: DPR, PDP, WNLI, WinoGender, WinoBias, WinoFlexi, WinoGrande. Se analizan los métodos propuestos en la literatura para abordar el problema, como también las variantes de este. Finalmente se plantea una conclusión acerca de la importancia que tiene representar conocimiento y razonamiento de sentido común para la comprensión del lenguaje y su repercusión en los sistemas de Inteligencia Artificial de hoy en día.

Palabras Clave: IA, WSC, Sentido común, PLN, resolución de pronombres.

1 Introducción

El sentido común computacional en Inteligencia Artificial (IA) es uno de los más grandes retos desde los comienzos del campo de la investigación en IA en los años cincuenta y posteriores (McCarthy, 1959; Minsky, 1974; McCarthy & Hayes, 1981; Lenat, 1995; Levesque et al., 2012; Schank y Abelson, 2013; Davis, 2014). El sentido común se define como el nivel básico de conocimiento práctico y razonamiento sobre situaciones y eventos cotidianos que son comúnmente compartidos entre la mayoría de las personas (Sap et al., 2020). El conocimiento de sentido común (CSC) y el razonamiento (RSC) son una parte fundamental de un sistema inteligente en la comprensión de lenguaje natural, ya que permite la capacidad de asociar entidades, realizar inferencias o responder a preguntas con ambigüedad acerca de situaciones simples. Un ejemplo de empleo de sentido común para un ser humano en la vida cotidiana es "saber que es posible jalar con una cadena, más no empujar" (Minsky, 1992), en una situación de este estilo, este conocimiento es obvio para una persona de cierta edad, mientras que para una máquina es difícil debido a la carencia de razonamiento y conocimiento del mundo.

La problemática actual que enfrentan los sistemas de IA hoy en día es su ausencia de generalidad en el sentido común, i. e., los sistemas se enfocan en problemas particulares de aprendizaje estadístico, predicción sobre conjuntos de datos, clasificación, sin

embargo, no son capaces de generalizar (Gunning, 2018; Tandon et al., 2018; Tirumala, 2020) a un nivel competitivo de rendimiento al nivel de inteligencia humana. Con el objeto de probar sentido común en los sistemas de IA se han propuesto diferentes marcos de referencia. Uno de ellos es WSC propuesto por Levesque et al. (2012), siendo un problema de correferencia o resolución de pronombres el cual resulta obvio de responder para un ser humano, mientras que, para un sistema de aprendizaje estadístico, aprendizaje profundo (modelos de lenguaje) o incluso un sistema de IA simbólica es difícil.

Representar el conocimiento del mundo o de la vida cotidiana de manera computacional no es una tarea sencilla, han surgido distintas herramientas para ello, algunas de estas son ConceptNet, ATOMIC, OpenCYC, WebChild entre otras. La estructuración del conocimiento se representa como un grafo de conceptos o la minería web de sentido común en motores de búsqueda.

En este artículo se revisan los distintos marcos de referencia para probar sentido común y razonamiento, así como los métodos empleados en el estado-del-arte y se plantea una conclusión sobre la importancia del sentido común en la IA, así como la dificultad persistente en la falta de generalidad de sentido común por parte de los modelos neurales y de lenguaje natural.

2 Marcos de referencia

En cuanto a tareas de comprensión de lenguaje natural se han propuesto marcos de referencia para abordar esta dificultad. Uno de ellos es el desafío del esquema de Winograd (WSC) propuesto por Levesque et al. (2012) con 100 ejemplos originalmente, extendido posteriormente a 273 (WSC273[1]) y 285 ejemplos (WSC285[2]). En este se plantean parejas de preguntas (en Inglés):

- The city councilmen refused the demonstrators a permit because they [feared/advocated] violence. Who [feared/advocated] violence?
- Answers: The city councilmen/the demonstrators.

Las preguntas consisten de los siguientes elementos (Levesque et al., 2012):

1. Dos frases nominales (sustantivos) de la misma clase semántica (hombre, mujer, objeto inanimado, grupo de objetos o personas).
2. Un pronombre ambiguo que puede referirse a una de las dos entidades anteriores.
3. Una palabra especial y una alternativa, tal que, si la palabra especial se reemplaza por la alternativa, el significado de la frase cambia (e.g., en el ejemplo anterior “feared” se intercambia por “advocated”).

[1] https://www.tensorflow.org/datasets/catalog/wsc273
[2] https://huggingface.co/datasets/winograd_wsc

Se tiene así, una pregunta pidiendo la identidad del pronombre ambiguo, y dos respuestas que corresponden a la frase en cuestión. El problema es dado de manera estandarizada incluyendo las respuestas, haciéndolo un problema binario.

Los esquemas invitan al razonamiento acerca del contexto y al conocimiento del mundo (sentido común) para determinar una respuesta correcta. Desde una perspectiva humana los esquemas son fáciles de contestar, sin embargo, para los sistemas de aprendizaje estadístico representan un reto inminente.

Debido a los resultados obtenidos por los modelos de lenguaje como BERT (Devlin et al., 2018) de más del 90% de precisión, se han propuesto variantes del WSC original, aumentando así la complejidad y prueba de nuevos modelos que no reflejan necesariamente sentido común y que consideran información sesgada en sus resultados.

DPR y PDP. Estos conjuntos de datos se refieren a *Definite Pronoun Resolution* y *Pronoun Disambiguation Problem* (Rahman y Ng 2012). El primero es una variante más sencilla de WSC el cuál consta de 1322 ejemplos de entrenamiento y 564 ejemplos de prueba, construidos manualmente. Para PDP se tienen 122 problemas de desambiguación de pronombres recopilados de literatura clásica y popular, periódicos y revistas.

WNLI (Winograd Natural Language Inference). WNLI (Wang et al., 2018) es un conjunto de datos que consiste de 634 ejemplos de entrenamiento, 70 ejemplos de validación y 145 ejemplos de prueba. WNLI forma parte del marco de referencia GLUE[3] y constituye una variante de vinculación textual del WSC original. Cabe destacar que no todas las preguntas vienen en pares y que no todas contienen la palabra especial anteriormente descrita en WSC original. La clase de ejemplos que constituye WNLI es dada una premisa, determinar si la hipótesis que sigue es verdadera o falsa:

- **Premise:** The city councilmen refused the demonstrators a permit because they feared violence
- **Hypothesis:** The demonstrators feared violence.
- **Answer:** True/False

WinoGender. WinoGender (Rudinger et al., 2018) fue creado como un conjunto de datos de diagnóstico con el objetivo de medir el sesgo de género por parte de los sistemas para la resolución de pronombres. Consiste de 120 plantillas de oraciones escritas manualmente junto con sujetos (neutrales al género) y pronombres. En cada oración uno de los sujetos es una ocupación (e.g., el cirujano), y el otro sujeto es un participante (e.g, el paciente), ambos sujetos neutrales al género. Con el total de pronombres que se pueden incluir se tiene un total de 720 esquemas de Winograd. Una oración válida puede ser:

- The *surgeon* operated on the *patient* with great care; **[his/her]** affection had grown over time.

El género del pronombre (his/her) no afecta la respuesta esperada. Cabe destacar que el objetivo de este conjunto de datos no es medir el rendimiento del modelo, si no ayudar a analizar el sesgo de género del modelo.

[3] The General Language Understanding Evaluation (GLUE): https://gluebenchmark.com/

WinoBias. Es un conjunto de datos para resolución de correferencia enfocado a sesgo de género (Zhao et al., 2018). Los autores introducen un conjunto de datos con 3160 oraciones, divididas por igual en desarrollo y prueba. Cada oración contiene dos candidatos que se seleccionan de una lista de trabajos con una proporción de género altamente desequilibrada. Se dan dos tipos de oraciones en el conjunto de datos. Las *oraciones de tipo 1* siguen una estructura que no revela ninguna pista sintáctica (oraciones más difíciles):

- The farmer knows the editor because [he/she] [is really famous/likes the book].

Las *oraciones de tipo 2* se pueden responder según la estructura de la frase (siendo oraciones más fáciles), permitiendo a los modelos desempeñarse mejor:

- The accountant met the janitor and wished [her/him] well.

El conjunto de datos se divide en **pro-estereotípicos** y **anti-estereotípicos**, dependiendo de si el género del pronombre coincide con el género más común de la ocupación de referencia o no. Observan que los modelos disponibles públicamente para la resolución de correferencia muestran una diferencia importante (hasta 21,1% F1) en el rendimiento de los subconjuntos pro y anti del conjunto de datos.

WinoGrande. Es un conjunto de datos de esquema de Winograd a gran escala, a diferencia del original con 44 mil ejemplos (Sakaguchi et al., 2021). Fue recolectado a través de colaboración colectiva en Amazon Mechanical Turk[4]. El conjunto de datos es amplio y variado en cuanto al contexto de cada pregunta evitando contenido repetitivo (se provee del tópico o contexto). Posteriormente los autores evalúan el conjunto de datos a través de una segunda capa de colaboradores para asegurarse de que las preguntas son difíciles y al mismo tiempo no ambiguas para el ser humano. Estas medidas se tomaron para garantizar que no haya sesgo a nivel de instancia. Los autores también presentan el algoritmo de filtrado adversarial **AFLITE** y utilizan un modelo de lenguaje **RoBERTa** (Liu et al., 2019) ajustado (fine-tuned) produciendo así un WinoGrande sin sesgo con 12,282 instancias y un WinoGrande con sesgo (sin filtrar) con 40,938 ejemplos.

WinoFlexi. Este conjunto de datos es similar a WinoGrande. Se construye a través de colaboración colectiva (Isaak y Michael, 2019). Constituyen 135 pares de esquemas de Winograd, siendo en total 270 ejemplos. A diferencia de WinoGrande, los trabajadores colectivos eligen su propio tópico. A pesar de esto, los autores encuentran que los esquemas recopilados tienen una calidad decente lograda a través de la supervisión manual entre trabajadores.

Cada marco de referencia (benchmark) aporta distintas formas de evaluación de sentido común y razonamiento para la identificación de género, inferencias u oraciones complejas. Sin embargo, el problema del sentido común aún persiste. Elazar et al. (2021) argumenta que el éxito obtenido por los modelos del *estado-del-arte* en la solución de WSC y sus variantes es principalmente artefactual. Los autores demuestran en la Figura 1 que se necesita de grandes cantidades de instancias de entrenamiento

[4] https://www.mturk.com/

para realizar pequeñas mejoras en el conjunto de prueba, esto demuestra la ineficacia de grandes conjuntos de entrenamiento para adquirir habilidades de razonamiento de sentido común.

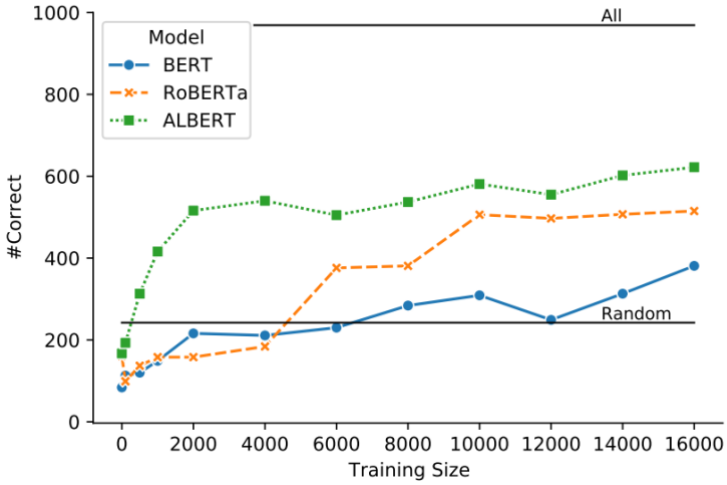


Fig. 1. Curva de aprendizaje para los modelos de lenguaje BERT, RoBERta y ALBERT (tomado de Back to Square One: Artifact Detection, Training and Common sense Disentanglement in the Winograd Schema (p. 8) por Elazar et al., 2021, arXiv preprint arXiv:2104.08161).

3 Métodos para la resolución de pronombres

Existen principalmente tres enfoques para la resolución de correferencias y ambigüedad de pronombres en los WSC y sus variantes.

Métodos basados en características. En este enfoque se plantean métodos que recolectan conocimiento del mundo formando reglas explícitas basadas en lógica a partir de bases de conocimiento, así como búsquedas por internet, sistemas basados en lógica y optimización para deducir la respuesta correcta.

Rahman y Ng (2012) presentan un modelo para la resolución de pronombres en WSC basado en aprendizaje en la Web (búsquedas en Google) para la adquisición de conocimiento. Emplean un enfoque basado en clasificación y entrenan su conjunto de datos usando SVM basado en rangos. Obtienen una precisión de 73,05%.

Bailey et al. (2015) establecen un marco para el razonamiento acerca de correlaciones entre oraciones con el objetivo de justificar las soluciones a los WSC. Argumentan la necesidad de un cálculo de correlaciones para el estudio de la coherencia en discursos.

Peng et al. (2019) constituyen un esquema de predicado que es instanciado con conocimiento adquirido de manera automática. Para manejar las inferencias ocupan programación lineal entera con restricciones. Evalúan su modelo en Winograd (con

76.41% en precisión) y WinoCoref (con 89.32% en AntePre). AntePre es una métrica propuesta por los autores.

Modelos neurales. Por otra parte el problema de WSC se ha abordado a través del uso de redes neuronales, así como redes neuronales profundas, sin el uso explícito de modelos de lenguaje pre-entrenados. La diferencia recae en que los modelos de lenguaje utilizan redes neuronales pre-entrenadas, mientras que los modelos neuronales se construyen desde el principio.

Uno de los primeros trabajos en presentar un modelo de redes neuronales para WSC fue propuesto por Liu et al. (2017a) donde introducen un modelo de asociación neuronal para modelar la causalidad y construyen automáticamente una gran colección (alrededor de 500 000) de pares causa-efecto, que se utilizan para entrenar el modelo. Los autores alcanzan una precisión del 70% para 70 esquemas de causa-efecto en WSC273. Posteriormente en su trabajo posterior Liu et al. (2017b) se extienden el modelo desarrollando sus propias incrustaciones de palabras pre-entrenados, cuya similitud semántica se correlaciona con las parejas de causa-efecto. Entrenan el modelo final en el conjunto de datos Ontonotes para resolución de correferencia (Hovy et al., 2006) alcanzando una precisión de 58.3% en PDP y 52.8% en WSC273.

Zhang y Song (2018) proponen una representación basada en distribución para capturar sentido común. Aumentan las incrustaciones de palabras que toma ventaja en las dependencias de las oraciones. El modelo que presentan es sin supervisión y no es entrenado con datos etiquetados. El enfoque presentado se prueba en un conjunto seleccionado manualmente de 92 esquemas fáciles de Winograd del conjunto de datos WSC273, logrando una precisión del 60,33%.

Modelos de lenguaje. En esta aproximación se emplean modelos de lenguaje entrenados en grandes corpus de texto. Muchos de los trabajos toman como base a BERT (Devlin et al., 2019) siendo un modelo de lenguaje de pre-entrenamiento por definición. La mayoría de las aportaciones se enfocan en realizar un mejor ajuste de parámetros (fine-tuning) en tales modelos y no en desarrollar nuevas arquitecturas.

Trinh y Le (2018) son los primeros en utilizar un modelo de lenguaje pre-entrenado. El modelo es entrenado en sobre un número masivo y diverso de corpus textual. Implementado como una arquitectura de Long Short-Term Memory (LSTM) y obteniendo resultados en WSC273 de 63.74% en precisión y para el problema de desambiguación de pronombres (PDP) un resultado de 70% en precisión.

Prakash et al. (2019) extienden este método con búsqueda de conocimiento (knowledge hunting) la cual se extrae de una fuente externa al modelo. Al igual que en Trinh y Le (2018) asignan probabilidad a las entidades (sustantivos) en WSC para la resolución de pronombres. Obtienen un resultado de 71.06% en precisión para WSC273 y 70.17% en precisión para WSC285.

Sakaguchi et al. (2020) ocupa un modelo de lenguaje RoBERTa (una variante de BERT) para resolver WinoGrande sin modificar ninguna capa de atención en el modelo. Entrenando sobre WinoGrande obtienen una precisión de 79.1% para este mismo conjunto de datos. Para WSC273 obtienen 90.1% en precisión, 87.5% en PDP, 85.6% en WNLI y 93.1% en DPR. Los autores afirman obtener una mejoría de nivel aleatorio en el conjunto de validación de WinoGrande cuando se entrena sobre WinoGrande sin sesgo.

4 Conclusiones

El desarrollo de un modelo o arquitectura de razonamiento y conocimiento de sentido común es aún necesario para generalizar el sentido común en este tipo de problemas de lenguaje. Kocijan et al. (2022) establece que la comprensión del sentido común en los sistemas de IA sigue siendo tan crucial como siempre. En este sentido los autores dan un ejemplo de carencia de generalidad. Finalmente, extrapolando la carencia de sentido común en los sistemas de IA, el cual no solo gira alrededor del lenguaje natural. Los sistemas de visión por computadora que reconocen escenas a partir de aprendizaje estadístico carecen de la comprensión del entorno, los sistemas de robots carecen de la comprensión del mundo donde interactúan, así como los traductores automáticos no pueden distinguir el contexto entre palabras como: *"fast food"* o *"quick food"*. Estos son solo algunos ejemplos reales de la necesidad de incorporar y modelar el sentido común artificialmente.

En lo referente a la resolución de pronombres, los modelos de lenguaje no consiguen resultados significativos para algunos conjuntos de prueba como lo es WinoGrande (Sakaguchi et al., 2021), e incluso para WSC273 obtienen únicamente resultados de 90% en precisión para subconjuntos de WSC, mas no para el conjunto de datos completo. Aunado a esto, aún falta por explorar si la ausencia o falta de generalidad en el sentido común se debe a los conjuntos de datos o a los métodos empleados para su adquisición. Es por esto, la necesidad de una búsqueda de arquitectura neural con la capacidad de generalizar el sentido común.

Referencias

1. Bailey, D., Harrison, A. J., Lierler, Y., Lifschitz, V., y Michael, J. (2015, March). *The winograd schema challenge and reasoning about correlation.* In 2015 AAAI Spring Symposium Series.
2. Davis, E. (2014). *Representations of commonsense knowledge*. Morgan Kaufmann.
3. Schank, R. C., y Abelson, R. P. (2013). *Scripts, plans, goals, and understanding: An inquiry into human knowledge structures*. Psychology Press.
4. Devlin, J., Chang, M. W., Lee, K., y Toutanova, K. (2018). Bert: *Pre-training of deep bidirectional transformers for language understanding*. arXiv preprint arXiv:1810.04805.
5. Elazar, Y., Zhang, H., Goldberg, Y., y Roth, D. (2021). *Back to Square One: Artifact Detection, Training and Commonsense Disentanglement in the Winograd Schema*. arXiv preprint arXiv:2104.08161.
6. Gunning, D. (2018). *Machine common sense concept paper*. *arXiv preprint* arXiv:1810.07528.
7. Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., y Weischedel, R. (2006). *OntoNotes: the 90% solution*. In Proceedings of the human language technology conference of the NAACL, Companion Volume: Short Papers (pp. 57-60).
8. Isaak, N., y Michael, L. (2019). *WinoFlexi: a crowdsourcing platform for the development of Winograd schemas*. In Australasian Joint Conference on Artificial Intelligence (pp. 289-302). Springer, Cham.
9. Kocijan, V., Davis, E., Lukasiewicz, T., Marcus, G., y Morgenstern, L. (2022). *The Defeat of the Winograd Schema Challenge*. arXiv preprint arXiv:2201.02387.

10. Levesque, H., Davis, E., y Morgenstern, L. (2012). *The winograd schema challenge*. In Thirteenth international conference on the principles of knowledge representation and reasoning.
11. Lenat, D. B. (1995). *CYC: A large-scale investment in knowledge infrastructure*. Communications of the ACM, 38(11), 33-38.
12. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... y Stoyanov, V. (2019). *Roberta: A robustly optimized bert pretraining approach*. arXiv preprint arXiv:1907.11692.
13. Liu, Q., Jiang, H., Evdokimov, A., Ling, Z. H., Zhu, X., Wei, S., y Hu, Y. (2017a). *Cause-Effect Knowledge Acquisition and Neural Association Model for Solving A Set of Winograd Schema Problems*. In IJCAI (pp. 2344-2350).
14. Liu, Q., Jiang, H., Ling, Z. H., Zhu, X., Wei, S., y Hu, Y. (2017b). *Combing context and commonsense knowledge through neural networks for solving winograd schema problems*. In 2017 AAAI Spring Symposium Series.
15. McCarthy, J. (1959). *Programs with common sense*. Cambridge, MA, USA: RLE and MIT computation center. pp. 300-307.
16. McCarthy, J., & Hayes, P. J. (1981). *Some philosophical problems from the standpoint of artificial intelligence*. In Readings in artificial intelligence. Morgan Kaufmann. pp. 431-450.
17. Minsky, M. (1974). *A framework for representing knowledge*.
18. Minsky, M. (1992). *Future of AI technology*.
19. Peng, H., Khashabi, D., y Roth, D. (2019). *Solving hard coreference problems*. arXiv preprint arXiv:1907.05524.
20. Prakash, A., Sharma, A., Mitra, A., y Baral, C. (2019, July). *Combining knowledge hunting and neural language models to solve the Winograd schema challenge*. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (pp. 6110-6119).
21. Rahman, A., y Ng, V. (2012). *Resolving complex cases of definite pronouns: the winograd schema challenge*. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (pp. 777-789).
22. Rudinger, R., Naradowsky, J., Leonard, B., y Van Durme, B. (2018). *Gender bias in coreference resolution*. arXiv preprint arXiv:1804.09301.
23. Sakaguchi, K., Le Bras, R., Bhagavatula, C., y Choi, Y. (2020, April). *Winogrande: An adversarial winograd schema challenge at scale*. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 34, No. 05, pp. 8732-8740).
24. Sakaguchi, K., Bras, R. L., Bhagavatula, C., y Choi, Y. (2021). *WinoGrande: an adversarial winograd schema challenge at scale*. Communications of the ACM, 64(9), 99-106.
25. Sap, M., Shwartz, V., Bosselut, A., Choi, Y., & Roth, D. (2020). *Introductory tutorial: Commonsense reasoning for natural language processing*. Association for Computational Linguistics (ACL 2020): Tutorial Abstracts, 27.
26. Schank, R. C., y Abelson, R. P. (2013). *Scripts, plans, goals, and understanding: An inquiry into human knowledge structures*. Psychology Press.
27. Tandon, N., Varde, A. S., y de Melo, G. (2018). *Commonsense knowledge in machine intelligence*. ACM SIGMOD Record, 46(4), 49-52.
28. Tirumala, S. S. (2020). *Artificial Intelligence and Common Sense: The Shady Future of AI*. In Advances in Data Science and Management (pp. 189-200). Springer, Singapore.
29. Trinh, T. H., y Le, Q. V. (2018). *A simple method for commonsense reasoning*. arXiv preprint arXiv:1806.02847.
30. Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., y Bowman, S. R. (2018). *GLUE: A multi-task benchmark and analysis platform for natural language understanding.* arXiv preprint arXiv:1804.07461.

31. Zhao, J., Wang, T., Yatskar, M., Ordonez, V., y Chang, K. W. (2018). *Gender bias in coreference resolution: Evaluation and debiasing methods*. arXiv preprint arXiv:1804.06876.
32. Zhang, H., y Song, Y. (2018). *A distributed solution for winograd schema challenge*. In Proceedings of the 2018 10th International Conference on Machine Learning and Computing (pp. 322-326).

Capítulo 10. Generador de frases estructuradas por medio de algoritmos genéticos, estructuras priónicas y estructuras proteínicas

César Zárate[1], Belém Priego[1], David Pinto[2]
[1]Universidad Autónoma Metropolitana unidad Azcapotzalco, Departamento de Sistemas, CDMX, México
[2]Benemérita Universidad Autónoma de Puebla, Facultad de Ciencias de la Computación, Pue, México
e-mail autor por correspondencia. al2203803198@azc.uam.mx

Resumen. La finalidad de este trabajo es presentar una metodología que permita combinar las estrategias del Procesamiento del Lenguaje Natural con la capacidad de selección de un algoritmo genético para crear un generador de frases automático. Partiendo de textos en el idioma español, siendo un corpus de naturaleza literaria, se propone la aplicación de las técnicas comunes de extracción, etiquetado y un etiquetador de partes de la oración (PoS) de manera paralela con los procesos de búsqueda de soluciones que aportan los algoritmos genéticos. Las palabras obtenidas por medio de las técnicas del PLN son consideradas como aminoácidos; éstos alimentan el algoritmo genético y son recombinados según reglas sintácticas previamente configuradas dentro de su función de aptitud (fitness). Las mezclas efectuadas por el algoritmo genético generan hileras de aminoácidos que pueden ser tomadas como proteínas o priones. Como resultado se generan los priones que son interpretados como frases correctas desde su enfoque sintáctico, dejando el aspecto semántico para futuras investigaciones.

Palabras clave: Algoritmos genéticos, Reglas sintácticas, Generador de frases.

1 Introducción

El Procesamiento del Lenguaje Natural (PLN) es un subcampo de la lingüística, informática e Inteligencia Artificial (IA) que estudia la interacción entre las computadoras y los lenguajes humanos IBM (2021). También estudia como programar las computadoras para procesar y analizar grandes cantidades de datos escritos en lenguaje natural (LN). La IA, según McCarthy (2004), puede ser observada en seres vivos y en máquinas; la IA es la capacidad que tienen las máquinas para resolver tareas simulando hacerlas como un ser vivo, una de ellas es el procesamiento de texto que puede ser resuelta por el PLN, siendo una subrama de la IA, permite detectar sintagmas dentro de oraciones escritas para facilitar estos procesamientos.

El idioma español, puede ser procesado por máquinas, a las cuales se les debe incorporar algoritmos de apoyo que mejoren las tareas realizadas por el PLN. Algunos tipos de algoritmos que pueden realizar esta función son los algoritmos evolutivos

(AE)[1] los cuales tienen la capacidad de encontrar soluciones de manera rápida por medio de métodos de búsqueda evolutiva. Dentro de ellos se encuentran los algoritmos genéticos (AG) que fueron concebidos por John Holland (1970) (Deb, 2001) y que simulan el comportamiento de la evolución biológica a través de un modelo matemático. Los AG tienen la virtud de encontrar soluciones dentro de grandes espacios de datos, de manera muy rápida y coordinada con un objetivo.

La estructura de este artículo está compuesta de 5 secciones las cuales se describen a continuación, en la Sección 2 se muestra el estado del arte, aquí se presentan los trabajos relacionados con el lenguaje natural, el uso de prefijos, AGs, el uso del PLN, el uso del modelo Job Shop Scheduling (JSS) y generación de slogans con AG; todos fueron comparados con este artículo. En la Sección 3 se muestra la metodología propuesta, la cual consiste en el procesamiento de textos en español por medio del PLN utilizando extracción, tokenización, etiquetados POS para la generación de frases con ayuda de un replicador de ADN y un AG el cual utiliza reglas gramaticales. Con el propósito de generar algunas frases, se utilizaron los formatos electrónicos de los trabajos (Octavio, 1950; Cortez y Vega, 2009). En la Sección 4 se muestran los resultados experimentales, donde se aplicó el PLN para obtener aminoácidos[2] útiles para el AG, y para las reglas gramaticales que tiene incorporadas se utilizó el software Weka y el algoritmo *farthestfirst.* Se revisaron frecuencias de ocurrencia de palabras y se constató que la Ley Zipf (Hernández Fernández et al., 2013) siempre está presente después de los procesamientos. En los experimentos se detectó que solo un determinado tipo de sintagmas como los adjetivos tienen una frecuencia de uso más alta que el resto de los sintagmas. En la Sección 5 se muestran las conclusiones y perspectivas obtenidas al aplicar los procesamientos con el PLN y los AG. Se comenta como la Ley Zipf[3] existe incluso aún después de todos los procesamientos y cómo es que determinados sintagmas tienen menor uso respecto de otros sintagmas en algunos textos.

2 Estado del arte

El Procesamiento del Lenguaje Natural es parte fundamental en la creación de frases, como se menciona en el trabajo de Cortez y Vega (2009) se modelan algunas partes que componen al LN, las cuales sirven en las tareas lingüísticas. Además, describe cómo es que puede ser aplicada la sintaxis y la semántica en el PLN. También habla sobre el lenguaje de programación y cinco niveles que permiten aterrizar el PLN a un nivel técnico. Todas estas relaciones permiten observar que el lenguaje puede ser

[1] Los algoritmos evolutivos (Tan, 2018) son un conjunto de algoritmos basado en procedimientos biológicos; dentro de este grupo se encuentran los algoritmos genéticos.

[2] Compuestos orgánicos con un grupo carboxilo (ácido) y otro amino (básico). La mayor parte de los aminoácidos naturales llevan ambos grupos, amino (NH2) y carboxilo (COOH) unidos a un mismo átomo de carbono en posición α. Estos pueden ser unidos por medio de puentes de hidrógeno para formar proteínas.

[3] Ley empírica formulada por George Kingsley Zipf (1940) que indica la frecuencia de aparición de distintas palabras en una determinada lengua, su función es $Pn \sim 1/n^a$ donde P es la frecuencia, n la n-ésima palabra y a un exponente real positivo.

procesado y programado con las estructuras mencionadas. Uno de los objetivos en este trabajo de investigación es el poder generar un algoritmo que permita abstraer todas estas partes para poder procesarlas.

Para programar el procesamiento del lenguaje, se debe tener en cuenta que dentro del lenguaje español se encuentran muchas partes sintácticas complementarias a las frases; algunas de éstas pueden ser los prefijos y los sufijos. En el trabajo Felíu (2002) se hace mención de dichas partes, las cuales son analizadas enfocándose en el impacto que tienen sobre el significado de una palabra; también se utiliza la hipótesis de Chomsky (1970) que se centra en las palabras complejas y en el contenido semántico.

Este documento usa como referencia reglas gramaticales para realizar el análisis al igual que en nuestro artículo, en donde nos enfocamos en la sintaxis y en las reglas gramaticales que puedan ser utilizadas por la IA para encontrar las palabras que generan menos contexto, es decir, las de mayor frecuencia. Al extraer frases se pueden tener distintos tipos de texto, como los literarios tal como se menciona en el artículo (Wedemann Roseli S. Moreno Jiménez Luis-Gil, 2020), el cual se centra en la generación de frases literarias particularmente para el español; aquí se indica que los textos literarios son más complejos que otros géneros, porque tratan sobre situaciones imaginarias o alegóricas, al igual que nuestro artículo, dicho trabajo también busca generar frases no presentes en el texto. Además, se menciona que, para crear textos, el hombre se vale de la creatividad y que existen tres tipos de creatividad: la combinatoria, la exploratoria y la transformacional, rasgos que las máquinas no tienen pero que pueden ser incorporados en un algoritmo para simular su comportamiento, para nuestro trabajo la parte combinatoria y exploratoria es la que nos interesa. Otro modelo importante que aplica la idea de distribución de trabajos asignados a máquinas de una tienda es el Job Shop Scheduling (JSS) (Miguel, 2018), donde utiliza a los AG para generar dicha distribución de la forma más equitativa posible; el beneficio es que los trabajos son distribuidos siempre de maneras distintas, esta forma de operar puede ser trasladado a la gramática, permitiendo generar por segmentos frases siempre diferentes.

En general el JSS consiste en encontrar soluciones que permitan distribuir tareas en una línea de trabajo dentro de una tienda, donde ninguna de las tareas por resolver en las líneas de trabajo debe ser iguales entre sí y cada trabajo debe ser realizado solo una vez por una sola máquina. Además, el tiempo total en cada línea de trabajo debe ser segmentado, para que se distribuya adecuadamente. Ya que nuestro artículo se centra en creación de frases, este modelo puede ser ajustado a la gramática si las líneas de trabajo son reemplazadas por hileras de aminoácidos y las máquinas que resuelven las tareas son cambiadas por aminoácidos.

Un artículo de gran aportación (Žnidaršič, 2015) en el cual se busca generar slogans sin intervención humana para compañías, para conseguir dicho efecto se basa de combinatorias, además su método está basado en un AG y de recursos lingüísticos. Uno de los recursos lingüísticos utilizados en dicho artículo es el etiquetado POS, las entradas que se procesan son obtenidas de un corpus a manera de lluvia de ideas, además centran su atención en los recursos semánticos que son de gran ayuda. Mientras que su enfoque está dirigido solo a slogans, el nuestro está centrado en cualquier clase de frases, debido a esto la combinatoria expuesta es distinta; en nuestro caso las frases

son creadas por medio de reglas gramaticales basadas en la sintaxis, mientras que sus slogans son creados por la combinatoria de otros slogans basados en la semántica. Todos estos artículos forman parte de la investigación que nos permitió formular las adaptaciones gramaticales y bioinformáticas de este artículo.

3 Metodología

Este trabajo se centra en la generación de frases obtenidas a partir de un conjunto de palabras adquiridas de un texto digital en el idioma español; hace uso de los procedimientos del PLN para poder extraer y procesar las palabras que son necesarias en la formación de estas frases. Los procedimientos del PLN que se utilizaron fueron la extracción de texto, la tokenización[4] y el etiquetado POS[5]. La primera etapa consiste en aplicar el pipeline[6] para el PLN, y estos son sus componentes:

1. **Extracción de palabras**. Para la extracción de palabras se utilizó Python con su herramienta pdfplumber, y el resultado es vaciado en un archivo para continuar con la tokenización, cada ítem se considera una línea de aminoácidos.
2. **Tokenización**. Con ayuda de la herramienta NLTK y RE de Python se realiza la exclusión de símbolos especiales como signos de puntuación, cifras, símbolos que no sean vocales o consonantes, acentos y todo aquel ítem que impida un etiquetado de palabras.
3. **Etiquetado**. Los ítems tokenizados en el paso anterior, son revisados por medio de Python con la herramienta treetaggerwrapper, el PoS permite separar las palabras en triadas palabra-raíz-sintagma[7], lo cual genera bolsas de palabras que son utilizadas posteriormente por el AG, el etiquetado es aprovechado por un subproceso que separa las palabras en genes y en puentes de hidrógeno.

Terminado este pipeline, las palabras pre-procesadas son enviadas como entradas al AG. La segunda etapa consiste en introducir las bolsas de palabras al AG para que este pueda construir frases de manera secuencial, el AG genera mezclas de aminoácidos por medio de reglas gramaticales incorporadas en su función de aptitud. Las etapas para generar las frases se muestran en la Figura 1. Estos son los componentes del AG:

[4] La tokenización consiste en la separación de los fragmentos de texto en unidades más pequeñas, existen 3 tipos tokenización: de palabra, de carácter y de sub-palabra.
[5] Part-of-Speech (POS) es el proceso de asignar a cada una de las palabras de un texto su categoría gramatical y la raíz de la palabra, y es ampliamente usado en lingüística computacional.
[6] Una serie de procesos que se realiza de manera secuencial.
[7] Un sintagma es una palabra o conjunto de palabras que se articula en torno a un núcleo y ejerce una función sintáctica; un sintagma puede ser adjetival, adverbial, conjuntivo, interjectivo, nominal, preposicional, pronominal, verbal.

- **Codificación inicial.** Que se relaciona con los aminoácidos de entrada contenidos en las bolsas de palabras y una matriz cuadrada numérica la cual es considerada una matriz de genes, cada fila es una hilera de aminoácidos. Se crea una población inicial aleatoria que se entrega a la cruza.
- **Cruza porcentual de genes padres.** De manera arbitraria se toman dos hileras de aminoácidos contenidas en la matriz de genes y se mezclan para generar la nueva población de hijos. Esta mezcla consiste en partir cada hilera de genes desde un punto de cruce aleatorio para juntarlo con otro segmento de gen de otro padre.
- **Mutación múltiple de genes hijos.** Algunos pares de aminoácidos de cada hijo generado son intercalados múltiples veces para crear pequeñas modificaciones en el gen hijo.
- **Fitness.** Entre el paso de mutación y selección se aplica el fitness del AG, el cual agrega el modelo JSS para segmentar los genes en bloques y detectar cuantas repeticiones de aminoácidos que se tienen por gen. En este punto, se busca que la función objetivo Z tenga un valor que converja al 0 para poder minimizarla.
- **Selección de hijos por elitismo.** De la matriz de genes hijos mutados que fue evaluada por medio de su fitness, se toma uno de los individuos más fuertes, este individuo se reserva en otra población para unirlo con los PHs.

Los individuos obtenidos por el AG son unidos unos contra otros por medio de PHs[8] los cuales son considerados sintagmas gramaticales, estos se obtienen de bolsas de palabras generadas en el etiquetado PoS. Un replicador de ADN se encarga de hacer estas uniones, dentro de dicho replicador existe una secuencia de unión que indica la regla gramatical que se está utilizando; en este artículo se propone la forma gramatical: *gen1 + ph1 + gen2 + ph2 + ph3 + gen3*, en donde los sintagmas de cada gen son preposiciones, sustantivos y verbos, así mismo, los de cada PH son pronombres, adjetivos y prefijos respectivamente. Esta regla gramatical puede cambiar a criterio si se desea formar otro tipo de frases, asignando otros sintagmas a cada gen y a cada PH. Finalmente, se pasa por un proceso de validación de priones[9] y proteínas. Este análisis, a pesar de ser semántico, también involucra su esencia sintáctica, la cual es de interés en este artículo.

[8] Un puente de hidrógeno es generado por medio de un enlace química a partir de la atracción existente de un átomo de hidrógeno y un átomo de oxígeno, las proteínas requieren de estos enlaces para permanecer unidas.

[9] Partícula infecciosa de naturaleza proteica que tiene la capacidad de transformarotras proteínas celulares normales en priones anómalos y que se encuentra en el origen de algunas enfermedades degenerativas del sistema nervioso central.

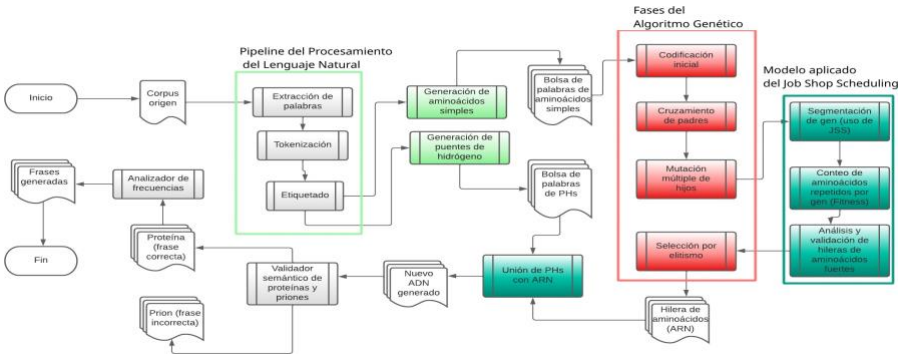


Fig. 1. Metodología propuesta para el procesamiento de palabras, transformación de palabras en aminoácidos y creación de nuevas frases.

Las primeras pruebas experimentales se realizaron con los trabajos de Jiménez-Carrión, 2018; Cortez y Vega, 2009). Estos son textos escritos en idioma español y de naturaleza literaria. El corpus del primer trabajo está dividido en capítulos que permiten extraer de manera parcial las palabras. Existen párrafos con longitudes variables que incluso pueden contener hasta 93 palabras antes de llegar a un salto de línea; comúnmente, un párrafo contiene palabras relacionadas a un contexto y al momento de que el AG las toma y las mezcla puede romper este contexto para crear otras frases que generen otro contexto nuevo. Para el caso del segundo corpus los párrafos son más cortos, por ello el número de palabras que se obtienen son menos. Se espera que la cantidad de priones que se puedan obtener de este tipo de corpus sea mayor que en el primero, lo que deriva en una menor probabilidad de encontrar proteínas, las cuales son frases más coherentes. Tomando las líneas de texto obtenidas anteriormente, se cuenta el número de veces que se repite cada una de las palabras. El análisis consiste en detectar que palabras existen en exceso en el corpus de entrada.

4 Resultados experimentales

En este experimento se utilizó el corpus (Octavio, 1950). Después de ser procesado se midió la frecuencia de las palabras obtenidas, las cuales se muestran en la Tabla 1. Se tomaron 5000 genes como entrada lo que permitió evaluar manual- mente a las frases que fueron generadas, para verificar si en otros experimentos dichas frases vuelven a ser generadas. Al revisar otras muestras, las frases fueron contrastadas sin obtener repeticiones.

Tabla 1. Repeticiones detectadas después de aplicación del AG.

Elemento Biológico	No. repeticiones	Clúster 1	Clúster 2	Clúster 3	Clúster 4
gen1	57, 53, 49 y 22	unas	con	durante	Sin
gen2	21,19, 11 y 13	catolicismo	felicidad	fertilidad	Gente
gen3	33, 19, 12 y 13	utilizarla	dejarlos	suprimido	Querer
ph1	87, 30, 37 y 38	ella	mi	sus	Su
ph2	5, 12, 5 y 3	valor	ojos	nuevas	Cielo
ph3	34, 25, 22 y 17	ambi	super	equi	Poli

Al realizar pruebas técnicas se observó que, con al menos 1000 genes, las muestras permitían detectar un aminoácido que tendía a repetirse muchas más veces que el resto. Al aumentar el tamaño de la muestra, el comportamiento era más notorio. Se decidió magnificar la muestra de 1000 genes de manera arbitraria 5 veces, lo que permitió observar el comportamiento de más aminoácidos de manera simultánea. El resultado fue siempre el mismo, solo un aminoácido era el que tendría a sobresalir respecto del resto.

Los experimentos se repitieron en diferentes ocasiones obteniendo el mismo comportamiento estadístico. Las gráficas fueron generadas por el software Weka; en cada una de ellas existe solo una palabra que tiende a sobresalir del resto. Debe mencionarse que las frases están formadas por genes y puentes de hidrógeno. En el primer experimento la regla que se siguió para mezclar las frases fue *gen1 + ph1 + gen2 + ph2 + ph3 + gen3*, siendo *gen(n)* el gen obtenido con los procesos del PLN y *ph(m)* el puente de hidrógeno que el AG intercala entre cada gen(n), donde m y n son las posiciones relativas de cada uno de estos elementos, siendo estos últimos números enteros positivos; los tipos de sintagmas que están relacionados con cada una de estas formas son p*reposición + pronombre + sustantivo + adjetivo + prefijo + verbo* respectivamente.

La tendencia a que un aminoácido se repita más que otros se manifiesta en cada uno de los sintagmas de las frases construidas por el AG. En este caso, el *gen1* tiene un máximo de 57 repeticiones para la palabra unas, y 53 para la palabra con. De igual forma, el número de elementos para el *ph2* es muy grande, Sea no puede trazarlo así que nuevamente se vuelve a revisar de manera manual con una hoja de cálculo, grafica que se muestra en la Figura 2, en donde es posible apreciar que el número máximo de repeticiones es de 12 con la palabra ojos, seguido de las palabras valor y nuevas, ambas con 5 repeticiones. En la Figura 2 se muestran los resultados de todas las gráficas de los sintagmas generados en este experimento, la cuales corresponden a los genes y PH *gen1, ph1, gen2, ph2, ph3 y gen3*, respectivamente.

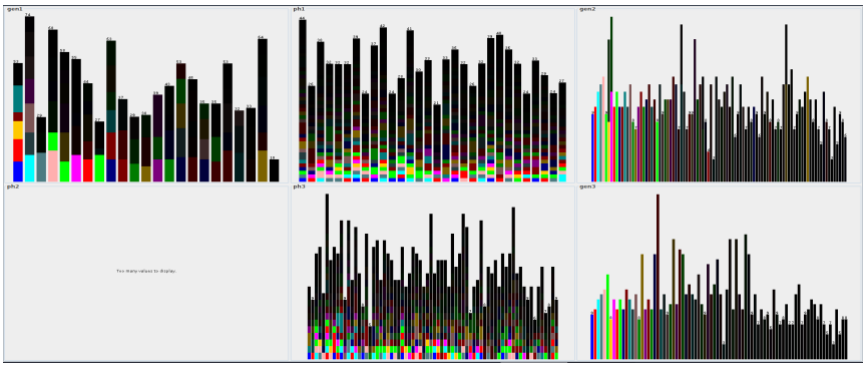


Fig. 2. Repeticiones de aminoácidos en todas las frases generadas.

La tendencia de repeticiones en el ph2 en diversos experimentos fue la misma. Dicho ph tiene una distribución más equitativa que el resto de los sintagmas, tal como se muestra en la Figura 3, lo cual indica que en el texto analizado predominan más los adjetivos, seguido de los sustantivos y los verbos.

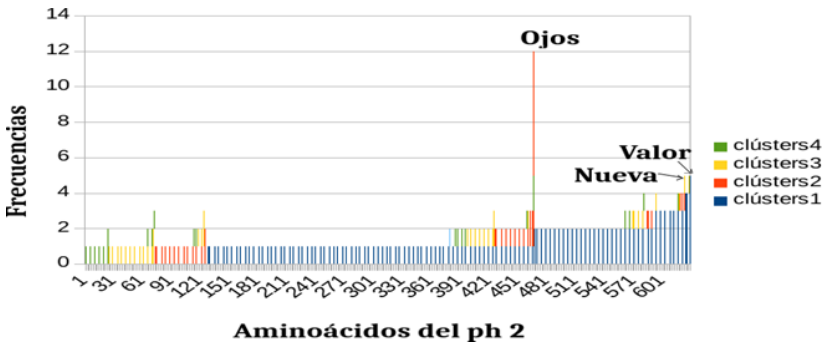


Fig. 3. Aminoácidos más repetidos del ph2.

En la Tabla 2 se muestran dos genes obtenidos por el AG, donde se aplicó el método de JSS para verificar si los genes contienen repeticiones, buscando minimizar la función objetivo Z. El AG encontró el gen 93, que es un gen fuerte porque su valor Z = 0, es decir, se tienen 0 repeticiones.

El segundo gen obtenido por el AG fue el 78, el cual tuvo un total de 3 repeticiones, su valor fitness fue de Z = 3, por ello es considerado un gen débil. Cada bloque obtenido por el método JSS es considerado un subX, los subX s que afectan el valor óptimo de Z para el gen 78 son [37, 60, 60, 79, 96], [32, 36, 36, 47, 59] y [1, 14, 14, 66, 72]. Por esta razón este gen fue descartado.

Tabla 2. Genes obtenidos por el AG y el modelo JSS aplicado.

Gen fuerte Individuo 93 [16, 30, 44, 52, 97], [9, 13, 30, 61, 85], [4, 5, 29, 69, 91], [13, 15, 19, 63, 76], [10, 23, 37, 47, 97], [3, 67, 81, 84, 86], [25, 44, 46, 50, 57], [2, 17, 69, 71, 85], [2, 4, 19, 43, 55], [30, 45, 69, 72, 98]
Gen débil Individuo 78 [3, 16, 25, 50, 61], [22, 47, 56, 79, 86], [28, 29, 45, 47, 60], [37, 60, 60, 79, 96], [26, 30, 43, 45, 89], [8, 35, 57, 70, 84], [23, 44, 60, 63, 70], [9, 15, 33, 35, 40], [1, 14, 14, 66, 72], [32, 36, 36, 47, 59]

Todas las hileras de la matriz de genes que existen dentro del AG fueron revisadas hasta que Z pudo converger a 0, luego estos genes fuertes fueron unidos con los PH siguiendo las reglas gramaticales descritas en la metodología, en la Tabla 3 se muestran los resultados de las proteínas y los priones obtenidos, así como algunas sugerencias para sus correcciones semánticas manuales.

Tabla 3. Frases generadas por el AG relacionadas con el gen 1 y ph 2.

Proteínas y priones generados con el gen1 y el ph2	
Proteínas	Priones
hasta su artificial bondad semi- recibir	hasta ellos mejor mejor ante empleadas
hasta el artificiales específicos neo- recibir	hasta ellas mejores organismos co- empleadas
hasta el mejor opciones pre- empleadas	hasta su mejor proba con
hasta sus artificiales problemas super- recibir	hasta le artificiales respecto archi
Con el ph2	
Con el gen1	
hasta sus artificiales bondades semi recibidas	hasta ellos mejor mejor ante empleados
hasta el artificial específico neo-recibo	hasta el mejor organismo co-empleado
hasta la mejor opción pre-empleada	hasta su mejor proba-con
hasta sus artificiales problemas super-recibidos	hasta los artificiales respectos archi
Con el ph2	
sobre sí acueducto xxxv auto-oliendo	consigo la bocacha cómica hipo enviada
entre nuestras francesas cómicas pluri-desgraciadas	uno tu furia cómica vice-expuesta
	durante ellas húmedas cómicas an-abrirse

5 Conclusiones y perspectivas

Se observa que, según la regla sintáctica impuesta en este experimento, al combinar el PLN con AGs, lo que más predomina en el texto revisado son los adjetivos, por ello la

diversidad de frases que el AG puede generar está enriquecida por los adjetivos. Los sustantivos y los verbos forman una parte importante para la generación de nuevas frases; el resto de los sintagmas siguen siendo importantes, pero aparecen con menor frecuencia en las frases generadas, tal como sucede con los pronombres. Las palabras que más se repiten pueden generar frases ineficientes, por ello se sugiere descartarlas siempre que sea posible.

Los resultados estadísticos de las frases generados por el AG demuestran que la ley Zipf no desaparece, incluso después de haber procesado todo el texto, lo que indica que la agregación de AG en el PLN es una estrategia que puede ser muy funcional para la generación de frases. Además, la incorporación de modelos adicionales como el JSS en el fitness del AG, mejora el comportamiento natural que ´este tiene, dándole un agregado a la metodología propuesta. Este artículo está centrado exclusivamente en la sintaxis de las oraciones creadas, automatizar la semántica queda como trabajo a futuro, dado su nivel de complejidad.

Referencias

1. Cortez A., Vega H., P. J. (2009). *Procesamiento de lenguaje natural*., 45–54. de Alarcón, P. A. (1874). El sombrero de tres picos.
2. Deb, K. (2001). *Multiobjetivo optimización usa evolucionar algoritmos*. John Wiley and Son.
3. Felíu, A. E. (2002). *La opacidad sintáctica de las palabras derivadas: una nueva perspectiva.*, 46.
4. Hernández Fernández, Antoni y Diéguez Vida, F. (2013). *La ley de Zipf y la detección de la evolución verbal en la enfermedad de Alzheimer*. Anuario de Psicología, 43, 17.
5. IBM. (2021). *Watt is natural language processing?* https://www.ibm.com/cloud/learn/natural-language-processing.
6. McCarthy, J. (2004). *What is artificial intelligence?*
7. Miguel, J. C. (2018, octubre). *Algoritmo gen ético simple para resolver el problema de programación de la tienda de trabajo (Job Shop Scheduling)*. Información tecnológica, 29 (5), 299–314. Descargado
8. Jiménez-Carrión, M. (2018). *Algoritmo Genético Simple para Resolver el Problema de Programación de la Tienda de Trabajo (Job Shop Scheduling)*. Información Tecnológica, 29(5), 299–314. doi:10.4067/s0718-07642018000500299
9. Octavio, P. (1950). *El laberinto de la soledad*. Random Housetrade.
10. Tan, Y. (2018). *Swarm intelligence vol 3 - applications*. Digital Library Theiet. y Wedemann Roseli S. Moreno Jiménez Luis-Gil, T. M. J. M. (2020). *Generación de frases literarias en español*. arXiv:2001.11381.
11. Žnidaršič, P. T. G. P. M. (2015). Using a genetic algorithm to produce slogans.

Editores

Dr. Juan Manuel González Calleros
Dra. Josefina Guerrero García
Dra. Claudia Zepeda Cortés
Dra. Darnes Vilariño Ayala

Avances de ingeniería del lenguaje, del conocimiento y la interacción humano máquina
Volumen II

Coordinado por
Dr. Juan Manuel González Calleros
Dra. Josefina Guerrero García
Dra. Claudia Zepeda Cortés
Dra. Darnes Vilariño Ayala
está disposición en la página
https://issuu.com/uajournals/docs
a partir de agosto de 2022