

DESARROLLO DE UN SISTEMA DE RECOMENDACIÓN
INTELIGENTE PARA SELECCIÓN DE PRENDAS UTILIZANDO
ANÁLISIS DE IMÁGENES CON REDES NEURONALES.

LAURA JULIANA PADILLA REYES
NATALIA BONIFAZ OVIEDO

UNIVERSIDAD SANTO TOMÁS
INGENIERIA ELECTRÓNICA
BOGOTÁ D.C

2021



DESARROLLO DE UN SISTEMA DE RECOMENDACIÓN
INTELIGENTE PARA SELECCIÓN DE PRENDAS UTILIZANDO
ANÁLISIS DE IMÁGENES CON REDES NEURONALES.

LAURA JULIANA PADILLA REYES
NATALIA BONIFAZ OVIEDO

Proyecto de grado presentado como requisito para optar al título de
INGENIERO ELECTRÓNICO

UNIVERSIDAD SANTO TOMÁS
INGENIERIA ELECTRÓNICA
BOGOTÁ D.C
2021

UNIVERSIDAD SANTO TOMÁS

Ingeniería Electrónica

Este proyecto de grado fue aceptado para la carrera de Ingeniero(a)
Electrónico(a) en Junio 2019

Director: Juan Manuel Calderón
Ing. Juan Manuel Calderón. Ph.D
Bogotá, Colombia

Co-director: Edgar Camilo Camacho Poveda
Magíster en Ingeniería Electrónica
Universidad Nacional
Bogotá, Colombia

Jurados: Dario Alejandro Segura Torres
Magíster en Ciencias de las comunicaciones y la información
Universidad Distrital
Bogotá, Colombia

José Guillermo Guarnizo Marín
Doctor en Automática, Robótica e Informática Industrial
Universidad Politécnica de Valencia
Valencia, España

Sustentación proyecto de grado: 5 Julio 2021, BOGOTÁ D.C

Laura Juliana Padilla Reyes _____

Natalia Bonifaz Oviedo _____

NOTA DE ACEPTACIÓN

*El trabajo de grado titulado **Desarrollo de un sistema de recomendación inteligente para selección de prendas utilizando análisis de imágenes con redes neuronales**, realizado por las estudiantes **Laura Juliana Padilla Reyes y Natalia Bonifaz Oviedo**, cumple con los requisitos exigidos por la **UNIVERSIDAD SANTO TOMÁS** para optar al título de **INGENIERO ELECTRÓNICO**.*

Juan Manuel Calderón

Edgar Camilo Camacho

Dario Alejandro Segura

José Guillermo Guarnizo

Declaración

Me permito afirmar que he realizado la presente tesis de manera autónoma y con la única ayuda de los medios permitidos y no diferentes a los mencionados en la propia tesis. Todos los pasajes que se han tomado de manera textual o figurativa de textos publicados y no publicados, los he reconocido en el presente trabajo. Ninguna parte del presente trabajo se ha empleado en ningún otro tipo de tesis.

Bogotá, D.C., 28.05.2021

Laura Juliana Padilla Reyes

Natalia Bonifaz Oviedo

Resumen

Palabras clave: Sistemas de Inteligencia y aplicaciones, Inteligencia artificial y aplicaciones, Deep Learning..

La digitalización en la industria de la moda está atrayendo la atención tanto de los consumidores como de los servicios de compra online. Por ello, cada vez es más importante contar con un sistema de recomendación personalizado y eficaz. Sin embargo, la mayoría de los sistemas tradicionales se centran en las recomendaciones sin tener en cuenta la relación entre la ropa y el usuario, lo que disminuye la precisión de las recomendaciones. Por ello, se propone un sistema de recomendación de moda basado en las preferencias del usuario. La capacidad de adaptación del sistema viene dada por dos fases. La primera genera una memoria a corto plazo que se actualiza constantemente con las interacciones del usuario. La segunda crea una memoria a largo plazo basada en una DNN. El sistema de recomendación está estructurado en 3 fases: Generador de Bases de Datos, Clasificación de Modelos y Perfilado Implícito. El Generador de Bases de Datos codifica las características visuales de las prendas. El Modelo de Clasificación se ocupa de la puntuación de las recomendaciones. El perfil implícito actualiza la clasificación según las preferencias del usuario. Por último, el sistema se evalúa utilizando imágenes proporcionadas por el usuario. A través de experimentos basados en la interacción con el usuario, el sistema demuestra su capacidad de adaptación al recomendar conjuntos similares a las selecciones anteriores del usuario. El sistema propuesto demostró la capacidad de ajustarse a las preferencias del usuario a través de las interacciones hombre-máquina, tal y como se requiere para este tipo de sistema de recomendación.

Abstract

Keywords: Intelligence Systems and applications, Artificial intelligence and application, Deep Learning..

Digitalization in the fashion industry is attracting the attention of both consumers and online shopping services. Therefore, a personalized and efficient recommendation system is becoming increasingly important. However, most of the traditional systems focus on recommendations without considering the outfit-user relationship, decreasing the accuracy of the recommendations. Therefore, we propose a fashion recommendation system based on user preferences. The adaptive capacity of the system is given by two phases. The first one generates a short-term memory that is constantly updated with the user's interactions. The second one creates a long-term memory based on a DNN. The recommendation system is structured in 3 stages: Database Generator, Model Ranking, and implicit profiling. The Database Generator encodes the visual characteristics of the garments. The Ranking Model deals with the scoring of the recommendations. The implicit profiling updates the ranking according to the user's preferences. Finally, the system is evaluated using images provided by the user. Through experiments based on user interaction, the system demonstrates adaptation capabilities by recommending similar outfits to the previous user selections. The proposed system demonstrated the ability to adjust to user preferences through human-machine interactions, as required for this type of recommendation system.

Declaración

Me permito afirmar que he realizado la presente tesis de manera autónoma y con la única ayuda de los medios permitidos y no diferentes a los mencionados en la propia tesis. Todos los pasajes que se han tomado de manera textual o figurativa de textos publicados y no publicados, los he reconocido en el presente trabajo. Ninguna parte del presente trabajo se ha empleado en ningún otro tipo de tesis.

Bogotá, D.C., 28.05.2021

Laura Juliana Padilla Reyes

Natalia Bonifaz Oviedo

Contenido

Resumen	VI
Abstract	VII
1. Introducción	2
2. Planteamiento Del Problema	4
3. Justificación	6
4. Impacto Social	8
5. Objetivos	10
5.1. General	10
5.2. Específicos	10
6. Marco Teórico	11
6.1. Aprendizaje Automático.	11
6.1.1. Aprendizaje Supervisado.	11
6.1.2. Aprendizaje No Supervisado	13
6.1.3. Aprendizaje por Refuerzo	14
6.2. Sistemas de recomendación	15
6.2.1. Elementos de un sistema de recomendación	16
7. Estado Del arte	17

8. Metodología	21
8.1. Revisión bibliográfica	21
8.2. Desarrollo práctico.	23
8.3. Verificación de resultados.	23
9. Diseño	24
9.0.1. Creación de Base de Datos	25
9.0.2. Generación de combinaciones	35
9.0.3. Modelo de ranking	36
9.0.4. Perfilamiento implícito	37
10.Experimentos	47
10.1. Experimento A	49
10.2. Experimento B	55
10.3. Experimento C	62
10.4. Experimento D	68
11.Resultados	74
Bibliografía	77

Capítulo 1

Introducción

El mercado de la moda ha estado en constante aumento, a tal punto que equivale a 406 mil millones de dólares estadounidenses [1]. Así mismo el mercado en línea se estima en 113 millones de dólares y crece anualmente un 18 % [2]. Dado el aumento del mercado de la moda y el crecimiento del mercado global online, es necesario generar sistemas que ayuden a optimizar la selección de productos de moda por parte de los usuarios. Del mismo modo, este sistema debería ayudar a optimizar los productos de moda que compran los usuarios y a reducir los costes de producción de las temporadas. Un sistema de recomendación basado en las preferencias del usuario ayudaría a la sociedad y a la economía global a centrarse en el nivel de inversión y a reducir la sobreproducción de artículos de moda. La producción del mercado de la moda podría ajustarse fácilmente a las preferencias del cliente en el mercado de la moda.

Un sistema de recomendación de prendas es un servicio emergente que tiene como objetivo recomendar prendas visualmente compatibles [3]. Sin embargo, la mayoría de sistemas entregan recomendaciones sin considerar la relación outfit-usuario, causando que la conexión inherente entre usuario y preferencias se ignore a lo largo del tiempo. Por esta razón, se propone un sistema de recomendación basado en aprendizaje profundo, con pesos ajustados por defecto a partir de un etiquetado profesional, para evitar el arranque aleatorio. El sistema propuesto garantiza

la compatibilidad entre prendas y la adaptación a las preferencias del usuario, utilizando memorias de largo y corto plazo. La adaptación del sistema se aborda a partir de dos fases de actualización. La primera fase realiza la actualización del puntaje de las recomendaciones cada vez que el usuario interactúa con el sistema. La segunda fase se encarga de actualizar los pesos de la *Deep Neural Network* (DNN), al completar n interacciones entre el usuario y el sistema. A partir de estas fases se tienen memorias de corto y largo plazo respectivamente, y de este modo el sistema considera las preferencias del usuario a lo largo del tiempo. Se presentan cuatro contribuciones diferentes. En primer lugar, un sistema de recomendación basado totalmente en la información visual. En segundo lugar, un conjunto de recomendaciones basado en la experiencia previa del usuario y en la elaboración de perfilamiento profesional. En tercer lugar, un sistema de recomendación adaptable a cualquier experiencia del usuario. Por último, un sistema de recomendación basado en la memoria a largo plazo y a corto plazo que interactúa para mejorar la experiencia del usuario.

Este trabajo de grado se desarrolló gracias al apoyo y soporte del Grupo de investigación y Desarrollo en Robótica de la Universidad Santo Tomas (G.E.D.). El grupo G.E.D. trabaja principalmente tres líneas de investigación enfocadas en Sistemas Inteligentes, Visión Artificial y Robótica. Dentro de estas líneas de investigación el grupo ha desarrollado diferentes proyectos de investigación enfocados a robótica cooperativa, robótica educativa, inteligencia artificial y robots humanoides. El grupo participó activamente de la iniciativa RoboCup en donde se desarrolló un equipo de fútbol de robots para la liga “Small Size” y actualmente se enfoca su participación en la liga “@Home” tal como se muestra en los trabajos [4], [5]. Se desarrollaron estrategias de control y lógica difusa para la reducción de la fuerza de impacto en robots humanoides cuando estos caen o saltan [6], [7], [8], [9]. Además han trabajado en el uso de la robótica para incentivar a estudiantes de colegio a optar por carreras afines a la matemática y la ingeniería [10], [11], [12]. Actualmente se desarrollan proyectos de investigación enfocados en la robótica de enjambre [13], [14], [15], [16] y sistemas de aprendizaje automático tal como [17], [18], [19], [20].

Capítulo 2

Planteamiento Del Problema

Cada vez más, la gente incrementa su interés por la forma en que se viste, debido a que la apariencia también es considerada una forma de expresión. Regularmente la gente se viste de manera consciente sobre su presencia ante otros, percibiendo que mediante el vestir la persona es interpretado e interpreta a los demás. Al tener en cuenta esta información se logra identificar que el vestuario trasciende de algo superfluo, a ser fundamental en el desarrollo personal y social de la persona [21].

Así mismo, el mercado de la moda en Colombia, representa el 9,4% sobre el total del PIB industrial y durante noviembre del año 2020 tuvo una inversión de 2,14 billones de pesos [22]. Durante el año 2019, la producción de prendas incrementó 3,1% con respecto al año 2018 [23].

Dado el aumento de la industria de la moda, es necesario diseñar sistemas que ayuden a optimizar la selección de productos de moda por parte de los usuarios. Ya que un sistema de recomendación basado en las preferencias del usuario ayudaría a la sociedad y a la economía a centrarse en el nivel de inversión reduciendo la sobreproducción de artículos de moda.

Sin embargo varias de las herramientas tecnológicas disponibles tales como *Stitch fix* [24] o *Project Muze*[25], tienen un enfoque únicamente comercial incitando al usuario a comprar, cosa que no todos están dispuestos a hacer. En otros casos las recomendaciones generadas no coinciden con gustos personales, o simplemente la base de datos con las que el sistema

genera la recomendación no coincide con lo que el usuario tiene disponible en su armario.

Teniendo en cuenta esta información, surge la inquietud: ¿Cómo diseñar un sistema de recomendación de prendas, a partir de un conjunto de imágenes previamente clasificadas y etiquetadas, teniendo en cuenta las preferencias del usuario?

Capítulo 3

Justificación

A partir del interés que gran parte de las personas muestran respecto a su forma de vestir se propone un sistema de recomendación basado en aprendizaje profundo. El sistema propuesto garantiza la compatibilidad entre las prendas y la adaptación a las preferencias del usuario, utilizando memorias de largo y corto plazo.

Este proyecto permitirá al usuario acceder a una experiencia personalizada que facilite la elección de la vestimenta, disminuyendo el tiempo que invierte en seleccionar las prendas. Del mismo modo, este sistema tiene la posibilidad de ayudar a optimizar los productos de moda que compran los usuarios y a reducir los costes de producción en las temporadas (de moda).

Un sistema de recomendación basado en las preferencias del usuario ayudaría a la sociedad y a la economía global a centrarse en el nivel de inversión, y a reducir la sobreproducción de artículos de moda. La producción del mercado de la moda podría ajustarse a las preferencias del cliente en el mercado de la moda. Junto con el alcance social y económico que llega a tener una herramienta de recomendación de prendas. También se destaca una relevancia tecnológica dentro del área de inteligencia artificial, orientado al aprendizaje automático y de clasificación [26][27]. Este campo es ampliamente utilizado para desarrollar sistemas que aprenden las características más importantes de datos de entrada, generando una salida que etiqüete de forma precisa o adecuada los datos.

Este proyecto se articula con la línea de investigación de inteligencia artificial, concretamente el sub-área de *Machine Learning* y *Deep Learning*. El proyecto implementa redes neuronales para identificar cada prenda con sus características, haciendo del proyecto viable debido al gran avance junto con estudios previos que se tiene en esta disciplina. Dentro del desarrollo de sistemas de recomendación también han surgido nuevas técnicas [28] [29] [30], popularizando los asistentes virtuales. Los asistentes virtuales son implementados en grandes plataformas como Amazon, Google, Apple y Microsoft, generando una gran inversión en la investigación y producción de estos agentes. Hoy en día existen sistemas de recomendación que se basan en los gustos de las personas [30]. También hay sistemas que identifican o distinguen objetos específicos en imágenes, los cuales presentan una recomendación sobre posibles combinaciones de ropa [31] [32]. No obstante, el sistema que se propone desarrollar, partirá de una base de datos personalizada que comprenda las prendas con que el individuo sienta mayor afinidad, para así, entrenarlo a partir de las decisiones adquiridas de ejemplos suministrados por el usuario.

Como resultado final del proyecto se generaría un motor de decisión enfocado únicamente a la persona. Este proyecto se destaca dentro de los sistemas de recomendación de prendas ya existentes debido que tiene en cuenta la individualidad del usuario en cada recomendación. Adicionalmente el desarrollo de este modelo destaca cuatro contribuciones diferentes. En primer lugar, un sistema de recomendación basado totalmente en la información visual. En segundo lugar, un conjunto de recomendaciones basado en la experiencia previa del usuario y en la elaboración de perfiles profesionales. En tercer lugar, un sistema de recomendación adaptable a cualquier experiencia de usuario. Por último, un sistema de recomendación basado en la memoria a largo plazo y la memoria a corto plazo que interactúa para la adaptación de las preferencias del usuario.

Capítulo 4

Impacto Social

“La moda expresa el espíritu del tiempo y es uno de los indicios más inmediatos de los cambios sociales, políticos, económicos y culturales” [33].

La forma de vestir es uno de los conceptos más difíciles de relacionar con el comportamiento humano, ya que los consumidores compran productos de moda más por su significado que por su utilidad [34] usando las prendas para enviar señales de su identidad [35]. Implementan la vestimenta para mejorar su imagen personal y darse a conocer de una manera en un entorno social, es por esto que adquiere un significado simbólico a través del proceso de socialización, [35] además es considerada ampliamente como un espejo de la sociedad, cultura y estado económico, a menudo interpretado como un tipo de “comunicación no verbal” con la sociedad [36]. Entwistle en su libro *El cuerpo y la moda: una visión sociológica* [37], propone la idea del vestir como una práctica en donde se comprende la relación entre el cuerpo, la ropa y la cultura, es decir que desde una perspectiva social, el vestir no sólo se trata de la prenda, también sobre la forma en que la ropa hace parte de nuestras actividades y se integra en nuestra cotidianidad [38][39]. De tal forma el modo en que la gente opta por vestirse se ha convertido en uno de los problemas más estudiados [39], intentando entender las complejidades de la moda, por sociólogos, psicólogos, y hasta economistas, dejando una mejor comprensión del porqué ciertos estilos han dominado en algunos puntos en el tiempo [40]. El sistema de recomendación ayudará a transformar la

percepción de cada usuario, considerando que vestir bien tiende a impulsar a ser más productivo y así mismo elevar la autoestima [41]. Es una necesidad humana fundamental, ya que de todos los juicios que formulamos en la vida el más importante es el que formulamos sobre nosotros mismos. Debido a que la autoestima viene siendo lo que inmuniza para no deprimirse, hundirse, o entre otras cosas, por esto, cuando esta es baja, hay menos recursos humanos para hacer frente a los retos de la vida (mundo intrapersonal, social, etc.) [41].

Capítulo 5

Objetivos

5.1. General

Implementar el motor de decisión para un sistema de recomendación sobre estilos de vestir basado en visión artificial y redes neuronales, que permita combinar prendas basado en la realimentación de los usuarios.

5.2. Específicos

- Clasificar y etiquetar de forma manual las imágenes de un conjunto de datos correspondiente a prendas de vestir, según su tipo de prenda.
- Diseñar un sistema basado en redes neuronales y visión artificial, capaz de clasificar de forma automática prendas de vestir, a partir del conjunto de datos clasificado y etiquetado.
- Diseñar un sistema capaz de combinar prendas de vestir a partir de realimentación de usuarios.

Capítulo 6

Marco Teórico

Para la realización y desarrollo de este proyecto es necesario conocer varios términos y conceptos que ayudaran al entendimiento e interpretación del mismo. A continuación, se introducirán dichos conceptos y términos necesarios tales como:

6.1. Aprendizaje Automático.

También conocido como *Machine Learning* o aprendizaje de máquinas, es una rama de las ciencias de la computación, o más específicamente de la inteligencia artificial, cuyo objetivo principal es dotar a las máquinas y equipos de cómputo la capacidad de aprender. Lo que se busca es generar métodos que permitan predecir comportamientos futuros usando información según la experiencia adquirida a partir de las decisiones o enseñanzas anteriores [42]. Los algoritmos implementados dentro del aprendizaje automático se pueden clasificar en diferentes grupos teniendo en cuenta como es abordado el problema de aprendizaje. El aprendizaje automático tiene como principales divisiones el aprendizaje supervisado y el aprendizaje no supervisado [43].

6.1.1. Aprendizaje Supervisado.

Este tipo de aprendizaje consiste en aprender a partir de un conjunto de ejemplos de entrenamiento, quiere decir, que se tiene datos con la

respuesta correcta, etiquetados por expertos, por tanto los algoritmos deben ser planteados para generalizar y de esta forma asignar correctamente la clase o etiqueta de pertenencia a toda entrada de la misma naturaleza.

Redes Neuronales Artificiales.

Las redes neuronales son un modelo el cual está inspirado en el funcionamiento de las neuronales del cerebro humano, consta de un conjunto de uniones o nodos capaces de procesar y enviar información entre ellos.

Los nodos se organizan en capas, estas suelen ser tres:

- Capa de entrada: Datos reales de entrada.
- Capa Oculta: Se denominan de esta forma, ya que no se conoce los valores de entrada y salida.
- Capa de salida: Es el resultado visible, se conoce el valor esperado y corresponde al valor asignado según la entrada.

En la capa de salida, se da las respuestas correctas, por tanto es una forma de aprendizaje supervisado, ya que el supervisor controla la salida de la red. En caso de que esta no coincida con la esperada, se procede a modificar los pesos de las conexiones, con el fin de conseguir que la salida obtenida se aproxime a la deseada. Se puede realizar diferentes tareas como de clasificación o regresión. La estructura básica de un RNA se puede modelar como se muestra en la Figura 6.1. Los datos de entrada generalmente se cargan en forma de un vector multidimensional a la misma capa de entrada la cual distribuye a las demás o a las capas ocultas. Estas toman decisiones a partir de la capa anterior y sopesan cómo un cambio estocástico dentro de la misma capa así perjudicando o mejorando el resultado final, esto se conoce como el proceso de aprendizaje. Tener múltiples capas ocultas apiladas unas sobre otras se llama comúnmente aprendizaje profundo o *Deep Learning* [44].

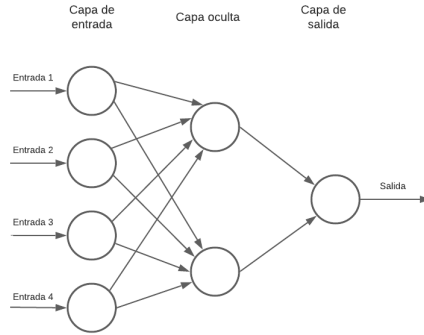


Figura 6.1: Red neuronal de tres capas

Redes Neuronales Convolucionales

Este tipo de redes se diferencia de las redes tradicionales, ya que son utilizadas principalmente en el campo del reconocimiento de patrones dentro de imágenes, estas permiten codificar características específicas de la imagen en la arquitectura, y al tiempo reduce aún más los parámetros necesarios para configurar el modelo [44].

En las redes convolucionales, CNN por sus siglas en inglés, se cuenta con filtros bidimensionales, este filtro es una neurona de las diferentes capas, la neurona convolucionaria con la entrada de la capa [45].

6.1.2. Aprendizaje No Supervisado

El objetivo en la mayoría de las formas de aprendizaje no supervisado es construir factores distintivos de las variables observadas, que realmente tanto las variables de entrada como las de destino, se pueden predecir [46]. en otras palabras en este tipo de aprendizaje el sistema recibe como entradas un conjunto de datos que no son clasificados correctamente, por tanto, el sistema trata de encontrar estructuras o patrones ocultos existentes en los datos de entrada para etiquetarlos [47].

A demás del aprendizaje supervisado y el aprendizaje no supervisado, existen otros tipos del aprendizaje automático. Un tercer paradigma, es el aprendizaje por refuerzo.

Agrupamiento por K-means

El agrupamiento es un tipo de análisis de aprendizaje no supervisado, por tanto no hay una respuesta correcta. El agrupamiento consta de la separación de los datos los cuales se agrupan teniendo en cuenta que dichos grupos se caracterizan porque los elementos tienen propiedades en común [48].

El algoritmo que más se implementa es el k -means, busca la separación de cierta cantidad de elementos en K grupos, para esto el algoritmo consta de cuatro etapas [49]:

1. Inicialización: Consta de la selección y localización de los centros en cada grupo k .
2. Clasificación: Cada elemento es asignado a un centro teniendo en cuenta la distancia, el elemento es clasificado en el centro más cercano y pasa a pertenecer al grupo del centro.
3. Cálculo de centros: Se realiza el cálculo de los centros de los grupos creados, ya que los centros iniciales fueron aleatorios.
4. Condición de convergencia: Esta condición se establece teniendo en cuenta el objetivo del agrupamiento, si llega no satisfacerse el procedimiento se reinicia hasta que los centros no cambien y la condición se cumpla.

6.1.3. Aprendizaje por Refuerzo

El aprendizaje por refuerzo o RL, *Reinforcement Learning*, es otro enfoque importante de aprendizaje automático que se basa en la intercomunicación del agente con el entorno. El agente aprende a comportarse a través del ensayo y error, de los resultados adquiridos de las acciones tomadas, es decir, aprende sobre la marcha o de la experiencia. Lo que se busca es asignar acciones a situaciones, con el fin de maximizar una señal de recompensa numérica [47]. El agente de aprendizaje no sabe previamente que acciones tiene que tomar, por lo que prueba cada acción y debe descubrir cuáles provocan la mayor recompensa

en diferentes estados posibles.

En este tipo de aprendizaje se puede identificar algunos términos principales, como[50]:

- Agente: Es el aprendiz, el que toma las decisiones y efectúa acciones.
- Entorno o medio ambiente: Es todo lo que esta fuera del agente, el mundo en el cual puede operar y se mueve.
- Estado: Es la ubicación actual del agente en el entorno.
- Acciones: Es el universo de todos los posibles movimientos y decisiones que el agente puede tomar.
- Señal de recompensa: Es un valor numérico generado por el entorno que sirve de retroalimentación al agente y permite determinar el éxito o fracaso de las acciones tomadas.
- Política: Es lo que define el comportamiento del agente en determinado estado.
- Función de valor: Es la estimación de la recompensa total acumulada en el futuro al tomar una acción en un estado dado, y sirve para determinar qué es lo que más le conviene al agente en largo plazo.
- Modelo del entorno: Permite realizar predicciones sobre cómo se comportará el entorno ante un estado y una acción para estimar el próximo estado y la próxima recompensa.

6.2. Sistemas de recomendación

Los sistemas de recomendación son una subclase de sistemas de filtrado de información que buscan predecir la valoración o preferencia que un usuario le daría a un elemento que aún no había considerado. Esto utilizando un modelo construido a partir de las características de un elemento (enfoques basados en el contenido) o el entorno social del usuario (enfoques de filtrado colaborativo) [51].

6.2.1. Elementos de un sistema de recomendación

Artículos

Elementos recomendados a los usuarios. Se le puede asignar un valor, el cual puede ser negativo lo que determina una decisión o positivo si es útil.

Usuarios

Puede tener diversos objetivos y características basadas en el escenario de la recomendación. Cualquier técnica que se utilice debe de ser dirigida al perfil del usuario, codificar todas las preferencias, necesidades y el comportamiento.

Transacciones

Información grabada entre un usuario y los sistemas de recomendación, se genera durante la interacción entre humano y el ordenador.

Capítulo 7

Estado Del arte

Los sistemas de recomendación son sistemas de filtrado que establecen la relación entre las recomendaciones y las preferencias del usuario. Este tipo de sistemas se utilizan en ámbitos como la moda y se denominan sistemas de recomendación de moda. El sistema de recomendación de moda debe garantizar dos requisitos básicos: la compatibilidad entre las prendas y la relación con las preferencias del usuario.

La compatibilidad entre las prendas se aborda desde la identificación y clasificación de las imágenes, ya que facilita el correcto emparejamiento entre las prendas. Algunos trabajos realizan la clasificación de imágenes a partir de las características visuales ya sea el color o el tipo de prenda. En el documento *Enhancing Visual Fashion Recommendations with Users in the Loop* [52], realizan la extracción de color en forma de un histograma HSV de K dimensiones, creando un vector que describe la prenda. Con el propósito de construir un sistema de recomendación a gran escala completamente autómata. Dentro de este trabajo también se propone una técnica basada en las preferencias del usuario, que a la vez reúne los resultados desde la “sabiduría de la multitud” para un análisis más profundo. Dentro de la investigación presentada en *Visually-Aware Fashion Recommendation and Design with Generative Image Models* [53] desarrollan un método de clasificación personalizada visualmente *Bayesian personalized ranking method* (DVBPR). DVBPR extrae simultáneamente características visuales y aprende los factores latentes del usuario. Otro

artículo que tiene en cuenta características visuales como el color es en *Large Scale Visual Recommendations From Street Fashion Images* [54]. En este artículo proponen la técnica *Texture Agnostic Retrieval (TAR)* basada en la noción de que los consumidores prefieren los colores sólidos que vayan bien con la ropa estampada. TAR básicamente evita tener prendas con patrones en la parte superior e inferior, ya que esto parece contra intuitivo.

La clasificación también se da estableciendo la relación entre un texto descriptivo y la imagen, como el modelo propuesto en *Aspect-based fashion recommendation with attention mechanism (AFRAM)* [55]. Dentro de AFRAM la palabra más importante del texto de entrada se asocia a la característica principal de la imagen, identificando así el tipo de prenda. También, dentro la identificación y clasificación de prendas es común el uso de *Convolutional Neural Networks (CNN)* [56]. En *Mining fashion outfit composition using an end-to-end deep learning approach on set data*, proponen un sistema híbrido entre redes CNN y *Recurrent Neural Networks (RNN)*. El sistema híbrido identifica la prenda, extrae las características visuales y las asocia al comportamiento del usuario. El artículo *Improving Outfit Recommendation with Co-supervision of Fashion Generation* [57] propuso un marco de aprendizaje de co-supervisión neural, llamado *Fashion Recommendation Machine (FARM)*. FARM mejora la comprensión visual, mediante la introducción de un nuevo mecanismo de capa a capa para fusionar la información estética de forma efectiva. Mediante amplios experimentos en dos conjuntos de datos disponibles públicamente demuestran que FARM supera a modelos más modernos por recomendación de atuendo, en términos de AUC y MRR. Los análisis detallados de los elementos generados y recomendados demuestran que FARM puede codificar mejor la información estética.

Por otro lado, para la relación entre los atuendos y las preferencias del usuario, se trabaja con algoritmos de agrupación y puntuación de las recomendaciones generadas. El agrupamiento y puntuación de recomendaciones se implementa para establecer patrones de acuerdo a las preferencias del usuario, generando un ranking de relevancia. La agrupación de recomendaciones se puede realizar según las características

de el ítem [58]. D. Verma, K. Gulati, y R. R. Shah, proponen una agrupación ponderada de características K-means, para que la descripción del grupo sea más detallada. En el artículo *Aesthetic-based clothing recommendation* [59] implementan *Bayesian personalized Ranking* (BPR) teniendo en cuenta lo que el usuario ha escogido y lo que no ha sido seleccionado, generando un ranking de recomendaciones organizado por puntajes. Otro método de agrupación se da calculando la diferencia entre elementos como se lleva a cabo en *Garment recommendation with memory augmented neural networks* [60], donde el resultado determina la pertenencia de una prenda a un grupo. Los patrones adquiridos con la agrupación y puntuación, pueden ser implementados en sistemas que se adaptan a las preferencias del usuario. En *Explainable fashion recommendation: A semantic attribute region guided approach* [3], se representa jerárquicamente la conexión usuario y atuendo, basándose en la interacción del usuario con el sistema. Así mismo, en el artículo *Multi-interest network with dynamic routing for recommendation at tmall* [61] utilizan el módulo *Fine-grained Preferences Attention* (FPA) para proyectar detalladamente la relación prenda-usuario. *Learning visual clothing style with heterogeneous dyadic co-occurrences* [62] representa las características de interés del usuario en vectores de entrada para el sistema. También, otros sistemas incluyen las preferencias del usuario teniendo presente información de compras realizadas anteriormente, como lo trabajan en *Learning Visual Clothing Style with Heterogeneous Dyadic Co-occurrences* [63]. En este caso, utilizan la información para hacer recomendaciones basadas en objetos similares, considerando la relación entre la prenda y el usuario. Para la función de transformación, utilizan una red neuronal convolucional siamés (CNN), donde los ejemplos de capacitación son pares de artículos compatibles o incompatibles. La compatibilidad la modelan basándose en la concurrencia escalar de datos sobre el comportamiento del usuario, en particular datos de compra de Amazon.com.

Así mismo se puede utilizar un contexto temporal como lo trabajan el proyecto por IBM y *The Fashion Institute of Technology* (FIT). El proyecto *Reimagine Retail* [64] toma diferentes señales como los colores y

estampados inspirados en los productos diseñados por Tommy Hilfiger, dando así la oportunidad de personalizar el tipo de prenda que se quiere ver en una tienda virtual, ya que provee un análisis en base a la personalidad del cliente, buscando una experiencia interactiva. En el artículo *POG: Personalized Outfit Generation for Fashion Recommendation at Alibaba iFashion* [65] proponen POG. POG es un modelo codificador-decodificador para generar atuendos de moda personalizados, que tienen en cuenta tanto la compatibilidad de los conjuntos y la personalización del usuario basados en los comportamientos recientes de los usuarios. Demostrando que POG supera claramente al método (Filtros Colaborativos) CF en un 70% en la métrica CTR *Click-through-Rate*.

Capítulo 8

Metodología

Teniendo en cuenta que el proyecto tiene un enfoque de investigación se plantea el diseño metodológico como se muestra en la Fig.8.1.

Tomando en consideración lo definido, se decide plantear la metodología plasmada en la Fig.8.1. Para llevar a cabo el desarrollo eficaz y satisfactorio del proyecto, teniendo en cuenta la gestión de la calidad en cada etapa. Requisitos de diferentes índoles se plantean, garantizando el cumplimiento de cada uno de los objetivos.

8.1. Revisión bibliográfica

Se realiza una búsqueda de diferentes estudios con temas afines al proyecto, con el objetivo de emplearlos como material bibliográfico. Esta investigación de artículos permite contextualizar el problema planteado, verificando y analizando avances previos de algoritmos que pueden ser utilizados como base teórica, y que fundamenten de manera sólida la intención.

1. **Elaboración y redacción del anteproyecto.**
2. **Definición y planteamiento del problema.** Se expone el tópico que se desea tratar, exponiendo el contexto social, y tecnológico de manera tanto global como local , para concluir con la pregunta final.

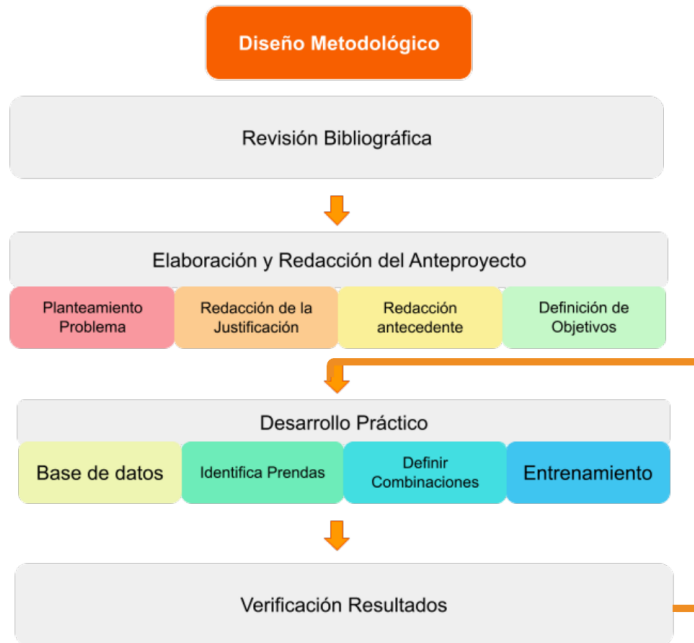


Figura 8.1: Diagrama de metodología

3. **Redacción de justificación.** Se plantea la razón del por qué se realiza este trabajo, presentando razones sociales, psicológicas y tecnológicas por los motivos que alegan el desarrollo del proyecto.
4. **Redacción de antecedentes.** Se presentan trabajos en temas afines que permitan contextualizar el proyecto, utilizándolos como medio para exhibir el estado en que se encuentra la investigación sobre sistemas de recomendación a nivel internacional.
5. **Administración del proyecto.** Se propone el cronograma de actividades, el cual se ordena y se estima la duración aproximada en el tiempo para la realización de las actividades propuestas, también se establece el presupuesto, en donde se considera la cantidad de dinero que se requiere para llevar a cabo el proyecto.

8.2. Desarrollo práctico.

En esta etapa se desarrolla el programa necesario para construir el motor de decisión del sistema, y se realizan pruebas de funcionamiento. Se subdivide en etapas específicas, que son:

1. **Creación Base de Datos.** Esta etapa consta de la creación de la base de datos utilizada como set de entrenamiento para el sistema. En este caso los datos son imágenes de prendas, etiquetadas según la clase a la que pertenecen.
2. **Identificación de Prendas** En esta etapa se diseña una red neuronal que tenga como entrada las imágenes de cada prenda, con el objetivo que identifique a que clase pertenece.

8.3. Verificación de resultados.

En esta etapa se analizan los resultados obtenidos de cada experimento realizado. Cada experimento valida el desempeño del sistema y su capacidad para adaptarse al usuario, brindando recomendaciones acordes con las preferencias del usuario. Los experimentos se basan en diferentes iteraciones entre el sistema con el usuario. Cada interacción intervendrá el usuario aceptando o rechazando los resultados, esté seleccionará las prendas según el gusto. Aunque haya una verificación con cierta cantidad de datos, se tiene en consideración, que al ser una evolución dinámica, los gustos de la persona pueden variar. Por esto, se tomarán estadísticas en las que se mostrará la disminución de errores, de este modo se sigue adaptando.

Capítulo 9

Diseño

El sistema de recomendación de prendas propuesto se divide en tres módulos: Creación de base de datos, Modelo del ranking y el perfilamiento implícito, tal y como se muestra en la Fig.9.1. En primer lugar, el módulo de la creación de base de datos propone un sistema de extracción y codificación de características de la imagen. Este módulo se encarga de transformar las imágenes en vectores que describen a cada una de las prendas.

La creación de base de datos se compone de un sistema basado en dos CNN, la primera reconoce el tipo de prenda y la segunda reconoce si es de estilo formal, o informal. También implementa un procesamiento de imagen que separa la prenda del fondo de la misma imagen, con la intención de extraer las características del color utilizando un algoritmo de agrupamiento $k - means$. Este módulo de creación de base de datos genera un descriptor de las prendas que recopila la información obtenida de las dos CNN y del procesamiento de la imagen. Dentro de este módulo se presentan 6 sub-procesos para cumplir con el objetivo de la extracción de características de la imagen satisfactoriamente.

Seguidamente, el módulo del modelo de ranking utiliza una DNN encargada de la clasificación de combinaciones para generar una tabla en la que se asocia un puntaje a cada recomendación que se genera, la cual es llamada *ranking table*.

Por último, el módulo del perfilado implícito genera un tablero de recomendaciones que se presentara al usuario según el modo de

recomendación del sistema, ya sea una recomendación abierta o una basada en una prenda preseleccionada por el usuario. También, dentro de este módulo se desarrolla un sistema de memoria basado en una DNN que se encarga de aprender de la *ranking table* después de las selecciones de moda del usuario cada cierto periodo de tiempo. El módulo de perfilado implícito presenta una memoria a corto plazo que se adapta diariamente con cada interacción con el usuario y un sistema de memoria a largo plazo que se adapta en función de las modificaciones realizadas en la *ranking table*.

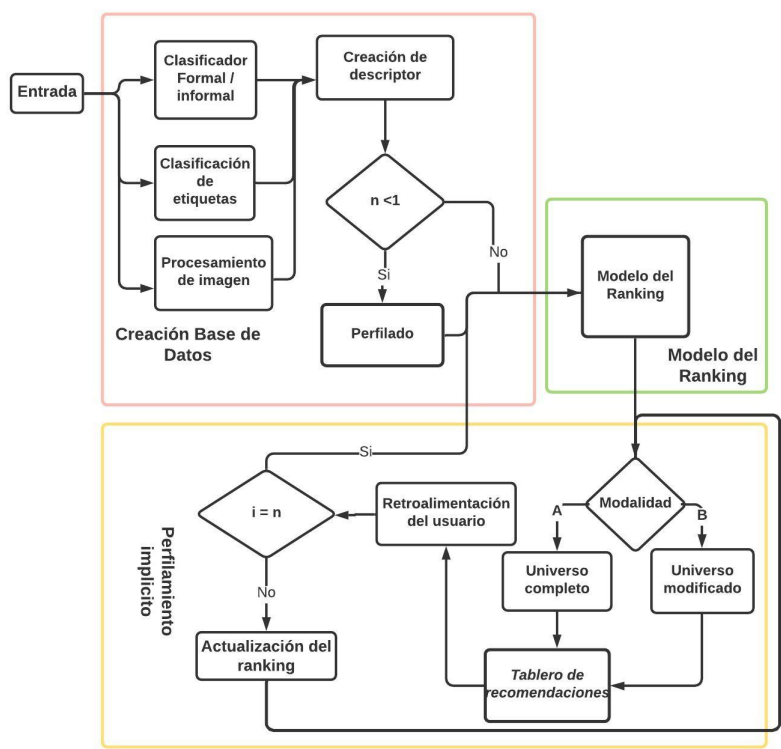


Figura 9.1: Diagrama de bloques del flujo del sistema

9.0.1. Creación de Base de Datos

El módulo de creación de Base de Datos se encarga de la codificación de características visuales de las imágenes de prendas. Este módulo se

construye a partir de 6 sub-procesos, para poder cumplir con el objetivo de la extracción de características de la imagen satisfactoriamente.

El módulo de creación de base de datos inicia con el procesamiento de imágenes para adquirir el color de la prenda y continua con dos clasificadores. Uno para las prendas, clasifica según su clase, y el otro para clasificarlas entre prendas formales o prendas informales. El resultado del procesamiento de imágenes y las dos CNN se concatenan para crear el descriptor de cada prenda.

Los descriptores de cada prenda se someten a una reclasificación entre tres grupos Alto, Medio y Bajo con el fin de crear conjuntos entre cada uno utilizando producto cartesiano.

Finalmente dentro de este módulo se construye el etiquetado por parte de profesionales. Los profesionales de moda son los encargados de etiquetar todas las combinaciones generadas con 0 o 1 (combina, no-combina), teniendo en cuenta cada una de las características y del estilo de las prendas de la combinación. El etiquetado por parte de profesionales genera un pre-ajuste de las *fashion recommendation*, de este modo el pre-ajuste es la clasificación con 0 o 1 (combina, no-combina) de las *fashion recommendation* brindado por los profesionales. Este pre-ajuste asegura que el sistema de recomendación de prendas comience a generar recomendaciones basadas en la opinión de un experto, evitando un arranque aleatorio en el sistema.

Procesamiento de imagen

El módulo de procesamiento de imágenes tiene como objetivo extraer el color característico de las prendas. El principal problema en la extracción de la característica del color en imágenes de prendas es el fondo. El fondo puede generar confusión al sistema cuando identifica los valores R, G y B de la imagen. Por esta razón dentro del módulo de procesamiento de imagen, se identifica la ubicación de la prenda de interés dentro de la imagen, para remover el fondo. La imagen de la prenda sin fondo es la que se utiliza para identificar su color.

La extracción del fondo de la imagen se realiza utilizando herramientas de la librería OpenCV. Por lo tanto la función encargada de este procedimiento

realiza el siguiente proceso: 1. Se aplica un desenfoque con filtro Gaussiano para reducir el ruido y los detalles. Luego se toma la imagen y se convierte a escala de grises. 2. Se procede a detectar los bordes y contornos, en este paso el algoritmo intenta encontrar las líneas de contraste, o bordes dentro de la imagen. Este paso es clave, ya que procesa previamente la imagen para ayudar a diferenciar cualquier objeto implementando el método Canny. Una vez que se encuentran los bordes, la búsqueda de contornos se vuelve mucho más fácil y precisa. 3. Se eliminan los contornos que sean demasiado grandes o demasiado pequeños para ser el primer plano. Los demás contornos se considerarán como el primer plano debido a que los contornos que ocupan la mayor parte de la imagen probablemente haga parte del fondo. Finalmente, se genera una máscara a partir de los contornos restantes y se combina con la imagen original. El resultado de la extracción del fondo se presenta en la segunda imagen de la Fig.9.2. Finalmente se implementa el algoritmo $k - means$ para agrupar los valores R, G, y B identificando los 3 colores predominantes por medio de un histograma sin tener en cuenta el fondo. El resultado de estos procesos se muestra en la Fig.9.2.



Figura 9.2: Extracción del color a las imágenes de prendas

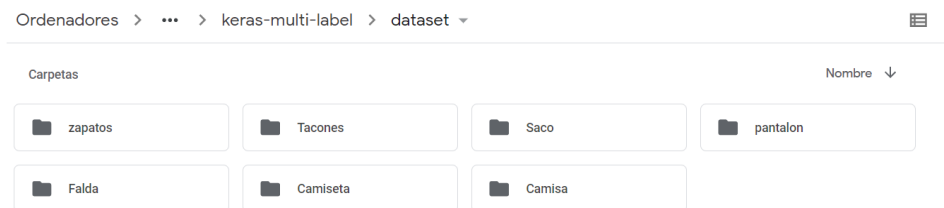
Clasificación de etiquetas

La clasificación de etiquetas se encarga de adquirir nuevas características visuales de la prenda. Dentro de este módulo se lleva a cabo la clasificación las imágenes entre sus respectivas clases.

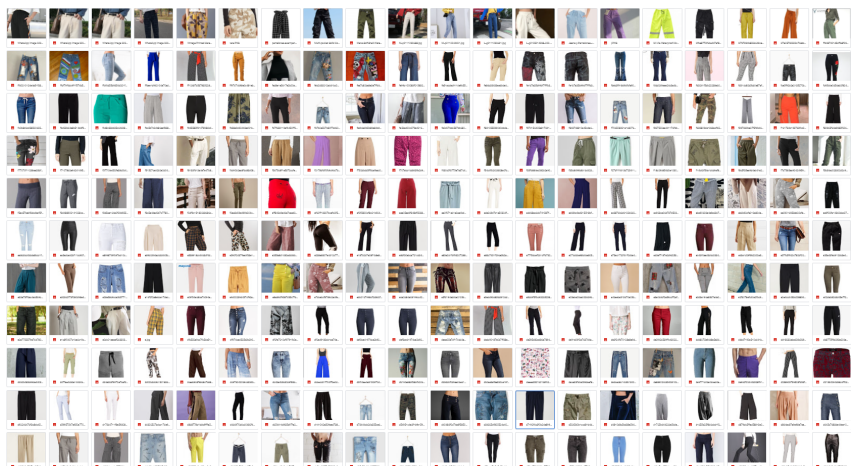
Cada una de las imágenes obtenidas para el entrenamiento se dividen en siete clases diferentes: pantalón, zapatos, tacones, falda, camisa, camiseta y buzo, como se muestra en la figura 9.4.

La base de datos para la clasificación fue construida manualmente y

cuenta con un total de 14000 imágenes públicas de prendas adquiridas de redes como Instagram y Pinterest. El proceso de selección de las imágenes consistió en buscar las imágenes de prendas, descargarlas y guardarlas en la carpeta a la que pertenece. La estructura de la base de datos se muestra en la Fig.9.3 a. Donde la base de datos contiene 7 carpetas cada una de las clases y con las imágenes correspondientes.



(a) Estructura de la base de datos



(b) Vista a la clase pantalón

Figura 9.3: Ejemplo de estructura de la base datos Fig.a junto con demostración de lo que contienen las clases, en este caso la clase pantalón Fig.b

Dentro de las 14000 imágenes existen 2000 imágenes de cada clase. Es decir se aseguro que las 7 diferentes clases que existen (pantalón, zapatos, tacones, falda, camisa, camiseta y buzo) tuvieran 2000 imágenes como se muestra en la Fig.b, para garantizar una base de datos balanceada, donde

las etiquetas tuviera la misma cantidad de información.

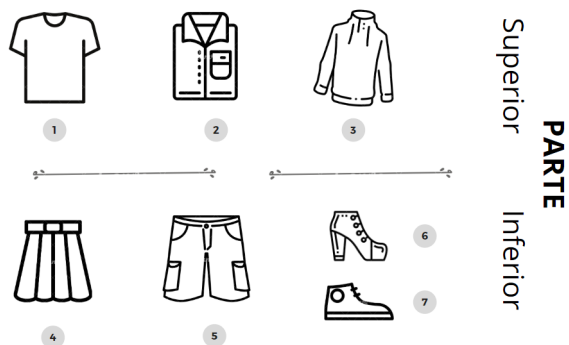


Figura 9.4: Distribución de etiquetas

El proyecto está desarrollado con la plataforma de aprendizaje automático *TensorFlow*, esta plataforma puede ejecutar *Application Programming Interfaces* (API) de aprendizaje profundo como Keras. Con el objetivo de definir el modelo que se implementara para esta clasificación, se realiza una comparación de rendimiento entre la red neuronal VGGNet-16 y ResNet-50. Estas pruebas se realizan utilizando las versiones pre-entrenadas disponibles en la librería de Keras.

Se elige entre ambos modelos utilizando indicadores como la sensibilidad (9.1), la especificidad (9.2), y la precisión (9.3) ya que estos indican el rendimiento de una red neuronal.

$$\text{Sen} = \frac{\text{TP}}{\text{FN} + \text{TP}} \quad (9.1)$$

La sensibilidad (Sen) se refiere a la capacidad del clasificador para detectar las muestras positivas.

$$\text{Spe} = \frac{\text{TN}}{\text{FP} + \text{TN}} \quad (9.2)$$

Contrario a lo anterior la especificidad (Spe) se refiere a la capacidad del clasificador para detectar las muestras negativas.

$$\text{Acc} = \frac{\text{TP} + \text{TN}}{\text{FP} + \text{TN} + \text{TP} + \text{FN}} \quad (9.3)$$

La precisión (Acc) se refiere al número total de predicciones correctas realizadas por el clasificador.

La matriz de confusión de este proceso tiene un tamaño de 7×7 , donde las etiquetas de las prendas predichas se clasifican según 4 casos diferentes como:

- Verdadero positivo (TP) designa el número de muestras Positivas predichas correctamente como P
- Verdadero Negativo (TN), designa el número de N muestras correctamente predichas como N.
- Falso Negativo (FN), designa el número de muestras P predichas como N.
- Falso Positivo (FP), designa el número de Negativos muestras predichas como Positivo.

Arquitectura del Primer Modelo Implementado: VGGNet-16

La arquitectura CNN que se propone para implementar en la clasificación es la versión VGGNet presentado por Simonyan y Zisserman en 2014 [66] con menos capas.

La arquitectura VGGNet cuenta con las siguientes características:

- Usan capas convolucionales con un tamaño 3×3 apiladas una encima de otra incrementando la profundidad del modelo.
- La matriz resultante de la convolución, es reducida utilizando el método Max Pooling.
- Capas completamente conectadas al final de la red antes de un clasificador con función de activación Softmax.

El método de diseño del modelo VGGNet-16 requiere de los siguientes parámetros:

- Dimensión del ancho de la imagen

- Dimensión de la altura de la imagen
- Canales de la imagen
- El número de clases del conjunto de datos

En la Fig.9.5 se muestra la gráfica de entrenamiento para el modelo VGGNet-16 durante 200 épocas. Se gráfica la perdida y la precisión que tiene con nuestros datos.

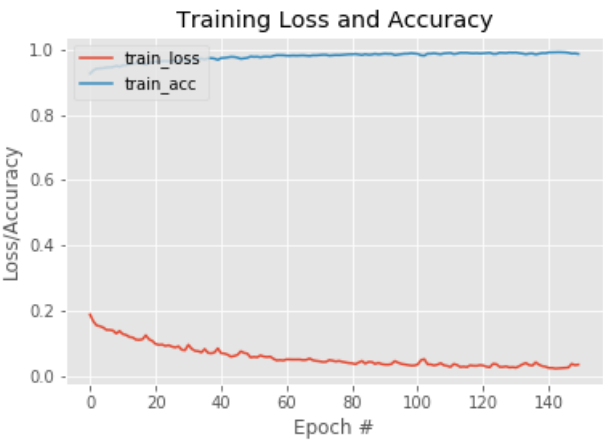


Figura 9.5: Gráfica de entreno para el modelo VGGNet-16

Para llegar a una mejor conclusión sobre el rendimiento del modelo, se utiliza la matriz de confusión. La Tabla9.1 es la matriz de confusión del modelo VGGNet-16. El resultado de las métricas para la Tabla. 9.1, son:

Tabla 9.1: Matriz de confusión VGGNet-16

Etiqueta real	Etiqueta predicha						
	Camisa	Camiseta	Falda	Buzo	Tacón	Pantalón	Zapato
Camisa	1,997	0	0	1	0	2	0
Camiseta	14	1,983	0	3	0	0	0
Falda	1	0	1,998	1	0	0	0
Buzo	12	3	1	1,984	0	0	0
Tacón	2	0	0	0	1,996	1	1
Pantalón	0	0	0	0	0	1,999	1
Zapato	0	0	1	0	2	0	1,997

$Acc = 0,99$. Promedio de la sensibilidad por clase $Sen = 0,98$. Promedio de la especificidad por clase $Spe = 0,99$

Arquitectura del Segundo Modelo Implementado: ResNet-50

La librería Keras proporciona una interfaz capaz de cargar modelos previamente entrenados, como el modelo ResNet-50 el cual es uno de los más populares para la clasificación de imágenes.

La arquitectura de ResNet-50 tiene cuatro. La etapa 1 de la arquitectura consta de tres bloques residuales que contienen tres capas cada uno. El tamaño de los núcleos utilizados para realizar la operación de convolución en las tres capas del bloque de la etapa 1 son 64, 64 y 128 respectivamente. A medida que avanza de una etapa a otra, el ancho del canal se duplica y el tamaño de la entrada se reduce a la mitad. Para cada función residual F , se apilan 3 capas una sobre la otra. Las tres capas son convoluciones 1x1, 3x3 y 1x1. Las capas de convolución 1x1 son responsables de reducir y luego restaurar las dimensiones. Finalmente, la red tiene una capa de agrupación promedio seguida de una capa totalmente conectada que tiene 7 neuronas que corresponde a la salida de clase. Igual que en el caso anterior con el modelo VGGNet-16 se trabajan con imágenes de entrada de 96x96 con tres canales.

En la Fig.9.6 se muestra la gráfica de entrenamiento para el modelo VGGNet-16 durante 200 épocas. Se grafica la perdida y la precisión que tiene con nuestros datos. Para llegar a una mejor conclusión sobre el rendimiento del modelo, se utiliza la matriz de confusión. La Tabla 9.2 es la matriz de confusión del modelo ResNet-50. El resultado de las métricas para la Tabla.9.2, son: $Acc = 0,97$. Promedio de la sensibilidad por clase $Sen = 0,96$. Promedio de la especificidad por clase $Spe = 0,96$

Como se observa en las tablas 9.1 y 9.2, existen errores en algunas etiquetas, sin embargo, con el modelo VGGNet-16 se obtienen mejores resultados, ya que tiene mejor distribución en cada una de las clases y las métricas de precisión, sensibilidad y especificidad son superiores a las del modelo ResNet-50.



Figura 9.6: Gráfica de entreno para el modelo ResNet-50

Tabla 9.2: Matriz de confusión ResNet-50

Etiqueta real	Etiqueta predicha						
	Camisa	Camiseta	Falda	Buzo	Tacón	Pantalón	Zapato
Camisa	1,990	2	0	1	2	2	3
Camiseta	64	1,800	70	3	3	2	5
Falda	1	15	1,898	1	1	100	0
Buzo	12	3	1	1,980	1	0	0
Tacón	2	0	5	4	1,994	0	1
Pantalón	0	8	38	0	0	1,961	1
Zapato	0	0	1	0	2	0	1,997

Clasificador Formal-Informal

Dentro del módulo de la creación de base de datos el proceso llamado Clasificador Formal-Informal se encarga de adquirir la ultimas características visuales de la prenda en el vector que la describe. Este proceso como su nombre lo indica tiene como propósito identificar si la prenda presentada es de estilo formal o informal. La clasificación binaria desarrollada dentro de este proceso implementa la misma CNN que se utiliza en el proceso de clasificación de etiquetas.

En el Clasificador Formal-Informal las imágenes de prendas se dividen entre tres nuevos grupos, Alto, Medio, y Bajo como se muestra en la Fig.9.7. En el grupo Alto (A) se encuentran camisa, camiseta, o saco. El

grupo Medio (M) se encuentran pantalones, o faldas. Finalmente el grupo Bajo (B) son tenis, o tacones. Los tres grupos se clasifican entre formal o informal, siendo 0 la clase informal y 1 la clase formal.

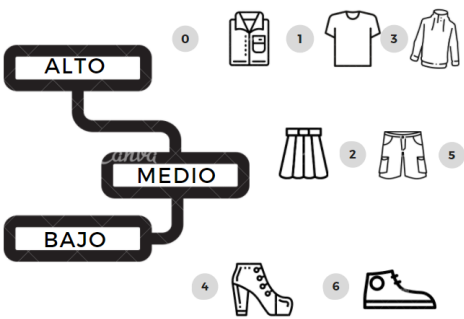


Figura 9.7: Nueva distribución de etiquetas entre alto, medio y bajo

Creación del descriptor

Este proceso tiene como objetivo la construcción del descriptor (9.4), ya que es el conjunto de características que representa cada prenda. El descriptor está formado por la unión de las 11 características adquiridas en el módulo de creación de base de datos. Las características visuales se van recopilando entre el procesamiento de imágenes, la clasificación formal/informal y la clasificación de etiquetas.

$$d = \{R \oplus G \oplus B \cup et \cup F\} \quad (9.4)$$

Donde d es el descriptor, R , G y B son los valores de color. et es la distribución de los porcentajes entre las siete etiquetas $et \cap (Camiseta, Falda, Chaqueta, Tacones, Pantalones, Zapatos)$, y F representa la categoría formal-informal.

La Fig.9.8 representa un ejemplo visual del resultado que se tiene en el descriptor para una prenda perteneciente a la clase de pantalón. Donde se muestra cada una de las categorías o características visuales que componen al descriptor.

	Nombre	Dataset\pantalon7.jpg
	R	196
	G	219
	B	244
	Camisa	0,0000000004740481011
	Camiseta	0
	Falda	0
	Saco	0
	Tacones	0
	Pantalon	100
	Zapatos	0
	Formal	0

Figura 9.8: Ejemplo del descriptor para una prenda de clase pantalón

9.0.2. Generación de combinaciones

Finalmente en el módulo de creación de base de datos se encuentra el proceso de generación de imágenes. Este proceso como su nombre lo indica se encuentra a cargo de la creación de recomendaciones de estilo a partir de la agrupación de descriptores. Dentro del proceso de generación de combinaciones se plantea el siguiente problema para generar todas las posibles agrupaciones.

Dado un conjunto $O_i \equiv \{d_a + d_m + d_b\}$ donde d_t es el descriptor de una prenda entre el grupo Alto (camisa, camiseta, saco). Donde d_m es el descriptor de una prenda entre el grupo Medio (pantalón, falda), y d_b es el descriptor de una prenda perteneciente al grupo Bajo (tenis, tacones). Cada uno de los conjuntos O_i debe contener tres descriptores diferentes, uno de cada grupo A, M, B .

Este O_i lleva el nombre de *fashion recommendation*. Por lo tanto O es la base de datos del sistema o el universo de discurso. O se puede definir matemáticamente (9.5) como el producto cartesiano de los grupos A, M, B .

$$\begin{aligned}
 A \times M \times B &= \{(a, m, b) | a \in A, m \in M, b \in B\} \\
 \therefore (a, m, b) &\in O_n \\
 O &\equiv \{O_1, O_2, O_3 | O_n \in A \wedge M \wedge B\}
 \end{aligned}
 \tag{9.5}$$

Donde A es el conjunto de descriptores de prendas dentro de Alto. M es el conjunto de descriptores de prendas dentro de Medio. B es el conjunto de descriptores de prendas dentro de Bajo.

Etiquetado profesional

El proceso de etiquetado profesional se encarga de etiquetar las combinaciones basándose en las opiniones de los profesionales. El etiquetado profesional consiste en presentar las imágenes del dataset de combinaciones generado en la sección 9.0.2 a profesionales de la moda. Los profesionales de moda son los encargados de etiquetar según su criterio que *fashion recommendation* consideran que combina y cuáles no. A nivel industrial los profesionales de moda serían profesionales expertos en estilo, en este caso nosotros mismos generamos el etiquetado original. El etiquetado de *fashion recommendation* se realiza teniendo en cuenta las características de cada prenda como los colores y estilo (formal o no formal) de cada una de las tres prendas. Las etiquetas generadas por los profesionales se utilizan en el primer entrenamiento del Modelo de ranking en la sección 9.0.3, antes de interactuar con el usuario.

Este proceso garantiza que el estado inicial del sistema de recomendación de prendas se base en el uso de un pre-ajuste de *fashion recommendation*, según el experto. El etiquetado profesional, también evita un arranque aleatorio asegurando que el sistema comience a generar recomendaciones basadas en la opinión de un experto, que tengan lógica.

9.0.3. Modelo de ranking

El módulo del Modelo de ranking establece la relación de cada O_i con su respectiva puntuación Q , creando una tabla denominada *ranking table*. La red DNN utiliza la *ranking table* como conjunto de entrenamiento. Sin embargo, el primer entrenamiento de la DNN utiliza las etiquetas profesionales, proporcionando un valor de peso inicial. La matriz de confusión muestra el rendimiento de la red en la Tabla 9.3.

Tabla 9.3: Matriz de confusión del modelo.

Clase real	Clase predicha	
	Falso	Verdadero
Falso	1, 867, 300	81, 800
Verdadero	49,100	1,900,000

El resultado de las métricas de validación son: $Acc = 0,966$, Sensibilidad $Sen = 0,97$, y especificidad $Spe = 0,97$, validando el desempeño de la red como satisfactorio. Según la interacción del sistema y el usuario, la *ranking table* es actualizada en la sección de perfilamiento implícito creando una memoria de corto plazo al sistema. La *ranking table* es fundamental para el desarrollo del sistema de recomendación de prendas, ya que es la fuente de información que alimenta la memoria de corto plazo.

9.0.4. Perfilamiento implícito

El último módulo dentro del algoritmo del sistema de recomendación propuesto se llama perfilamiento implícito. Este módulo genera los procesos que permitan al sistema de recomendación adaptarse al usuario de forma implícita cada vez que exista una interacción entre el usuario y el sistema.

El perfilamiento implícito se encarga de la generación del tablero de recomendaciones que se presenta al usuario, también de la actualización de la *ranking table* y finalmente de determinar cuándo se debe de entrenar la DNN. Para llevar a cabo los objetivos de la sección de perfilamiento implícito, se ha dividido en tres procesos denominados modalidad del sistema, Tablero de recomendaciones, y actualización del ranking.

Modalidad del sistema

El proceso denominado modalidad del sistema genera la *ranking table* que se presenta al sistema para que ejecute el modelo. De este modo, el usuario decide entre una recomendación abierta o basada en una prenda preseleccionada.

a) Recomendación Abierta.

En esta modalidad de recomendación la *ranking table* mencionada en la sección 9.0.3 se presenta completa y ordenada de forma descendente según el valor Q .

b) Recomendación basado en prendas pre-seleccionadas.

En esta modalidad la *ranking table* se modifica basándose en la prenda preseleccionada. La tabla es creada solo con las O_i que contengan la prenda seleccionada por el usuario. Organizando la tabla también de forma descendente según el Q .

Tablero de recomendaciones

Dentro del módulo de perfilamiento implícito, se presenta el proceso denominado Tablero de recomendaciones. El tablero de recomendaciones tiene por objetivo presentar visualmente las diferentes combinaciones O_i al usuario de forma organizada, como se muestra en la Fig.10.1.

La estructura del tablero de recomendaciones consta de *clusters* clasificados en tres categorías. Las categorías se definen según el valor Q , la puntuación de cada O_i en la *ranking table* creada en la sección 9.0.4. Las categorías son:

1. *Best*: Esta categoría representa las primeras j recomendaciones con mayor Q de la *ranking table*, siendo j la cantidad de O_i por categoría presentada al usuario.
2. *No – select*: Esta categoría contiene los O_i con Q alto, pero que no están dentro de las primeras j combinaciones de la *ranking table*.
3. *Worst*: Es la categoría que contiene los O_i con puntuación Q baja. Estas combinaciones hacen parte del grupo no compatible con las preferencias del usuario, pero que igualmente se presenta al usuario para brindarle una máxima exploración.

Dentro de cada categoría, las recomendaciones de moda se agrupan dinámicamente según características similares con el algoritmo K-means. Esta sub-agrupación dentro de cada categoría se realiza para permitir observar los diferentes estilos de combinaciones que se pueden tener en una sola categoría. Por último, el tablero de recomendaciones se presenta al usuario para que elija una recomendación de moda denominada O_u .

Actualización de la ranking

El objetivo de la Actualización del ranking es generar el cambio de los valores Q del *ranking table*, permitiendo que el sistema se adapte según las preferencias del usuario. Cuando el usuario selecciona una recomendación, se genera la actualización del valor Q de cada O_i a partir del O_u dado por el usuario.

La norma $L_2(x)$ entre O_u y O otorga el ranking actualizado. La *ranking table* se ordena teniendo en cuenta el valor de x_i , esta vez en orden ascendente según x . Las recomendaciones de moda que se acercan a O_u se compensan (9.6a), y las que tienen una diferencia mayor se castigan (9.6b).

$$Q = Q + ((1 - Q) * \alpha) \quad (9.6a)$$

$$Q = Q + (Q * \alpha) \quad (9.6b)$$

La ecuación (9.6a) se implementa cuando $\alpha > 0$ y cuando $\alpha \leq 0$ se utiliza (9.6b). Donde Q representa la puntuación de O_i , y α es el factor que varía dependiendo de la diferencia entre la O_u y cada O_i . El factor α , se calcula utilizando una función Sigmoidal. (9.7).

$$\alpha = 2 * \left(1 - \frac{1}{1 + e^{(-\sigma * (x_i - x_j))}} - 0,5 \right) \quad (9.7)$$

Donde σ es la pendiente, x_j es el valor de la diferencia y j indica el número de recomendaciones de moda presentadas al usuario. Con las ecuaciones (9.6a) y (9.6b) se actualiza el valor de Q , variando el ranking cada vez que el usuario interactúa con el sistema. Con las ecuaciones (9.6a) y (9.6b) se actualiza el valor Q , variando el ranking cada vez que el usuario interactúa con el sistema.

Actualización de la red neuronal

Este proceso es el encargado de determinar cuándo hay que volver a entrenar la red neuronal. La arquitectura DNN que se propone para implementar la actualización se observa en el cuadro 9.1. La red cuenta con 9 capas completamente conectadas, la primera capa maneja 33 entradas, y tiene salida única. Las Funciones de activación de cada capa son relu y en la salida maneja sigmoid. Adicionalmente se utiliza la herramienta *early stopping*. La parada temprana o *earlystopping* se implementa como forma de regularización para evitar el sobreajuste cuando se está re-entrenando el modelo, en este caso la parada temprana del entrenamiento se basa en el monitoreo de "loss" la pérdida.

El sistema dispone de un contador de interacciones del usuario que determina cuándo se debe volver a entrenar la DNN.

La DNN debe ser entrenada cada vez que alcance el número total de interacciones definido por el usuario. El re-entrenamiento de la red neuronal utiliza la versión más reciente del *ranking table*. El valor del peso se restablece, generando una nueva tabla que se adapta al usuario.

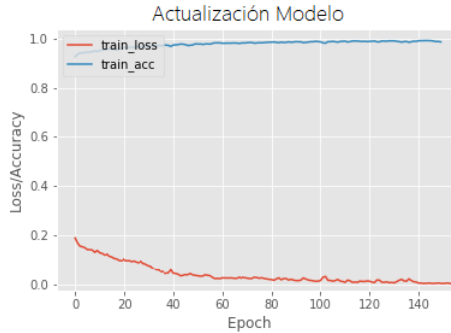


Figura 9.9: Gráfica de entreno para el reentrenamiento del modelo DNN

En la Fig.9.9 se muestra el resultado de un reentrenamiento, graficando la pérdida y el accuracy del entrenamiento, para poder validar el buen comportamiento de la red. Más adelante en la sección de Resultados en las figuras Fig.10.11, Fig.10.21, Fig.10.29 y Fig.10.37 se muestra el resultado de las recomendaciones después de la actualización del modelo. La red

funciona como un sistema de memoria a largo plazo basado en las interacciones del usuario.

Finalmente se presentan las listas de código 9.2, 9.3, 9.4 y 9.5 se muestra el pseudocódigo del desarrollo implementado para el sistema de recomendación.

En la primera lista de código 9.2 se definen las variables y se comienza la definición de funciones como: 1. Color que se encarga de la extracción del fondo y de retornar los valores R, G, y B predominantes en la prenda. 2. F esta función se encarga de entregar los diferentes casos de grupos entre Alto, Medio y bajo según el tipo de prenda que el usuario seleccione en la modalidad basada en una prenda preseleccionada (ver sección 9.0.4).

En la segunda lista de código 9.3 se continúa las definiciones de funciones como: 1.Creacion_combinaciones, en esta función se desarrolla lo explicado en la sección 9.0.2. 2.Crea_ranking_table, crea la tabla que contiene el valor Q de las combinaciones y 3. Entreno es la que se encarga del entreno de la red y actualización de la red neuronal.

Dentro de la tercera lista de código 9.4 se definen dos funciones mas la primera llamada grafica se construye la imagen de las combinaciones para presentarcas al usuario de forma grafica y en la funcion llamada dinamica se implementa el proceso explicado en la seccion 9.0.2.

En la cuarta y ultima lista de código 9.5 se muestra el flujo del código, que se plantea en el diagrama de bloques Fig.9.1.

Listing 9.1: Arquitectura red

Model: "DNN"		
Layer (type)	Output Shape	Param #
dense_84 (Dense)	(None, 255)	8670
dense_85 (Dense)	(None, 255)	65280
dense_86 (Dense)	(None, 128)	32768
dropout_6 (Dropout)	(None, 128)	0
dense_87 (Dense)	(None, 128)	16512
dense_88 (Dense)	(None, 24)	3096
dense_89 (Dense)	(None, 12)	300
dense_90 (Dense)	(None, 12)	156
dense_91 (Dense)	(None, 6)	78
dense_92 (Dense)	(None, 1)	7

Listing 9.2: Pseudocodigo

```

input: imagen
Modalidad: Abierta o Based_preselecc
D: Descriptor(R, G, B, Et_prenda, Et_estilo)
O_u: Seleccion del usuario
valor_Q: Resultado combinacion entre 0 a 1
path: Ruta imagen original
### DEFINICION VARIABLES
historial = []
Bd = data_frame[
    cols:R, G, B, Et_prenda, Et_estilo, path
]

j = cantidad de recomendaciones deseadas
n = Interacciones antes de re-entrenar
best = data_frame()
best_noselect = data_frame()
worst = data_frame()
###FUNCIONES
def color(image):
    1. quita el fondo
    2. realiza kmeans para los colores
    return R,G,B
def f(x):
    return {
        Camisa : (Bajo, Medio, prenda),
        Camiseta : (Bajo, Medio, prenda),
        Pantalón : (Bajo, prenda, Alto),
        Buzo : (Bajo, Medio, prenda),
        Zapatos : (prenda, Medio, Alto),
        Falda : (Bajo, prenda, Alto),
        Tacones : (prenda, Medio, Alto),
    }.get(x)

```


Listing 9.3: Pseudocodigo

```

def Creacion_combinaciones(Bd,i):

    Alto = where Bd.Et_prenda in
            [Camisa, Camiseta, Buzo]
    Medio = where Bd.Et_prenda in [Pantalon, Falda]
    Bajo = where Bd.Et_prenda in [Zapatos, Tacones]

    if Modalidad = Based_preselecc:
        (Bajo, Medio, Alto) =
            f(D_prenda.Et_prenda)
    else Modalidad = Abierta:
        pass
    O = Alto x Medio x Bajo
return O

def entreno(status,tablaQN):
    if status = False # si el sistema no ha
#interactuado nunca con el usuario
        DNN = DNN.fit(pre_ajuste)
    else:
        DNN = DNN.fit(tablaQN)
return DNN

def crea_ranking_table(DNN, Datos):

    valor_Q = DNN.predict(Datos)
    tablaQN = concatenate(Datos, valor_Q)
return tablaQN.sort(asc)

```

Listing 9.4: Pseudocodigo

```

def grafica(grupo,clusters,categoria):
    1. grafica imagen construida de cada
        categoria entre best, best_noselect y worst

###Interaccion con usuario
def dinamica(n,TablaQN):
    for cada i de n do :
        rango = valor_Q promedio 0.6
        noselect =
            tablaQN[Where valor_Q >= rango][j:rango]
        best = tablaQN[Where valor_Q >= rango][0:j]
        worst = tablaQN[Where valor_Q<rango][rango:]
        grafica(best,clusters,'best')
        grafica(worst,clusters,'worst')
        grafica(noselect,clusters,'noselect')
        O_u = input()
        if O_u not in historial:
            historial = historial.add[O_u]

        x = euclidean_distance(O_u,tablaQN)
        dist = concatenate(tablaQN,x).sort(x)
        inflexion = dist.x[200] #distancia
        #en posc. 200

        factor = sigmoid(distancias,0,inflexion)

        if factor > 0:
            valor_Q = valor_Q +((1-valor_Q)*factor))
        else:
            valor_Q = valor_Q + (valor_Q)*factor
return tablaQN

```

Listing 9.5: Pseudocodigo

```

##codigo
for cada imagen del dataset do :
    r,g,b=color(imagen)
    Et_prenda = Etiquetas.predict(imagen)
    if Et_prenda in [Camisa, Camiseta, Buzo]:
        Et_estilo = estilo_alto.predict(imagen)
    if Et_prenda in [Pantalon, Falda]:
        Et_estilo = estilo_medio.predict(imagen)
    if Et_prenda in [Zapatos, Tacones]:
        Et_estilo = estilo_bajo.predict(imagen)
    D=concatenate(R,G,B,Et_prenda, Et_estilo)
    Bd.agrega(D)
Modalidad = input( )
O = Creacion_combinaciones(Bd)
DNN = entreno(False , _)
tablaQN = crea_ranking_table(DNN,Pre_ajuste)
loop :
    tablaQN_final = dinamica(n,TablaQN)
    tablaQN = entreno(True , tablaQN_final)
    tablaQN = crea_ranking_table(DNN,tablaQN_final)

```

Capítulo 10

Experimentos

El sistema se evalúa realizando cuatro experimentos de verificación, usando imágenes brindadas por el usuario con el fin de verificar el proceso de adaptación del sistema a las preferencias del usuario. La adaptación del sistema se verifica con la variación del valor Q de cada O_i . Estas variaciones se presentan visualmente en gráficas, para cada categoría.

En los primeros dos experimentos de las secciones 10.1 y 10.2, el sistema implementa la modalidad de recomendación abierta durante 20 iteraciones. El usuario selecciona la misma recomendación de moda que pertenece a la categoría Worst 10.1 y No-select 10.2. En el tercer experimento desarrollado en la sección 10.3, se seleccionan cinco combinaciones diferentes, las cuales son alternadas entre las 20 interacciones. Finalmente en el cuarto experimento desarrollado en la sección 10.4, se selecciona tres combinaciones diferentes uno de cada categoría también alternándose entre 30 interacciones. La Fig.10.1 presenta el tablero de recomendaciones de O_i inicial dividido en sus tres respectivas categorías Best Fig.a, No-select Fig.b y Worst Fig.c. Esta distribución está dada según el valor de peso por defecto, a partir del etiquetado profesional.



(a) Categoría Best



(b) Categoría No-select



(c) Categoría Worst

Figura 10.1: Tablero de recomendaciones basado en el etiquetado profesional donde muestra los diferentes estilos de las categorías *best* Fig.a, *No – select* Fig.b y *worst* Fig.c

10.1. Experimento A

El primer experimento tiene como finalidad demostrar como la elección de una combinación de la categoría *Worst*, afecta al sistema. La categoría *Worst* se presenta en el tablero de recomendaciones para que el usuario pueda explorar más opciones, no solo las que presentan un valor Q alto. Primero se presenta el tablero de recomendaciones para que el usuario seleccione un O_i . El número de recomendaciones de moda dentro de cada categoría es de 30, como se muestra en la Fig.10.1, lo que quiere decir que $j = 30$.



Figura 10.2: Selección del usuario O_u

La recomendación de moda seleccionada (O_u) utilizada se presenta en la Fig.10.2

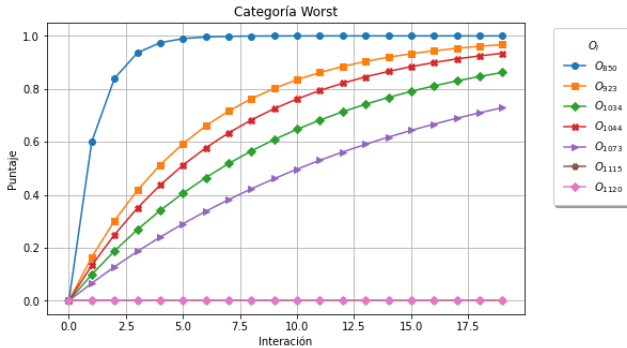


Figura 10.3: Variación del valor Q en la categoría *Worst*

En la Fig.10.3 se ve la variación del valor Q de O_u y cada O_i que representan una variación en la categoría *Worst*. Esto con el propósito de identificar como la selección de O_u afecta a los demás O_i de la misma categoría.



Figura 10.4: Representación visual de los conjuntos O_i en la categoría *Worst*

En la Fig.10.4 se muestra cada una de las combinaciones pertenecientes a cada O_i utilizadas en la gráfica de variación del valor Q Fig.10.3. La combinación O_{850} es la O_u , es decir la combinación seleccionada (ver Fig.10.2). La Fig.10.4 evidencia que las combinaciones O_{1115} y O_{1120} que no reciben ninguna recompensa (ver Fig.10.3) son diferentes a la O_u . Esto debido a que estas dos combinaciones manejan otro estilo, siendo las únicas que tienen falda. Adicionalmente los O_i que representa a cada clúster dentro de la categoría *Best* se muestran en la Fig.10.5.

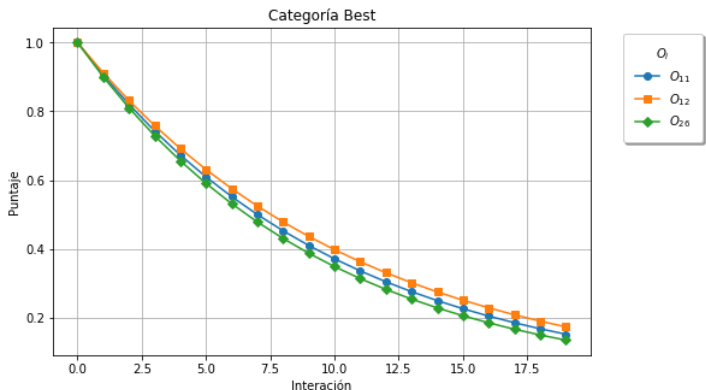


Figura 10.5: Variación del valor Q en la categoría *Best*

La Fig.10.6 contiene cada una de las combinaciones utilizadas en la gráfica de variación del valor Q Fig.10.5. En este caso ninguna de las combinaciones es recompensada (ver Fig.10.5). La Fig.10.6 evidencia que las combinaciones de esta categoría son bastantes diferentes y no se alinean con el estilo de O_u (ver Fig.10.2).



Figura 10.6: Representación visual de los conjuntos O_i de la categoría Best

Los O_i que representa a cada clúster dentro de la categoría *No – select* se muestran en la Fig.10.7. En ambas figuras Fig.10.5 y en la Fig.10.7, se expone cómo el sistema varía el valor Q de cada combinación O_i . La variación de los valores de Q demuestra cómo el sistema se adapta a las preferencias del individuo.

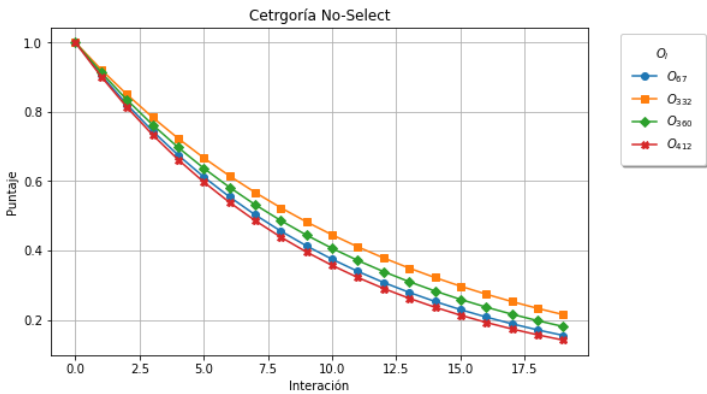


Figura 10.7: Variación del valor Q en la categoría *No – select*

La Fig.10.8 contiene cada una de las combinaciones utilizadas en la gráfica de variación del valor Q Fig.10.7. Este caso es similar a lo que sucede en la categoría *Best*. Las combinaciones dentro de *No – select* no son recompensadas (ver Fig.10.7). El castigo de estas combinaciones es debido a que ninguna de las presentadas en la Fig.10.8 tienen alguna similitud con el estilo de la combinación seleccionada (ver Fig.10.2).



Figura 10.8: Representación visual de los conjuntos O_i en la categoría No-select

En la Fig.10.9 se rastrea el cambio de las O_i similares a O_u . Como se muestra en la Fig.10.9, no solo O_u es recompensada. Todo *fashion recommendation* que pertenezca, o sea similar a la selección del usuario se ve beneficiado.

La Fig.10.10 contiene cada una de las combinaciones utilizadas en la gráfica de variación del valor Q Fig.10.9. Esta imagen presenta todas las combinaciones que el sistema identifica como estilos o posibilidades similares a la preferencia del usuario. Cada una de las combinaciones presentadas en la Fig.10.10 tiene coherencia, debido a que comparten el mismo estilo de zapatos, y de pantalón los cuales son informales.

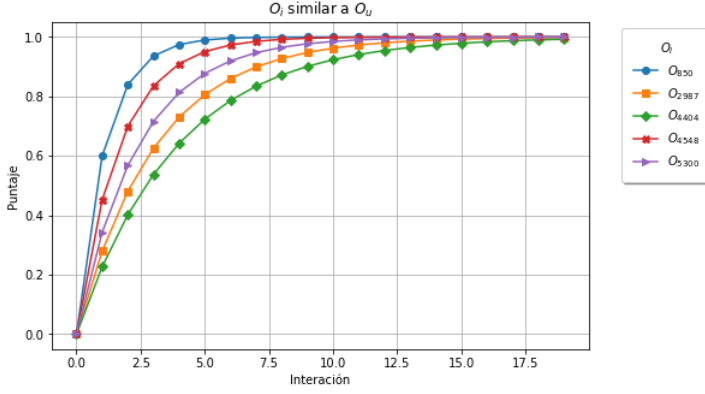


Figura 10.9: Variación del valor Q en los O_i similares a la selección del usuario (O_u)

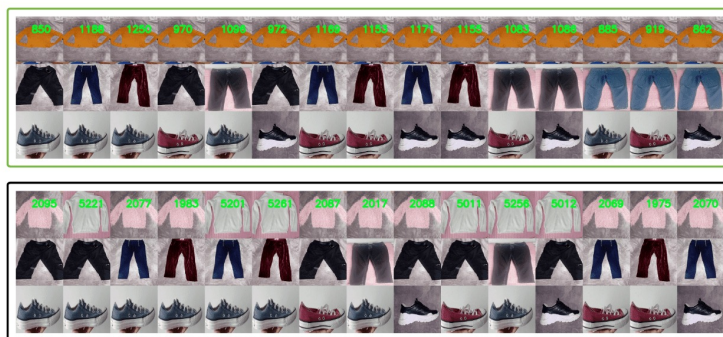


Figura 10.10: Representación visual de los conjuntos O_i similares a O_u

En las figuras Fig.10.5, Fig.10.3, Fig.10.7, y Fig.10.9 se contempla de forma visual la variación del ranking. Generado por la actualización de los valores Q cada vez que el usuario interactúa con el sistema.

Como se menciona en la sección 9.0.4 después de 20 interacciones el modelo construye nuevo *ranking table* basada en el perfil del individuo.

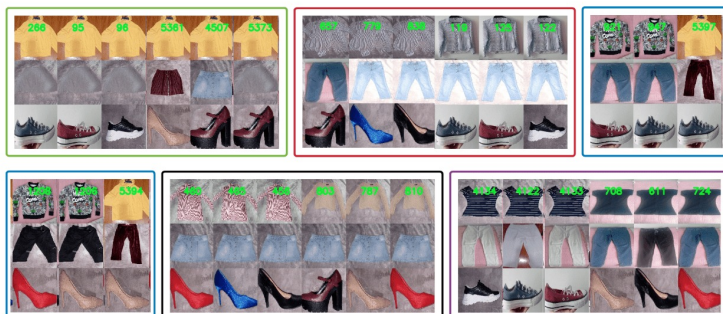
Tras la interacción del usuario con el sistema, el table de recomendaciones cambia tal y como se muestra en la Fig.10.11. Ahora, este tablero se basa totalmente en las preferencias del usuario. El tablero de recomendaciones con el modelo actualizado Fig.10.11 sigue manteniendo la estructura entre sus tres respectivas categorías Best Fig.a, No-select Fig.b y Worst Fig.c.



(a) Categoría best



(b) Categoría No-select



(c) Categoría Worst

Figura 10.11: Tablero de recomendaciones adaptado a las preferencias del usuario donde muestra los diferentes estilos de las categorías *best* Fig.a, *no – select* Fig.b y *worst* Fig.c

10.2. Experimento B

En el segundo experimento realizado, el sistema también implementa la modalidad de recomendación abierta durante 20 iteraciones. El usuario selecciona la misma recomendación de moda que en esta ocasión pertenece a la categoría *No-select*. Esto con el fin de verificar el proceso de adaptación del sistema a las preferencias del usuario y observar el comportamiento del sistema cuando se selecciona una recomendación que no pertenece la categoría *Best*. Igual que antes, como primer paso se presenta el tablero de recomendaciones generado por el etiquetado profesional para que el usuario seleccione el O_i dentro de la categoría *No-select*.



Figura 10.12: Selección del usuario O_u del experimento B

La recomendación de moda seleccionada (O_u) se presenta en la Fig. 10.12

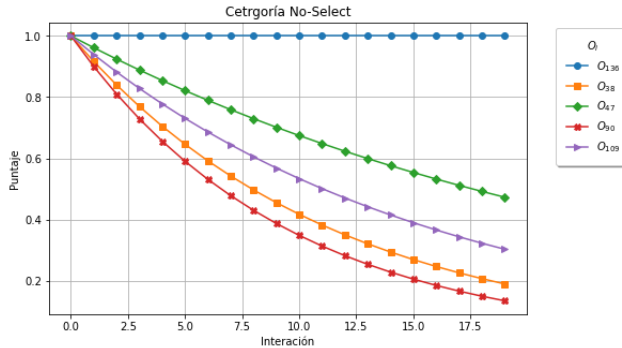


Figura 10.13: Variación del valor Q en la categoría *No-select*

En la Fig.10.13 se ve la variación del valor Q de O_u y cada O_i que representan las combinaciones con mayor variación en la categoría *No – select*. Esto con el propósito de identificar como la selección de O_u afecta a los demás O_i de la misma categoría.



Figura 10.14: Representación visual de los conjuntos O_i de la categoría No-select

En la Fig.10.14 se muestra cada una de las combinaciones utilizadas en la gráfica de variación del valor Q (ver Fig.10.13). La combinación O_{136} es la O_u , es decir la combinación seleccionada (ver Fig.10.12). La Fig.10.14 evidencia que las O_i que no reciben ninguna recompensa (ver Fig.10.13) son diferentes a la O_u . Esto debido a que ninguna de las combinaciones O_{38} , O_{47} , O_{90} , y O_{109} se asemejan a la preferencia del usuario, por lo que es totalmente válido que el sistema los castigue.

Los O_i que representa a cada clúster dentro de la categoría Best se muestra en la Fig.10.15. En la figura, se expone cómo el sistema varía la Q de cada O_i . La forma en que cambian los valores de Q demuestra cómo el sistema se adapta a las preferencias del individuo. Esta vez en la Fig.10.15 se ve como las O_i van variando de forma más lenta comparando con la Fig.10.5, debido a que el castigo y premiación depende del valor Q , por tanto la categoría *Best* y *No – select*, no se ve muy afectada cuando la O_i pertenece a estos grupos.

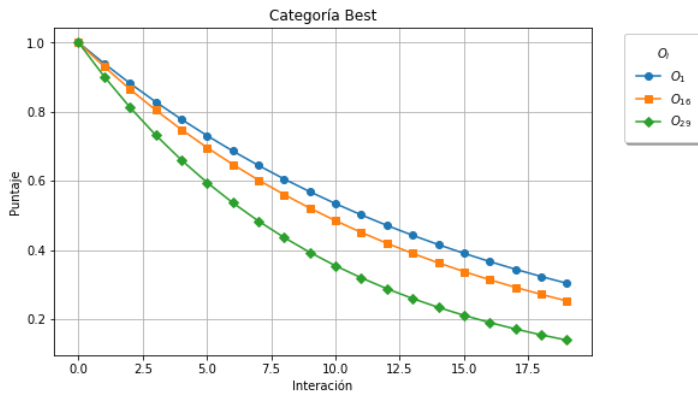


Figura 10.15: Variación del valor Q en la categoría Best

En la Fig.10.16 se muestra cada una de las combinaciones utilizadas en la gráfica de variación del valor Q (ver Fig.10.15). Dentro de la categoría best sucede algo similar que con los O_i presentados en la Fig.10.13. Esto debido a que ninguna de las características visuales de las combinaciones se acerca a la preferencia del usuario, por lo que es totalmente válido que el sistema los castigue.



Figura 10.16: Representación visual de los conjuntos O_i de la categoría Best

Además, los O_i que representa a cada clúster dentro de la categoría No-select se muestra en la Fig.10.17.

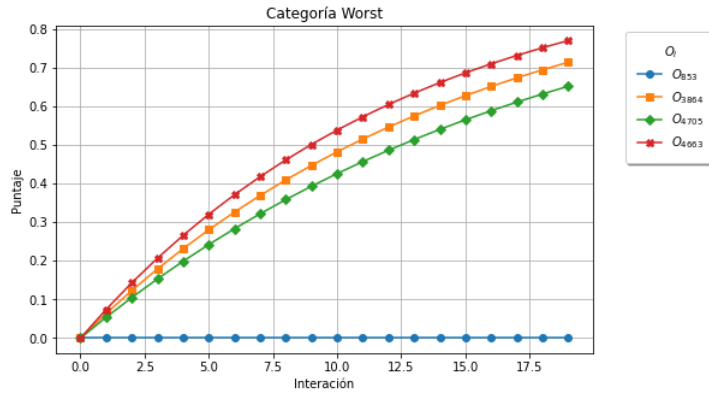


Figura 10.17: Variación del valor Q en la categoría Worst

En la Fig. 10.17 se muestran las combinaciones que se ven afectadas positivamente al seleccionar O_i . Estas combinaciones aunque pertenecen a la categoría *Worst*, son similares a la combinación seleccionada, como se evidencia en la Fig.10.18. La recompensa proporcionada, depende del valor Q actual y la distancia, por lo tanto, aunque son premiadas no alcanzan a llegar a 1, sin embargo, su puntaje después de las 20 iteraciones, es alto.



Figura 10.18: Representación visual de los conjuntos O_i de la categoría Worst

En la Fig.10.19 se rastrea el cambio de las O_i similares a O_u . Como se muestra en la Fig.10.19, no solo O_u es recompensada. Todo *fashion recommendation* que pertenezca, o sea similar a la selección del usuario se ve beneficiado.

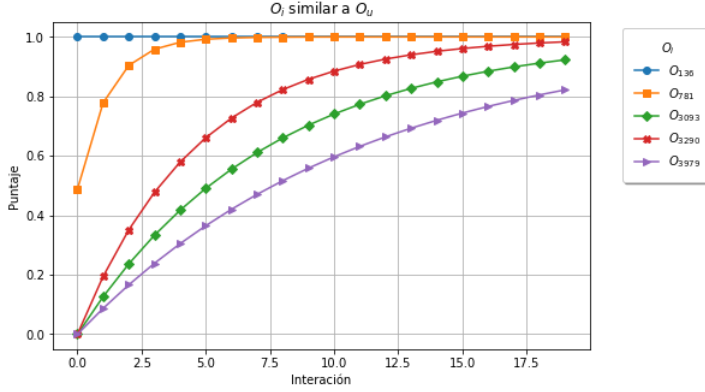


Figura 10.19: Variación del valor Q en los O_i similares a la selección del usuario (O_u)

En la Fig.10.20 se muestran las combinaciones que representan a cada O_i en la Fig.10.19. Esta imagen se presenta con el propósito de corroborar que las combinaciones que el sistema de recomendación identifica como combinaciones similares que satisfagan al usuario según su preferencia sean correctas.



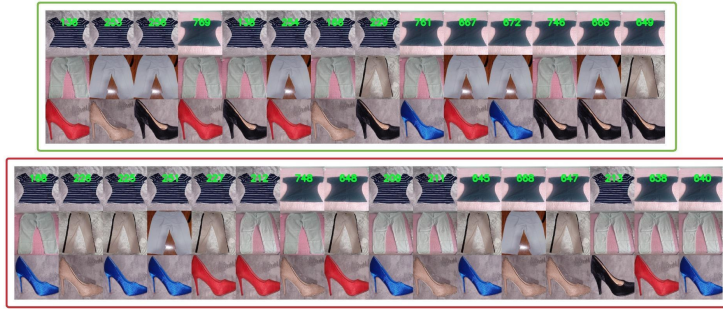
Figura 10.20: Representación visual de los conjuntos O_i similares a O_u

En las figuras Fig.10.15, Fig.10.17, Fig.10.13, y Fig.10.19 se contempla de forma visual la variación del ranking. Generado por la actualización de los valores Q cada vez que el usuario interactúa con el sistema.

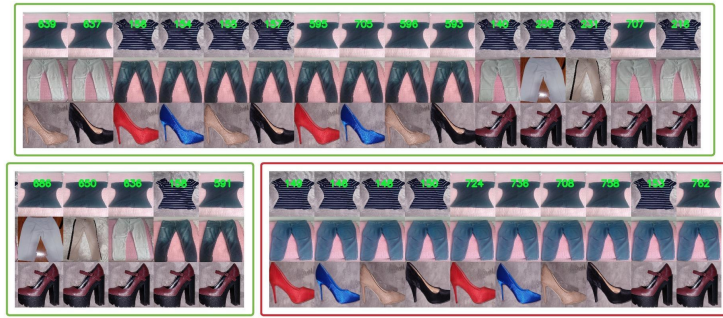
Como se menciona en la sección 9.0.4 después de cierta cantidad interacciones el modelo construye nuevo *ranking table* basada en el perfil

del individuo. En este caso después de las 20 interacciones se genera la nueva tabla.

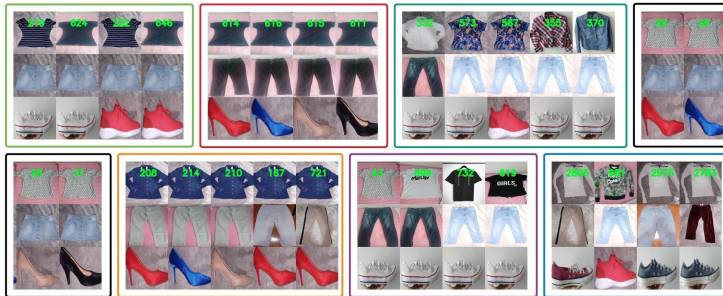
Tras la interacción del usuario con el sistema, el tablero de recomendaciones cambia tal y como se muestra en la Fig.10.21. Ahora, este tablero se basa totalmente en las preferencias del usuario cumpliendo con las expectativas y con el objetivo del experimento.



(a) Categoría Best



(b) Categoría No-select



(c) Categoría Worst

Figura 10.21: Tablero de recomendaciones adaptado a las preferencias del usuario donde muestra los diferentes estilos de las categorías *best* Fig.a, *No – select* Fig.b y *Worst* Fig.c

10.3. Experimento C

En el tercer experimento realizado, el usuario selecciona 5 recomendaciones de moda que pertenecen a diferentes categorías con diferentes estilos, alternándose entre cada interacción. Donde O_{169} pertenece a la categoría *No – select*, O_8 pertenece a la categoría *Best*, O_{42} es de *No – select*, O_{14} pertenece a *Best*, y O_{853} pertenece a la categoría *Worst*. Esto con el fin de verificar el proceso de adaptación del sistema a las preferencias del usuario teniendo en cuenta todos los estilos de cada recomendación seleccionada.

Las recomendaciones seleccionadas (O_u) se presentan en la Fig.10.22



Figura 10.22: Historial de selecciones del usuario, en este caso 5 O_u del experimento C

En la Fig.10.23 se ve la variación del valor Q de cada O_i que representan los clusters de la categoría *Worst*. Esto con el propósito de identificar como las selecciones de O_u afecta a los demás O_i en cada categoría. Los picos que existen en cada una de las O_i graficadas en la Fig.10.23 evidencian como las prendas son castigadas cuando se selecciona un O_i lejano a estas.

En el experimento c la selección del usuario no es una sola combinación en todas las iteraciones. En este caso se seleccionan 5 diferentes combinaciones alternando cada una entre las 20 interacciones. Cada vez que se selecciona una *fashion recommendation* el algorithmo premia el valor Q de esa combinación junto con las O_i cercanas según la distancia

como se explica en la seccion 9.0.4. Es por esto, que cuando la combinación es diferente, el valor Q de la combinación que en su momento aumento, se afecta negativamente junto con las O_i similares a esta. Como se mencionó al inicio de la sección 10.3, el experimento se realizó con combinaciones de diferentes categorías. En el caso de la categoría *Worst* es la combinación O_{853} , en la gráfica 10.23 se evidencia como el valor Q de esa combinación tiene picos de subida, representan las veces en que esta combinación fue seleccionada. Así mismo los picos de bajada son las veces en que otra de las cuatro restantes combinaciones Fig.10.22 fueron seleccionadas y esta combinación O_{853} es castigada.

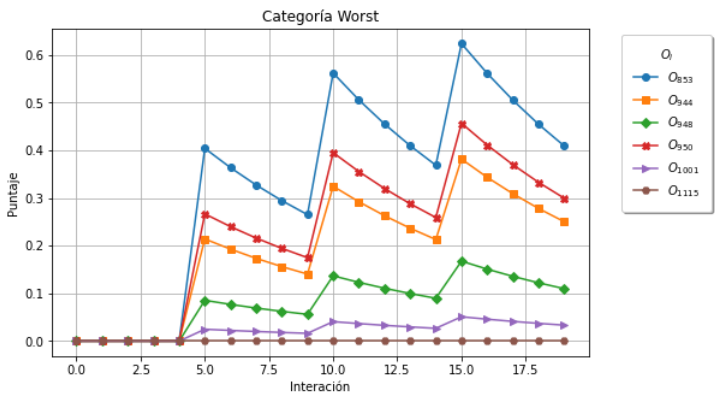


Figura 10.23: Variación del valor Q en la categoría Worst

La Fig.10.24 muestra cada una de las combinaciones pertenecientes a cada O_i utilizadas en la gráfica de variación del valor Q Fig.10.23. La combinación O_{853} es una de las O_u , es decir la combinación fue seleccionada por el usuario (ver Fig.10.22). La Fig.10.24 evidencia que las combinaciones O_{1001} y O_{1115} que no reciben grandes recompensas (ver Fig.10.23) son diferentes a la O_u . Esto comprueba que la dinámica de adaptación del sistema tiene lógica.



Figura 10.24: Representación visual de los conjuntos O_i de la categoría *Worst*

En la Fig.10.25 se muestran los O_i que representa la categoría *Best*. Las combinaciones seleccionadas en este experimento que pertenecen a la categoría *Best* son las O_8 y O_{14} en la gráfica 10.25 se evidencia como el valor Q de esas combinaciones tiene picos de subida. Los picos representan las veces en que fueron seleccionadas. Así mismo los picos de bajada son las veces en que otra de las *Fashion recommendation* restantes (ver Fig.10.27) fueron seleccionadas generando que las combinaciones (O_8 y O_{14}) sean castigadas junto con todas las O_i similares a estas.

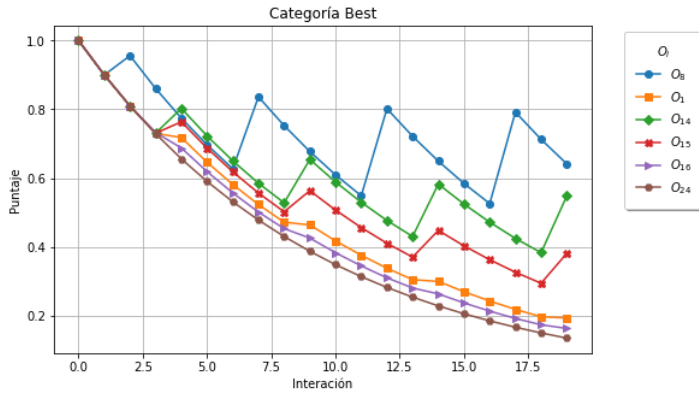


Figura 10.25: Variación del valor Q en la categoría *Best*

La Fig.10.26 se muestra cada una de las combinaciones pertenecientes a cada O_i utilizadas en la gráfica de variación del valor Q Fig.10.25. La Fig.10.26 evidencia que las combinaciones que su puntaje (valor Q)

disminuye (ver Fig.10.25) es debida a la diferencia respecto a las O_u .



Figura 10.26: Representación visual de los conjuntos O_i de la categoría Best

También, diferentes combinaciones O_i dentro de la categoría *No – select* se muestran en la Fig.10.27. En ambas figuras Fig.10.27 y Fig.10.25, se expone cómo el sistema varía la Q de cada O_i . La forma en que cambian los valores de Q generando los picos demuestra cómo el sistema se adapta a las preferencias del individuo cada vez que se selecciona una *fashion recommendation*. Las *fashion recommendation* seleccionadas de la categoría *No – select* son O_{169} y O_{42}

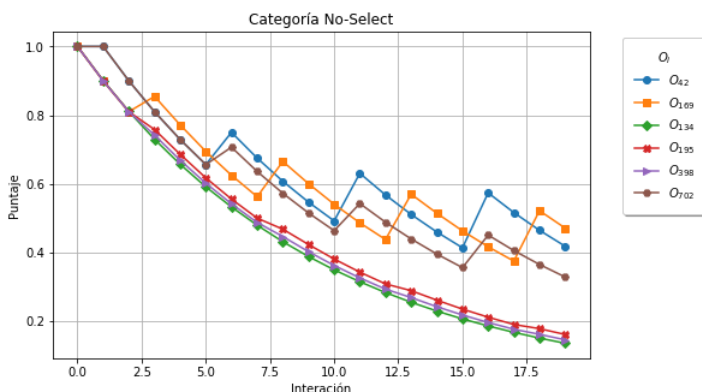


Figura 10.27: Variación del valor Q en la categoría No-select

La Fig.10.28 evidencia que las combinaciones castigadas (ver Fig.10.27) difieren de estilo respecto a las O_u . También que a pesar de

O_{42} hacer parte de las recomendaciones seleccionadas al elegir otra entre las 5 O_u se ve afectada.

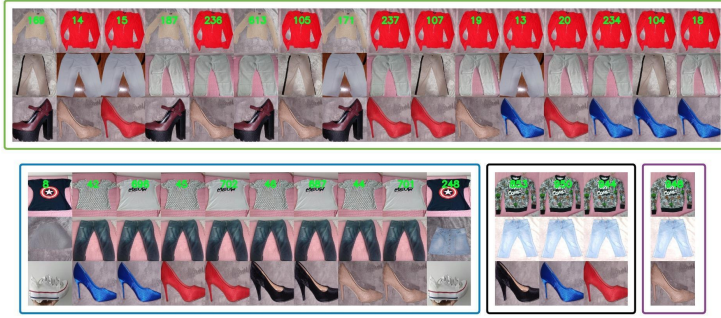


Figura 10.28: Representación visual de los conjuntos O_i de la categoría No-select

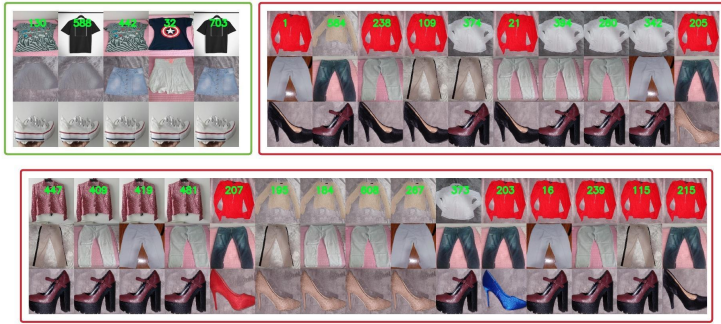
En las figuras Fig.10.25, Fig.10.23 y Fig.10.27 se contempla de forma visual la variación del ranking. Generado por la actualización de los valores Q cada vez que el usuario interactúa con el sistema.

Como se menciona en la sección 9.0.4, después de las 20 interacciones se genera la nueva *ranking table*.

Tras la interacción del usuario con el sistema, el tablero de recomendaciones cambia tal y como se muestra en la Fig.10.29. Ahora, este tablero se basa totalmente en las preferencias del usuario. Como se observa, esta nueva versión del tablero está marcado por los estilos de las O_i seleccionados al inicio del experimento, como se muestra en la Fig.10.22. Demostrando que el sistema es capaz de deducir y adaptarse a todos los estilos preferentes por el usuario adaptándose a ellos y recomendando opciones similares.



(a) Categoría Best



(b) Categoría No-select



(c) Categoría Worst

Figura 10.29: Tablero de recomendaciones adaptado a las preferencias del usuario donde muestra los diferentes estilos de las categorías *Best* Fig.a, *No – select* Fig.b y *Worst* Fig.c

10.4. Experimento D

En el cuarto experimento realizado, el sistema se implementa durante 30 iteraciones. El usuario selecciona 3 recomendaciones de moda cada una perteneciente a diferente categoría. El objetivo de este experimento es demostrar que el sistema se adapta no solo a un estilo sino a los 3.



Figura 10.30: Selección del usuario O_u del experimento D

La recomendación de moda seleccionada (O_u) utilizada se presenta en la Fig. 10.30

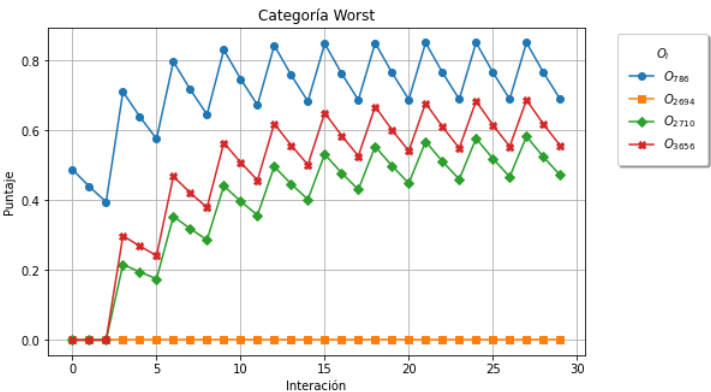


Figura 10.31: Variación del valor Q en la categoría Worst

En la Fig.10.31 se ve la variación del valor Q de cada O_i que representan los clusters de la categoría *Worst*. Esto con el propósito de

identificar como la selección de O_u afecta a los demás O_i de la misma categoría. Los picos que existen en cada una de las O_i graficadas en la Fig.10.31 evidencian como las prendas son castigadas cuando se selecciona un O_i lejano a estas.

En el experimento D la selección del usuario no es una sola combinación en todas las iteraciones. En este caso se seleccionan 3 diferentes combinaciones alternando cada una entre las 30 interacciones. Cada vez que se selecciona una *Fashion recommendation* el algoritmo premia el valor Q de esa combinación junto con las O_i cercanas según la distancia como se explica en la sección 9.0.4. Es por esto, que cuando la combinación es diferente, el valor Q de la combinación que en su momento aumento, se afecta negativamente junto con las O_i similares a esta.

Como se mencionó al inicio de la sección 10.3, el experimento se realizó con combinaciones de diferentes categorías. En el caso de la categoría *Worst* es la combinación O_{786} , en la gráfica 10.31 se evidencia como el valor Q de esa combinación tiene picos de subida, representan las veces en que esta combinación fue seleccionada. Así mismo los picos de bajada son las veces en que otra de las cuatro restantes combinaciones Fig.10.30 fueron seleccionadas y esta combinación O_{786} es castigada.



Figura 10.32: Representación visual de los conjuntos O_i de la categoría Worst

En la Fig.10.32 se muestra cada una de las combinaciones que representan a cada O_i graficados en la Fig.10.31. Donde la mayoría de

combinaciones en este caso como O_{2710} y O_{3356} son bastante similares respecto a O_{786} , ya que comparten la misma falda y zapatos. Es por esta razón que en la gráfica Fig.10.31 el valor Q de estas combinaciones aumenta más rápido comparado con la combinación O_{2694} . Esta diferencia es debido a que el sistema calcula menor similitud entre O_{2694} y O_{786} , ya que del grupo es la única que tiene zapatos de color diferente.

En la Fig.10.33 y en la Fig.10.35 se muestran las combinaciones O_i de las categorías *Best* y *No-select*, respectivamente. En ambas figuras, se expone cómo el sistema varía la Q de cada O_i . La forma en que cambian los valores de Q demuestra cómo el sistema se adapta a las preferencias del individuo. La forma en que cambian los valores de Q generando los picos demuestra

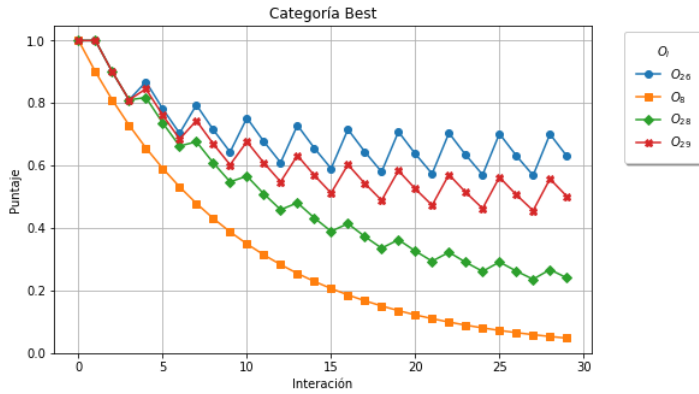


Figura 10.33: Variación del valor Q en la categoría Best

cómo el sistema se adapta a las preferencias del individuo cada vez que se selecciona una *fashion recommendation*. La *fashion recommendation* seleccionada de la categoría *Best* es O_{26}

En la Fig.10.34 se muestra cada una de las combinaciones que representan a cada O_i graficados en la Fig.10.33. Si se compara las combinaciones O_{28} y O_{29} son similares, ya que tienen la misma camisa y falda respecto a O_{26} por esta razón en la gráfica Fig.10.33 el valor Q de estas combinaciones disminuye más lento comparado con la combinación O_8 . Esta diferencia es debido a que el sistema ve mayor similitud entre O_{28} y O_{29} .



Figura 10.34: Representación visual de los conjuntos O_i de la categoría Best

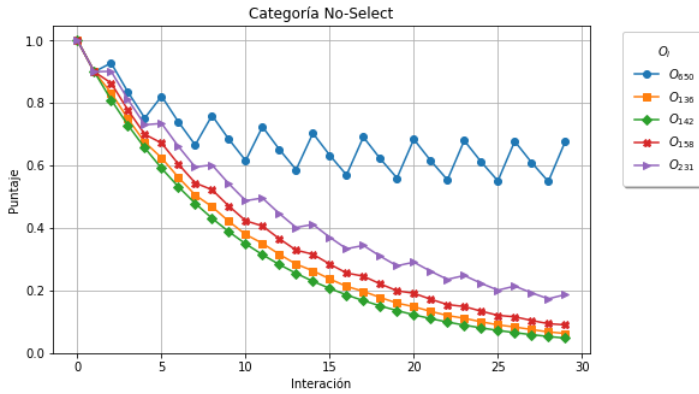


Figura 10.35: Variación del valor Q en la categoría No-select

En la Fig.10.36 se muestra cada una de las combinaciones que representan a cada O_i graficados en la Fig.10.35. Si nos detallamos O_{650} pertenece a las combinaciones seleccionadas Fig.10.30. Donde las combinaciones O_{136} , O_{142} , y O_{158} son bastante diferentes respecto a O_{650} por esta razón en la gráfica Fig.10.35 el valor Q de estas combinaciones disminuye más rápido comparado con la combinación O_{231} . Esta diferencia es debido a que el sistema ve mayor similitud entre O_{231} y O_{650} .

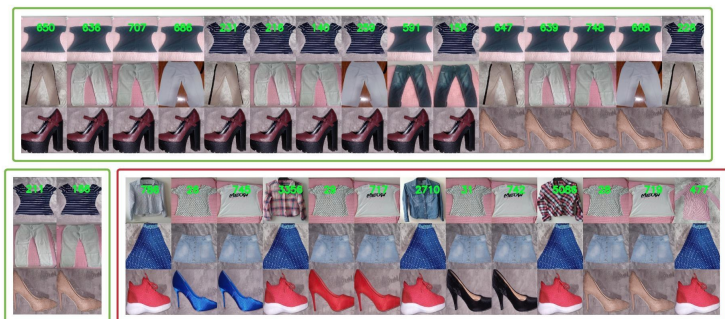


Figura 10.36: Representación visual de los conjuntos O_i de la categoría No-select

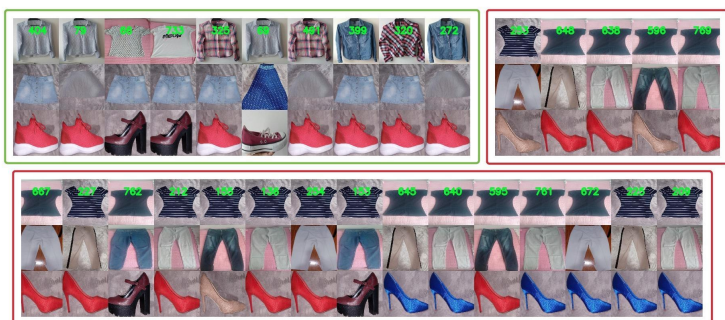
En las figuras Fig.10.33, Fig.10.31, y Fig.10.35 se contempla de forma visual la variación del ranking. Generado por la actualización de los valores Q cada vez que el usuario interactúa con el sistema.

Después de las 20 interacciones se genera la nueva tabla, como se menciona en la sección 9.0.4.

Tras la interacción del usuario con el sistema, el tablero de recomendaciones cambia tal y como se muestra en la Fig.10.37. Ahora, este tablero se basa totalmente en las preferencias del usuario. Como se observa, esta nueva versión del tablero está marcado por los estilos de las O_i seleccionados al inicio del experimento, como se muestra en la Fig.10.30. Demostrando que el sistema es deduce e incluye todos los estilos preferentes por el usuario y brinda recomendaciones similares. Concluyendo que el sistema se adapta satisfactoriamente al usuario.



(a) Categoría Best



(b) Categoría No-select



(c) Categoría Worst

Figura 10.37: Tablero de recomendaciones adaptado a las preferencias del usuario donde muestra los diferentes estilos de las categorías *Best* Fig.a, *No – select* Fig.b y *Worst* Fig.c

Capítulo 11

Resultados

En este trabajo se propone el diseño de un sistema de recomendación de moda mediante el procesamiento de información visual. El sistema de extracción de características utilizado para transformar las imágenes en vectores que describen las prendas (descriptor), está basado en CNN, que reconoce el tipo de prenda, y un algoritmo de clustering de k-means para la extracción de color. El estado inicial del sistema de recomendación se basa en el etiquetado profesional, y garantiza que el sistema empiece a generar recomendaciones de moda basadas en la opinión de un experto. Las recomendaciones se adaptarán gradualmente al estilo de moda del usuario a través de las interacciones usuario-sistema.

La capacidad de adaptación del sistema al estilo de moda del usuario se propuso mediante dos módulos: El primer módulo desarrolla una memoria a corto plazo que se adapta diariamente con cada interacción del usuario. El segundo desarrolla una memoria a largo plazo basada en una DNN encargada de aprender de la tabla de clasificación cada cierto tiempo. Las recomendaciones de moda se dividen en tres categorías: Best, No-select y Worst. La categoría de ropa (worst) tiene las puntuaciones más bajas, pero puede ser una opción de exploración para el usuario.

El sistema propuesto se evalúa utilizando imágenes de la base de datos de prendas de vestir y con una selección de conjuntos preestablecida por un profesional. Los experimentos consisten en seleccionar O_i pertenecientes a diferentes categorías. En el primero se utiliza una

recomendación perteneciente al grupo Worst. En el segundo se utiliza solo una recomendación del grupo No-select. En el tercero se escogen 5 recomendaciones de diferentes categorías. Estos tres se realizan durante 20 interacciones con el usuario. En el último experimento se seleccionó una recomendación de cada categoría durante 30 interacciones. Con el objetivo de comprobar si el sistema de recomendación consigue adaptarse completamente al nuevo estilo de moda del usuario. Las figuras 10.3, 10.5, 10.7, 10.9 demuestran cómo cambia la evaluación de los diferentes O_i al seleccionar el usuario siempre el mismo "peor conjunto". Las figuras 10.17, 10.15, 10.13, y 10.19 muestran cómo cambia el valor Q de los diferentes O_i al seleccionar el usuario siempre el mismo conjunto de la categoría No-select. Las figuras 10.23, 10.25, 10.27, 10.31, 10.33, y 10.35 muestran cómo cambia la evaluación de los diferentes conjuntos O_i al seleccionar varias recomendaciones. La figura 10.1 expone el estado inicial del sistema y las figuras 10.11, 10.21, 10.29, y 10.37 muestran el estado final de las recomendaciones, respectivamente. Los resultados muestran que el sistema se adapta completamente a los nuevos estilos de moda y las pruebas de esta adaptación son las figuras 10.11, 10.21, 10.29, y 10.37. Además, es evidente cómo la tabla de clasificación se adapta gradualmente a las nuevas pautas.

En conclusión, el sistema cumple con los requisitos de un sistema de recomendación totalmente adaptable al estilo del usuario **destacando cuatro contribuciones diferentes**. En primer lugar, un sistema de recomendación basado totalmente en la información visual. En segundo lugar, un conjunto de recomendaciones basado en la experiencia previa del usuario y en la elaboración de perfilamiento profesional. En tercer lugar, un sistema de recomendación adaptable a cualquier experiencia del usuario. Finalmente, el desarrollo también aporta un sistema de recomendación basado en memorias de largo y corto plazo que interactúa para mejorar la experiencia del usuario. Por otra parte dentro de las cosas a mejorar dentro del desarrollo del sistema es la adquisición de más características específicas de la prenda como el tipo de estampado entre otras. Adicionalmente extender el tipo de etiquetas, sin limitarse a 7. El trabajo futuro prevé ampliar el sistema a un enfoque multiusuario que

permita compartir estilos e intentar generalizar las tendencias de moda de un grupo concreto. Este nuevo enfoque ayudaría a centrarse en las decisiones de diseño de moda y, muy posiblemente, a reducir los costes de producción de nuevas prendas al disponer de información tan precisa sobre el estilo de moda de una población concreta.

Bibliografía

- [1] “Global fashion statics,” <https://fashionunited.com/global-fashion-industry-statistics/>, 2021, accessed: 2021-03-20.
- [2] “Online marketing trends -clothing/fashion. available online:,” 2019. [Online]. Available: <http://www.econovill.com/news/ArticleView.html?idxno=376137>
- [3] X. Li, X. Wang, X. He, L. Chen, J. Xiao, and T.-S. Chua, “Hierarchical fashion graph network for personalized outfit recommendation,” in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 159–168.
- [4] C. Quintero, S. Rodríguez, K. Pérez, J. López, E. Rojas, and J. Calderón, “Learning soccer drills for the small size league of robocup,” in *Robot Soccer World Cup*. Springer, 2014, pp. 395–406.
- [5] S. Rodríguez, E. Rojas, K. Pérez, J. López, C. Quintero, and J. Calderón, “Fast path planning algorithm for the robocup small size league,” in *Robot Soccer World Cup*. Springer, 2014, pp. 407–418.
- [6] G. A. Cardona, W. Moreno, A. Weitzenfeld, and J. M. Calderon, “Reduction of impact force in falling robots using variable stiffness,” in *SoutheastCon 2016*. IEEE, 2016, pp. 1–6.
- [7] E. Elibol, J. Calderon, M. Llofriu, C. Quintero, W. Moreno, and A. Weitzenfeld, “Power usage reduction of humanoid standing process using q-learning,” in *Robot Soccer World Cup*. Springer, 2015, pp. 251–263.

- [8] E. Elibol, J. Calderon, M. Llofriu, W. Moreno, and A. Weitzenfeld, "Analyzing and reducing energy usage in a humanoid robot during standing up and sitting down tasks," *International Journal of Humanoid Robotics*, vol. 13, no. 04, p. 1650014, 2016.
- [9] J. Calderon, G. A. Cardona, M. Llofriu, M. Shamsi, F. Williams, W. Moreno, and A. Weitzenfeld, "Impact force reduction using variable stiffness with an optimal approach for falling robots," in *Robot World Cup*. Springer, 2016, pp. 404–415.
- [10] J. M. Calderon, E. R. Rojas, S. Rodriguez, H. R. Baez, and J. A. Lopez, "A robot soccer team as a strategy to develop educational initiatives," in *Latin American and Caribbean Conference for Engineering and Technology, Panama City, Panama*, 2012.
- [11] J. G. G. Marin, G. C. O. Díaz, A. F. T. Rodríguez, and É. C. C. Poveda, "Entorno pedagógico para la enseñanza en básica primaria mediante el uso de sistema robótico comercial," *Ingeniería*, vol. 26, no. 1, 2021.
- [12] H. Báez, K. Perez, E. Rojas, S. Rodriguez, J. López, C. Quintero, and J. M. Calderón, "Application of an educational strategy based on a soccer robotic platform," in *2013 16th International Conference on Advanced Robotics (ICAR)*. IEEE, 2013, pp. 1–6.
- [13] C. Higuera, F. Lozano, E. C. Camacho, and C. H. Higuera, "Multiagent reinforcement learning applied to traffic light signal control," in *International Conference on Practical Applications of Agents and Multi-Agent Systems*. Springer, 2019, pp. 115–126.
- [14] G. A. Cardona and J. M. Calderon, "Robot swarm navigation and victim detection using rendezvous consensus in search and rescue operations," *Applied Sciences*, vol. 9, no. 8, p. 1702, 2019.
- [15] G. A. Cardona, J. Ramirez-Rugeles, E. Mojica-Nava, and J. M. Calderon, "Visual victim detection and quadrotor-swarm coordination control in search and rescue environment." *International Journal of Electrical & Computer Engineering (2088-8708)*, vol. 11, no. 3, 2021.

- [16] J. León, G. A. Cardona, A. Botello, and J. M. Calderón, “Robot swarms theory applicable to seek and rescue operation,” in *International Conference on Intelligent Systems Design and Applications*. Springer, 2016, pp. 1061–1070.
- [17] J. Calderon, A. Obando, and D. Jaimes, “Road detection algorithm for an autonomous ugv based on monocular vision,” in *Electronics, Robotics and Automotive Mechanics Conference (CERMA 2007)*. IEEE, 2007, pp. 253–259.
- [18] J. G. Guarnizo and L. F. Niño, “Clonal selection algorithm applied to object recognition in mobile robots.” in *MLDM (1)*, 2019, pp. 49–62.
- [19] S. Amaya and A. Mateus, “Tasks allocation for rescue robotics: a replicator dynamics approach,” in *International Conference on Artificial Intelligence and Soft Computing*. Springer, 2019, pp. 609–621.
- [20] Y. Suarez, C. Higuera, and E. C. Camacho, “Inverse reinforcement learning application for discrete and continuous environments,” in *International Conference on Advanced Engineering Theory and Applications*. Springer, 2019, pp. 345–355.
- [21] N. Squicciarino and J. L. A. Sánchez, *El vestido habla: consideraciones psico-sociológicas sobre la indumentaria*. Cátedra Barcelona, 1990.
- [22] “Informe del sistema moda noviembre 2020 – sala de prensa.” [Online]. Available: <http://www.saladeprensainexmoda.com/informe-noviembre/>
- [23] “Informe del sistema moda – diciembre 2019 – sala de prensa,” Feb 2020. [Online]. Available: <http://www.saladeprensainexmoda.com/informe-del-sistema-moda-diciembre-2019-textil/>
- [24] K. Zielnicki, “Simulacra and selection: Clothing set recommendation at stitch fix,” in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2019, pp. 1379–1380.

- [25] “Project muze: Fashion inspired by you, designed by code,” 2019. [Online]. Available: <https://blog.google/around-the-globe/google-europe/project-muze-fashion-inspired-by-you/>
- [26] R. Díaz Badra, “Clasificador de vestimenta basado en redes neuronales,” B.S. thesis, 2018.
- [27] Y. Liang, S.-H. Lee, and J. E. Workman, “Implementation of artificial intelligence in fashion: Are consumers ready?” *Clothing and Textiles Research Journal*, vol. 38, no. 1, pp. 3–18, 2020.
- [28] G. Shani and A. Gunawardana, “Evaluating recommendation systems,” in *Recommender systems handbook*. Springer, 2011, pp. 257–297.
- [29] J. Bobadilla, A. Hernando, F. Ortega, and J. Bernal, “A framework for collaborative filtering recommender systems,” *Expert Systems with Applications*, vol. 38, no. 12, pp. 14 609–14 623, 2011.
- [30] K. Christidis and G. Mentzas, “A topic-based recommender system for electronic marketplace platforms,” *Expert Systems with Applications*, vol. 40, no. 11, pp. 4370–4379, 2013.
- [31] A. D. R. Oliveira, L. N. Bessa, T. R. Andrade, L. V. L. Filgueiras, and J. S. Sichman, “Trust-based recommendation for the social web,” *IEEE Latin America Transactions*, vol. 10, no. 2, pp. 1661–1666, 2012.
- [32] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, “Collaborative filtering recommender systems,” in *The adaptive web*. Springer, 2007, pp. 291–324.
- [33] A. Sánchez-Contador Uría, “La identidad a través de la moda,” *Revista de Humanidades*, vol. 29, pp. 131–152, 2016.
- [34] D. Le Breton, *La sociología del cuerpo*. Siruela, 2018, vol. 99.
- [35] A. Lurie, *The language of clothes*. Random House Inc, 1981.
- [36] E. C. Hirschman and M. B. Holbrook, “Hedonic consumption: emerging concepts, methods and propositions,” *Journal of marketing*, vol. 46, no. 3, pp. 92–101, 1982.

- [37] J. Entwistle, *El cuerpo y la moda: una visión sociológica*. Paidós Barcelona, 2002.
- [38] M. N. Casá, “La mirada semiótica de la moda: la dimensión comunicativa del vestido,” B.S., Facultad de Ciencia Política y Relaciones Internacionales, 2019.
- [39] P. CHACÓN and L. CASABLANCA, “El hombre vestido. una visión sociológica, psicológica y comunicativa sobre la moda.” *Cartaphilus*, no. 13, 2014.
- [40] M. Margulis and M. Urresti, “Moda y juventud,” *Estudios sociológicos*, pp. 109–120, 1995.
- [41] S. Monrós-Sánchez, “Cómo educar la autoestima en educación infantil,” B.S., 2013.
- [42] F. Sancho Caparrini, “Introducción al aprendizaje automático,” *Recuperado de [http://www. cs. us. es/~ fsancho](http://www.cs.us.es/~fsancho)*, 2017.
- [43] P. Pury, “Fundamentos de aprendizaje automático,” *Córdoba, España. Recuperado de [http://www. famaf. proed. unc. edu. ar/pluginfile. php/19002/mod_ resource/content/2/01. introduccion. pdf](http://www.famaf.proed.unc.edu.ar/pluginfile.php/19002/mod_resource/content/2/01.introduccion.pdf)*, 2015.
- [44] K. O’Shea and R. Nash, “An introduction to convolutional neural networks,” *arXiv preprint arXiv:1511.08458*, 2015.
- [45] S. Khan, Rahmani and Bennamoun, *A Guide to Convolutional Neural Networks for Computer Vision*, ser. Synthesis Lectures on Computer Vision. Morgan and Claypool Publishers, 2018.
- [46] W. SARLE, “Neural networks and statistical models,” in *Proceedings Of The 19th Annual SAS Users Group International Conference*. SAS Institute, 1994, pp. 1538–1550.
- [47] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

- [48] D. López and A. Fernández, “Aplicación en los medios de prensa de un agrupamiento k-means(clustering k-means),” *Revista chilena de economía y sociedad.*, vol. 12, no. 1, 2018.
- [49] J. Pérez, M. Henriques, R. Pazos, L. Cruz, G. Reyes, J. Salinas, and A. Mexicano, “Mejora al algoritmo de agrupamiento k-means mediante un nuevo criterio de convergencia y su aplicación a bases de datos poblacionales de cáncer,” *Liver-2do Taller Latino Iberoamericano de Investigacion de Operaciones “la IO aplicada a la solución de problemas regionales*, pp. 1–7, 2007.
- [50] W. O. Quesada Moncayo *et al.*, “Generación de comportamientos de enjambre en robots móviles a través del uso del aprendizaje por refuerzo.” 2019.
- [51] L. Sharma and A. Gera, “A survey of recommendation system: Research challenges,” *International Journal of Engineering Trends and Technology (IJETT)*, vol. 4, no. 5, pp. 1989–1992, 2013.
- [52] A. Bhardwaj, V. Jagadeesh, W. Di, R. Piramuthu, and E. Churchill, “Enhancing visual fashion recommendations with users in the loop,” *arXiv preprint arXiv:1405.4013*, 2014.
- [53] W.-C. Kang, C. Fang, Z. Wang, and J. McAuley, “Visually-aware fashion recommendation and design with generative image models,” in *2017 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2017, pp. 207–216.
- [54] V. Jagadeesh, R. Piramuthu, A. Bhardwaj, W. Di, and N. Sundaresan, “Large scale visual recommendations from street fashion images,” in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 1925–1934.
- [55] W. Li and B. Xu, “Aspect-based fashion recommendation with attention mechanism,” *IEEE Access*, vol. 8, pp. 141 814–141 823, 2020.

- [56] Y. Li, L. Cao, J. Zhu, and J. Luo, “Mining fashion outfit composition using an end-to-end deep learning approach on set data,” *IEEE Transactions on Multimedia*, vol. 19, no. 8, pp. 1946–1955, 2017.
- [57] Y. Lin, P. Ren, Z. Chen, Z. Ren, J. Ma, and M. de Rijke, “Improving outfit recommendation with co-supervision of fashion generation,” in *The World Wide Web Conference*, 2019, pp. 1095–1105.
- [58] D. Verma, K. Gulati, and R. R. Shah, “Addressing the cold-start problem in outfit recommendation using visual preference modelling,” in *2020 IEEE Sixth International Conference on Multimedia Big Data (BigMM)*. IEEE, 2020, pp. 251–256.
- [59] W. Yu, H. Zhang, X. He, X. Chen, L. Xiong, and Z. Qin, “Aesthetic-based clothing recommendation,” in *Proceedings of the 2018 world wide web conference*, 2018, pp. 649–658.
- [60] L. De Divitiis, F. Becattini, C. Baecchi, and A. Del Bimbo, “Garment recommendation with memory augmented neural networks,” *arXiv preprint arXiv:2012.06200*, 2020.
- [61] M. Hou, L. Wu, E. Chen, Z. Li, V. W. Zheng, and Q. Liu, “Explainable fashion recommendation: A semantic attribute region guided approach,” *arXiv preprint arXiv:1905.12862*, 2019.
- [62] C. Li, Z. Liu, M. Wu, Y. Xu, H. Zhao, P. Huang, G. Kang, Q. Chen, W. Li, and D. L. Lee, “Multi-interest network with dynamic routing for recommendation at tmall,” in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 2019, pp. 2615–2623.
- [63] A. Veit, B. Kovacs, S. Bell, J. McAuley, K. Bala, and S. Belongie, “Learning visual clothing style with heterogeneous dyadic co-occurrences,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4642–4650.
- [64] H. O. Alejandres-Sánchez, J. G. González-Serna, and N. González-Franco, “Efecto de explicaciones sobre la confianza del usuario

- en sistemas de recomendación sensibles al contexto,” *Ingeniería, investigación y tecnología*, vol. 17, no. 4, pp. 419–428, 2016.
- [65] W. Chen, P. Huang, J. Xu, X. Guo, C. Guo, F. Sun, C. Li, A. Pfadler, H. Zhao, and B. Zhao, “Pog: Personalized outfit generation for fashion recommendation at alibaba ifashion,” in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 2662–2670.
- [66] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.