Artículo de Investigación Original

Revista de investigación de sistemas e informática
17(2): 25-32 (2024)
DOI: https://doi.org/10.15381/risi.v17i2.28406

ISSN: 1815-0268 (Impreso) / ISSN: 1816-3823 (Electrónico)
Facultad de Ingeniería de Sistemas e Informática - UNMSM

Sistema recomendador de recursos académicos para estudiantes universitarios

Academic resource recommendation system for university students

Giancarlo Ricardo Silva Gomez
giancarlo.silva1@unmsm.edu.pe
Universidad Nacional Mayor de San Marcos, Lima, Perú

Roberto Sifuentes Marcelo
roberto.sifuentes@unmsm.edu.pe
Universidad Nacional Mayor de San Marcos, Lima, Perú

Recibido: 30/10/2024 - Aceptado: 22/11/2024 - Publicado: 31/12/2024

RESUMEN

La pandemia generó una mayor importancia a la educación online. Cientos de estudiantes se inscriben a cursos online ofertados por universidades u otras instituciones. Sin embargo, por la gran cantidad de opciones esto supone un problema al elegir el curso, además en el país existe un desbalance entre la oferta y demanda laboral. Las empresas están solicitando profesionales con más capacidades tecnológicas; por lo que el objetivo del sistema es mejorar el proceso de orientación académica a estudiantes universitarios: Recomendar cursos basándose en sus búsquedas. El sistema fue desarrollado según la metodología del ciclo de desarrollo de software que consta de 4 etapas: Análisis, Diseño, Desarrollo y Testeo, con una duración estimada de 2 meses, donde los datasets usados fueron 3 de la plataforma Kaggle basándose en plataformas de e-learning como Udemy. Para validar, se realizó una encuesta a 20 alumnos de último ciclo de la facultad de Ingeniería de Sistemas de la UNMSM porque estos alumnos están más capacitados en la carrera y tienen más definidas sus dudas sobre las competencias que desean adquirir. Los resultados mostraron que, como mínimo, un 75% de los encuestados considera que el sistema es útil; les ayuda en su orientación académica y les ahorra tiempo de búsqueda considerablemente cumpliendo con el objetivo de la investigación en un nivel considerable en la mayoría de encuestados.

Palabras clave: Ciclo de vida de software, Sistema de recomendación, Kaggle, e-learning, aprendizaje automático.

© Los autores. Este artículo es publicado por la Revista de investigación de sistemas e informática de la Facultad de Ingeniería de Sistemas e Informática de la Universidad Nacional Mayor de San Marcos. Este es un artículo de acceso abierto, distribuido bajo los términos de la licencia Creative Commons Atribución 4.0 Internacional (CC BY 4.0) [https://creativecommons.org/licenses/by/4.0/deed.es] que permite el uso, distribución y reproducción en cualquier medio, siempre que la obra original sea debidamente citada de su fuente original. Para mas información, por favor póngase en contacto con revistarisi.fisi@unmsm.edu.pe

ABSTRACT

The pandemic generated a greater importance to online education. Hundreds of students sign up for online courses offered by universities or other institutions. However, due to the large number of options this poses a problem when choosing the course, in addition in the country there is an imbalance between supply and demand for labor. Companies are requesting professionals with more technological skills; therefore, the objective of the system is to improve the process of academic orientation for university students: Recommend courses based on their searches. The system was developed according to the methodology of the software development cycle consisting of 4 stages: Analysis, Design, Development and Testing, with an estimated duration of 2 months, where the datasets used were 3 of the Kaggle platform based on e-learning platforms as Udemy. To validate, a survey was carried out on 20 students at the last cycle of the Faculty of Systems Engineering of the UNMSM because these students are more qualified in the career and have more definite doubts about the competences they wish to acquire. The results showed that at least 75 per cent of respondents found the system useful; helps them in their academic orientation and saves them considerable search time by meeting the research objective at a considerable level in most respondents.

Keywords: Software lifecycle, Recommendation system, Kaggle, e-learning, machine learning.

I. INTRODUCCIÓN

Actualmente, el rápido crecimiento de Internet en este milenio ha facilitado a la creación y acceso de una cantidad colosal de información para todas las personas de manera rápida y global. Un estudio nos indica que el 90% de todos los datos del mundo son posteriores a 2014 y su volumen se estima que se duplica cada 1,2 años (Iberdrola, 2019). La pandemia de COVID 19 también trajo una revolución en la forma de estudiar y trabajar de las personas: el estudio y trabajo online o remoto. Si bien estos conceptos ya se observaban desde antes; fue con la pandemia que se hicieron de manera masiva así las organizaciones estén o no preparadas e impulsó la nueva forma de educación en línea. Se calculó a fines del 2020, más de 900 universidades ofrecían más de 13000 cursos y un total de 180 millones de estudiantes nuevos se matriculaban (Shah,2020). Esto sin contar plataformas alternativas como Udemy, Platzi, freeCodeCamp, edX, etc. que brindan cursos online con profesionales de experiencia en la materia. Esta situación representa un problema para cualquier usuario que desee aprender; por la gran cantidad de cursos por internet y el riesgo que dicho material o ruta de aprendizaje no sea agradable para el estudiante en un futuro.

En el Perú, la educación virtual ha significado un medio para combatir la desigualdad de oportunidades y la baja calidad académica de la educación superior. Plataformas como Coursera tienen una gran demanda en el territorio. Según Gestión, Perú se encuentra en el sexto lugar de los países más activos de la plataforma que como meta continuar atrayendo a más profesionales peruanos, en respuesta a la alta demanda por programas profesionales en modalidad virtual. (Diario Gestión, 2024). Esto se debe al cambio en la demanda laboral peruana donde las organizaciones exigen habilidades relacionadas al mundo digital y resulta un criterio exclusivo en el proceso de selección en las empresas. Según el estado, existe una gran desarticulación con la oferta y demanda laboral donde el 75% de las empresas no pueden conseguir trabajadores con competencias digitales que le satisfagan. Además, indica que los institutos y universidades están capacitando en materias que la demanda no requiere (Ministerio de Trabajo y Promoción del Empleo, 2023).

Todo esto resulta un panorama complicado para los universitarios prontos a ingresar al mercado laboral: Deben capacitarse con cursos externos a su centro de estudios donde es muy probable que busquen y aprendan por internet ya que les permite la modalidad virtual de enseñanza. Pero se encontrarán con el gran mar de información de cursos y les generaría incomodidad y dificultades encontrar el curso adecuado según sus preferencias y metodología de aprendizaje. Todo este análisis explicado se muestra en el siguiente árbol de problemas (ver Figura 1).

La parte inferior vendrían a ser las causas del problema: “Baja calidad de enseñanza”, “Poco conocimiento de la carrera en que estudian”, “Diversa información en internet” que generan las siguientes consecuencias: “Baja calidad de los egresados”, “Disminución de la cantidad de egresados con empleo”, “Poca capacidad investigativa de cursos. Todo ello en base al problema principal el cual es que existe una escasa orientación académica a los universitarios que complementen su perfil profesional que se busca solucionar a través de la implementación de un sistema de recomendación de cursos académicos. El objetivo de estos sistemas es filtrar información según los gustos y elecciones del usuario para que al usuario le aparezca menos información, pero en función de sus decisiones y

gustos. Los sistemas de recomendación tienen ya tiempo siendo aplicados principalmente en el campo del entretenimiento como recomendación de películas, series o productos. El objetivo de este artículo es implementar una metodología emergente como los sistemas de recomendación para mostrar al usuario recursos educativos de calidad de su agrado para mejorar el proceso de orientación académica a estudiantes universitarios. Esto permite al usuario del sistema ahorrar tiempo de investigación y mostrándole recursos de buena calidad en internet para su desarrollo académico-laboral.

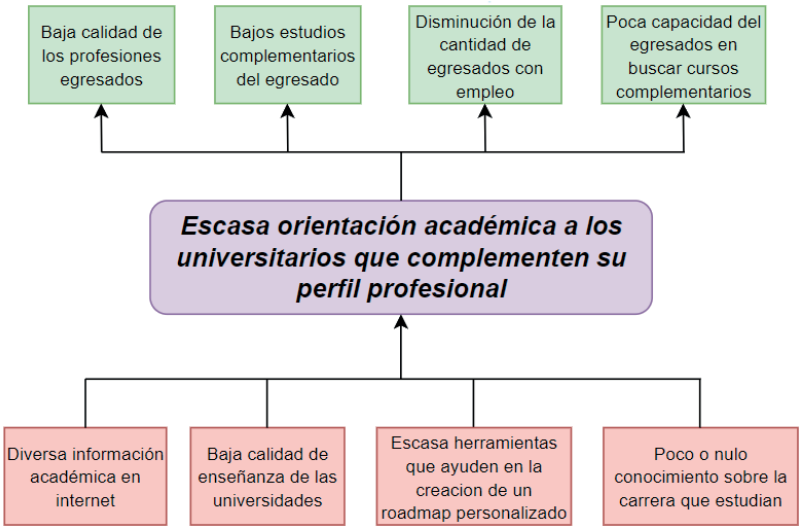
MATERIALES Y MÉTODOS

Propuesta general de la solución

El sistema de recomendación se alimenta de 3 datasets obtenidos en Kaggle con información de los cursos disponibles los cuales se dividen en cursos de: Udemy, Coursera y Otros que albergan de diversas plataformas (ver Figura 2). Se decidió separar los datasets según el tipo de plataforma debido que al realizar la limpieza y las pruebas el sistema no recomendó de manera eficiente además que

Figura 1

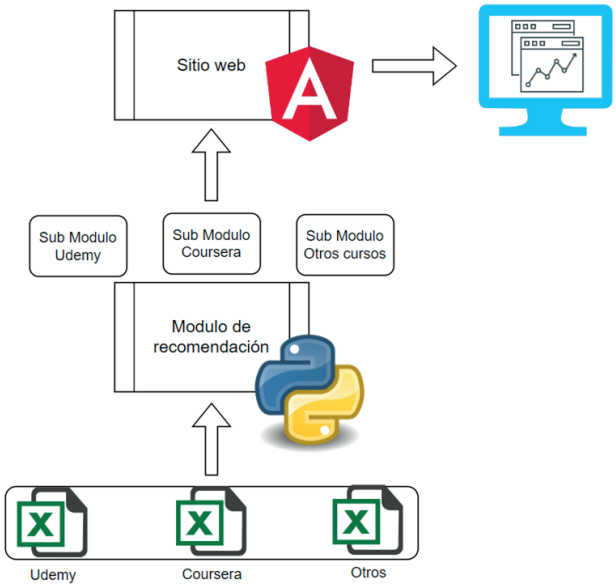
Árbol de problemas



Elaboración propia.

Figura 2

Diagrama general de la solución



Elaboración propia.

cada datasets tenía sus propios atributos y era difícil relacionarlo uno con otros

Luego los datasets pasan por un proceso de limpieza de data donde se eliminan duplicados y caracteres especiales para pasar al módulo de recomendación. El algoritmo para la recomendación de cursos será la similitud de cosenos a través de la librería de Python scikit-learn para el aprendizaje automático. Por ultimo los resultados serán mostrados en una página web desarrollada en Angular para el usuario final.

Metodología

No existe un estándar específico para implementar un sistema de recomendación para este campo; si existen algoritmos de recomendación como el filtrado colaborativo, similitud coseno, etc. No obstante, hay pasos generales para implementar un sistema de recomendación con estos pasos.

Recolección de datos

Como se mencionó anteriormente, la data utilizada para los cursos proviene de 3 datasets extraídos de la plataforma Kaggle y tienen una fecha de antigüedad menor a 5 años.

Tabla 1

Datasets a utilizar

Año	Nombre	Cantidad
2024	Udemy course recomendation	3678 registros
2021	Coursera Courses Dataset	3424 registros
2020	IT & Software Courses	8976 registros

Elaboración propia.

Preparación de datos

El módulo de recomendación fue desarrollado en Python y para la manipulación y limpieza de datos se utilizó la librería Pandas que es de código abierto. La preparación de datos consistió en los siguientes pasos:

- Para los 3 datasets se eliminan elementos duplicados
- Para la dataset de Udemy, se eliminaron los cursos no relacionados a la tecnología.
- Para el dataset de Coursera, se filtraron los campos más importantes como título, calificación, entre otros.
- En el dataset de Coursera y cursos diversos, se realizó un filtrado por idioma donde solo se analizarán cursos en ingles
- Por último, para el dataset de cursos diversos se realizó un cambio en el campo “Dificultad” ya que muchos cursos no tienen ese dato y se les puso “Indeterminado”
- El campo duración de los 3 datasets pasan a minutos para un mejor análisis del modelo de recomendación

Algoritmo de recomendación

El algoritmo elegido para la recomendación de cursos es el método de similitud coseno. Este algoritmo es una medida de similitud que se calcula entre dos vectores distintos de cero dentro del espacio interno del producto que mide el coseno del ángulo entre ellos; es usado en el aprendizaje automático y en el filtrado colaborativo en las recomendaciones. Este algoritmo se usará en los cursos para que muestre recomendaciones según un curso elegido por el usuario y será implementado con la librería scikit útil en escenarios de aprendizaje automático. En el módulo de recomendación, se pone cada campo de cada registro de los cursos en un array y se las agrega en una nueva columna “tags” donde será evaluada en el modelo con el atributo título.

Luego se pasa al modelo de recomendación basado en la similitud por coseno donde se utiliza la columna “tags” recién creada y la transforma en una matriz dispersa donde cada cada fila representa un registro en la tabla 'Cursos' y cada columna una palabra única. Por último, con la librería sklearn se aplica la similitud coseno en dicha matriz para obtener la matriz de similitud con los puntajes obtenidos y pueda ser usado en la función “recommend” para la recomendación del curso en base al input que vendría a ser un curso elegido (ver Figura 3).

Evaluacion del modelo

Unidad de Análisis

La unidad de análisis sería el ente que se estudiará en el trabajo de investigación, esta nos permitirá definir mejor la población y la muestra para el trabajo de investigación. En este caso, la unidad de análisis es el estudiante universitario de la FISI-UNMSM. Un estudiante activo que esté estudiando carreras relacionadas con la tecnología y el desarrollo de software.

Figura 3

Modulo de recomendación en Python

```python
#Funcion lambda la cual pone cada dato como un array
cursos['price'] = cursos['price'].apply(lambda x: [x])
cursos['num_subscribers'] = cursos['num_subscribers'].apply(lambda x: [x])
cursos['content_duration_in_minutes'] = cursos['content_duration_in_minutes'].apply(lambda x: [x])
cursos['published_timestamp'] = cursos['published_timestamp'].apply(lambda x: [x])

#Suman todas las columnas a una columna tags
cursos['tags'] = cursos['price'] + cursos['num_subscribers'] + cursos['content_duration_in_minutes']
+ cursos['published_timestamp'] + cursos['subject']

from sklearn.feature_extraction.text import CountVectorizer
from sklearn.metrics.pairwise import cosine_similarity

cv = CountVectorizer(max_features=5000,stop_words='english')
#Proceso de transformacion y devuelve una matriz dispersa
vector = cv.fit_transform(Cursos['tags']).toarray()
similarity = cosine_similarity(vector)

def recommend(course):
    course_index = Cursos[Cursos['course_title'] == course].index[0]
    #calcula la distancia entre cursos dados y la demmpas usando la matriz similarity
    #se almacenan distncias en lista distances
    distances = sorted(list(enumerate(similarity[course_index])),reverse=True,key = lambda x: x[1])
    #iteramos las primeras 10 cursos más similares
    for i in distances[1:11]:
        print(Cursos.iloc[i[0]].course_title)
```

Elaboración propia.

Población de Estudio

La población de estudio del trabajo de investigación son todos los estudiantes universitarios de la FISI-UNMSM del último año de carrera que estén estudiando carreras relacionadas en el campo de la tecnología y el desarrollo de software. Se escoge esta población para el trabajo porque son estudiantes que ya poseen conocimientos importantes sobre la carrera que estudian y tienen definido a que campo de acción quieren dedicarse; además tienen nociones básicas sobre los cursos que desean estudiar para ser más competentes en el mercado laboral. Caso contrario, los estudiantes de menor ciclo aún están aprendiendo los conceptos y aun no definen el campo de acción en el que quieren trabajar. La selección de esta población se amolda a los objetivos de este trabajo: Servir como guía académica al estudiante y que sea más competente en el mercado laboral.

Tamaño de la muestra

Según datos de la FISI-UNMSM, aproximadamente 400 alumnos se encuentran en el último año de la carrera; no obstante, la muestra no será el total, sino que se buscará una muestra representativa menor. Por lo que la muestra es de 20 estudiantes activos de la FISI-UNMSM en el último año de carrera.

Dataset y Organización

Los datasets fueron utilizados de 3 fuentes distintas por lo cual en cada uno se utilizaron ciertas variables pero relacionados.

Tabla 2

Metadata de los cursos a recomendar.

Variable	Descripción	Datasets
Título	Título del material	Todos
Duración	Duración en completar el material	Udemy
Palabras clave	Conjunto de habilidades que aprenderás	Udemy y Otros
Nivel	Grado de dificultad	Otros
Fecha	Fecha de publicación del curso	Todos
Puntaje	Calificación del recurso académico	Todos
Precio	Precio del curso (si lo tiene)	Todos
Suscriptores	Cantidad de suscriptores	Udemy y Coursera

Elaboración propia.

Diseño del experimento

- Variables
  - Variable Dependiente: Orientación académica
  - Variable Independiente: Sistema de recomendación de recurso académicos

- Tratamiento
  - Exposición breve sobre el sistema de recomendación: sus funciones y que problema busca solucionar.
  - Interacción de la muestra con el sistema de recomendación para poder responder la encuesta
  - Aplicación de la encuesta para evaluar su opinión del sistema y en qué grado lo ayuda como orientación académica.
- Instrumento
  - El instrumento para la experimentación será una escala de Likert de 4 puntos. Se eligió este tipo de validación ya que es usado en trabajos similares como: "Sistema inteligente web de recomendación de actividades turísticas para una provincia del Perú" (Castillo, Aguilar; 2019) que realiza una encuesta usando Likert de 5 preguntas evaluando la usabilidad y la calificación de la recomendación del sistema. A continuación, se mostrará la tabla con las preguntas y las 4 opciones de cada pregunta:

Tabla 3

Preguntas a usuarios

P1: ¿Qué tan diversas le parecen las recomendaciones del sistema?	No hay diversidad Esta OK Algo Diversas Muy diversas
P2: ¿Cuánto tiempo considera que le ahorra el sistema en buscar cursos?	Retrasa mucho Es igual Algo de tiempo Mucho tiempo
P3: ¿Qué le parece la calidad y difusión de la orientación académica en la facultad?	Muy Mala. Mala Algo buena. Muy buena.
P4. ¿Qué tan capacitado se considera usted para conseguir un empleo o practica?	Nada capaz Poco capaz Muy capaz Perfectamente capaz
P5. ¿Consideras que los cursos recomendados brindan una mejor orientación académica?	Nada mejora Normal Algo de mejora Gran mejora

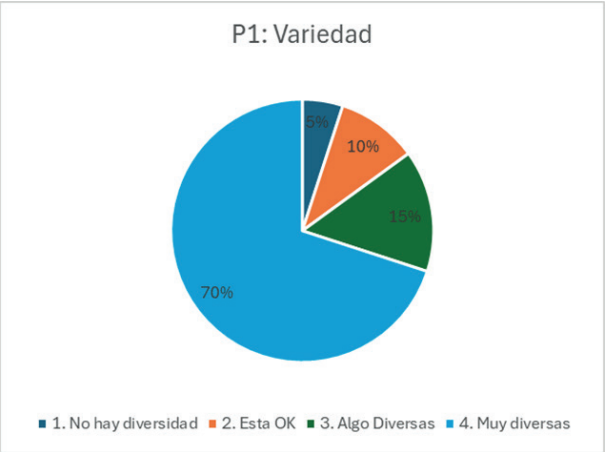
Elaboración propia.

RESULTADOS

Como se mencionó en la sección anterior, se realizó un cuestionario de 5 preguntas a 20 estudiantes del último ciclo. A continuación, se mostrarán los resultados de las encuestas

Figura 4

Resultados a la pregunta 1 de la encuesta



Elaboración propia.

Figura 5

Resultados a la pregunta 2 de la encuesta

Elaboración propia.

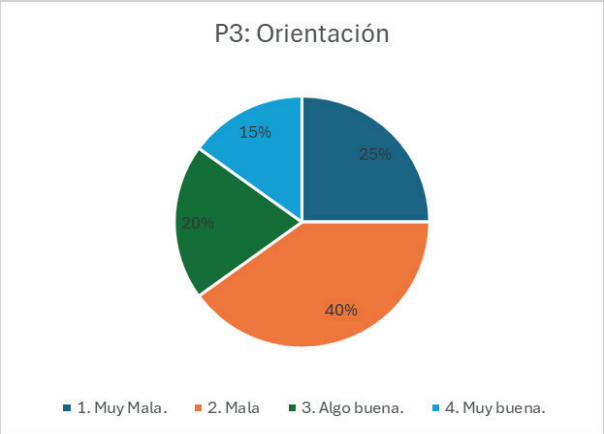
Los resultados de la pregunta 1 nos indica que un 85% de los encuestados considera que existe variedad en los cursos ofrecidos en el sistema justificando el uso de los 3 datasets usados en Kaggle. Mientras que la pregunta 2, un 75% de los encuestados considera que el sistema les ahorra tiempo en la búsqueda de cursos.

Respecto a la situación actual del alumno y la universidad, un 65% los encuestados consideran que la universidad tiene una mala orientación académica. En adición, un 55% de los encuestados no se sienten preparados para una práctica laboral; lo cual es preocupante en estudiantes del último año de carrera. Para cerrar, un 62% de encuestados indican que el sistema ayudó en su orientación académica mientras que un 28% indica que no tuvo cambios en su orientación pero que usaría el sistema constantemente. Por lo cual, se cumple el objetivo del artículo, pero no a un porcentaje esperado.

Para un aumento en el porcentaje, se buscará agregar más datos de cursos al sistema; además de añadir nuevos campos, para una recomendación más específica.

Figura 6

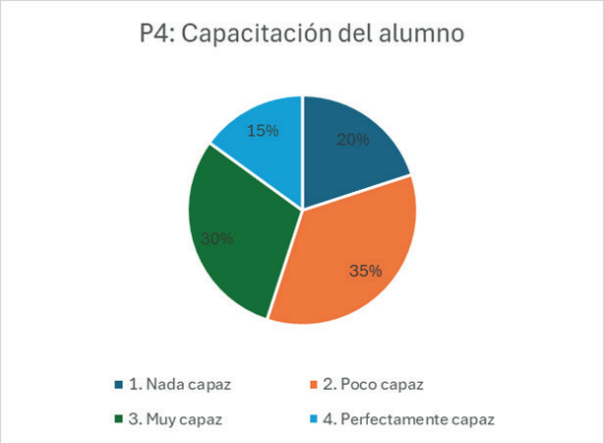
Resultados a la pregunta 3 de la encuesta



Elaboración propia.

Figura 7

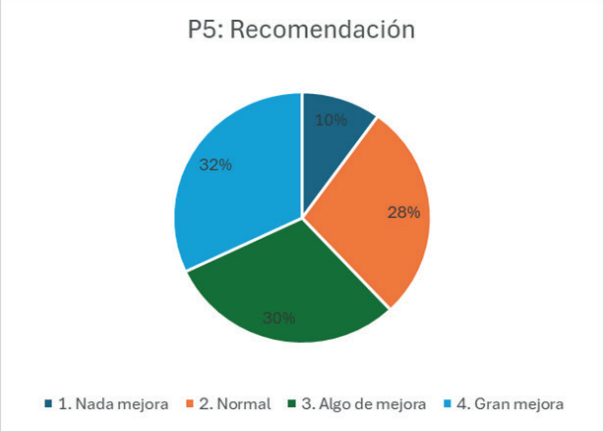
Resultados a la pregunta 4 de la encuesta



Elaboración propia.

Figura 8

Resultados a la pregunta 5 de la encuesta



Elaboración propia.

DISCUSIÓN DE RESULTADOS

Para concluir, los sistemas de recomendación pertenecen al tipo de sistemas de filtrado de información que busca mostrar o recomendar al usuario ítems en base a la información de este. Han sido comúnmente utilizados para recomendar películas, libros o mercancías; no obstante, debido a pandemia y a la gran cantidad de datos habientes en internet se están haciendo implementaciones en el campo académico como recomendar libros o artículos científicos. En este artículo, se está utilizando para la recomendación de cursos académicos para estudiantes universitarios para mejorar el proceso de orientación académica del alumno. El algoritmo de recomendación fue la similitud coseno que vectoriza los cursos y los compara entre sí para encontrar una cantidad de similitud y es usado ampliamente en los sistemas de recomendación como el filtrado colaborativo.

Para el proceso de validación, se realizó un cuestionario a 20 alumnos universitarios del último ciclo de la facultad de ingeniería de sistemas de la UNMSM ya que se considera a estos estudiantes más capacitados y con más conocimiento de la carrera y pueden declarar que cursos desean estudiar. La encuesta fue de 5 preguntas donde cada pregunta calificaba un aspecto clave del sistema (Usabilidad, Ahorro de tiempo, Calidad de la recomendación, etc.) y los resultados fueron optimistas se detectó que un 62% de los encuestados considera al sistema que le ayudo en su proceso de orientación y un 85% considera que había variedad en los cursos del sistema y que la recomendación le ayudó mucho. Sin embargo, consideramos que dicho porcentaje aún es bajo y se puede aumentar a través de la recolección de más datos y el ingreso de más atributos para un mejor análisis.

REFERENCIAS

[1] https://gestion.pe/economia/management-empleo/empleo-en-peru-el-75-de-empresas-en-peru-no-consiguen-trabajadores-con-competencias-digitales-mtpe-ministerio-de-trabajo-noticia/

[2] https://www.iberdrola.com/innovacion/data-mining-definicion-ejemplos-y-aplicaciones

[3] Dhawal Shah (2020c). Online degrees slowdown: A review of MOOC stats and trends in 2020. https://www.classcentral.com/report/mooc-stats-2020. [Online; accessed 7-December-2020]. https://gestion.pe/economia/empresas/la-formula-de-coursera-

para-llegar-a-17-millones-de-peruanos-al-cierre-de-2025-educacion-online-empresa-capacitacion-educacion-para-el-trabajo-peru-cursos-online-noticia/?ref=gesr

[4] (Khalid, A., Lundqvist, K., & Yates, A., A literature review of implemented recommendation techniques used in Massive Open online Courses, 2022)

[5] Yadav, S., Vikesh, Shreyam, & Nagpal, S. (2018). An Improved Collaborative Filtering Based Recommender System using Bat Algorithm. *Procedia Computer Science*, *132*, 1795–1803. https://doi.org/10.1016/j.procs.2018.05.155

[6] Urdaneta-Ponte, M. C., Mendez-Zorrilla, A., & Oleagordia-Ruiz, I. (2021). Recommendation Systems for Education: Systematic Review. *Electronics*, *10*(14), 1611. https://doi.org/10.3390/electronics10141611

[7] Mao, C., Chen, J., Yi, X., & Wen, L. (2024). An empirical study on metamorphic testing for recommender systems. Information And Software Technology, 107410. https://doi.org/10.1016/j.infsof.2024.107410

[8] Lops, P., Polignano, M., Musto, C., Silletti, A., & Semeraro, G. (2023). ClayRS: An end-to-end framework for reproducible knowledge-aware recommender systems. Information Systems, 119, 102273. https://doi.org/10.1016/j.is.2023.102273

[9] Pinedo, I., Larrañaga, M., & Arruarte, A. (2024). ArZiGo: A recommendation system for scientific articles. Information Systems (Oxford), 102367. https://doi.org/10.1016/j.is.2024.102367

[10] Zhang, H., Shen, X., Yi, B., Wang, W., & Feng, Y. (2023). KGAN: Knowledge Grouping Aggregation Network for course recommendation in MOOCs. Expert Systems With Applications, 211, 118344. https://doi.org/10.1016/j.eswa.2022.118344

[11] M, V. M., Dabade, M. S., P, S., Borhade, R. R., Barekar, S. S., Navarro, C., Concha, U. R., & Rodríguez, C. R. (2023). Smart education system to improve the learning system with CBR based recommendation system using IoT. Heliyon, 9(7), e17863. https://doi.org/10.1016/j.heliyon.2023.e17863

[12] Esteban, A., Zafra, A., & Romero, C. (2020). Helping university students to choose elective courses by using a hybrid multi-criteria recommendation system with genetic optimization. Knowledge-based Systems, 194, 105385. https://doi.org/10.1016/j.knosys.2019.105385

[13] Verma, M., & Patnaik, P. K. (2024). An automatic college library book recommendation system using optimized Hidden Markov based weighted fuzzy ranking model. Engineering Applications Of Artificial Intelligence, 130, 107664. https://doi.org/10.1016/j.engappai.2023.107664

[14] Zhang, H., Shen, X., Yi, B., Wang, W., & Feng, Y. (2023). KGAN: Knowledge Grouping Aggregation Network for course recommendation in MOOCs. Expert Systems With Applications, 211, 118344. https://doi.org/10.1016/j.eswa.2022.118344

[15] Yun, W. (2023). Online course resource recommendation based on deep learning. Procedia Computer Science, 228, 638-646. https://doi.org/10.1016/j.procs.2023.11.074

Financiamiento:

Propia.

Conflictos de interés:

Los autores declaran no tener conflictos de interés.

Contribuciones de autoría:

Ambos autores participaron en el desarrollo del artículo.