Actas de las Jenui, vol. 8. 2023.
Páginas: 351-358

Un recomendador para ayudar en la evaluación de la participación en foros de aulas virtuales

Javier Luis Cánovas Izquierdo
IN3 – UOC
jcanovasi@uoc.edu

José Antonio Hernández López
Universidad de Murcia
joseantonio.hernandez6@um.es

Jesús Sánchez Cuadrado
Universidad de Murcia
jesusc@um.es

David Bañeres Besora
Universitat Oberta de Catalunya
dbaneres@uoc.edu

Resumen

Los canales de comunicación virtuales como los foros de debate se ofrecen en numerosas asignaturas de titulaciones universitarias. La participación del estudiantado en estos espacios permite evaluar competencias como su capacidad comunicativa escrita, de argumentación o de trabajo en equipo, entre otras. Así, en muchas asignaturas se proponen actividades donde el alumnado ha de trabajar en grupo, argumentando y llegando a acuerdos para elaborar la solución. Esta argumentación puede ser representada como grafos de colaboración que permiten analizar y visualizar el entorno discursivo del foro. En este trabajo presentamos una solución basada en técnicas de aprendizaje automático que recomienda una calificación de la participación de cada estudiante en foros de debate virtuales a partir de grafos de colaboración, ayudando al profesorado en el proceso de evaluación. Analizamos dos arquitecturas de aprendizaje automático: modelos que usan métricas del grafo y redes neuronales de grafos; siendo la segunda la que ofrece mejores resultados en términos de error.

Abstract

Virtual communication channels such as discussion forums are offered in many subjects of bachelor's degrees. The students' participation in these spaces allows the assessment of skills such as written communication, argumentation ability or collaborative work, among others. Thus, many subjects propose activities where students have to work collaboratively, discuss and reach agreements to develop a solution. This discussion can be represented as collaboration graphs that allow visualising the discursive environment of the forum. In this work, we present a solution based on Machine Learning techniques that recommends an assessment mark of the students' participation in virtual discussion forums based on collaboration graphs, thus helping the teaching staff in the evaluation process. We analyze two machine learning architectures: models using graph metrics and graph neural networks, being the latter the one that offers the lowest error rate.

Palabras clave

Debate, docencia virtual, evaluación, aprendizaje colaborativo.

1. Motivación

El uso de medios electrónicos de comunicación se ha incorporado de manera gradual en asignaturas de titulaciones universitarias. Estas herramientas son especialmente relevantes en modelos docentes virtuales, donde el medio fundamental de aprendizaje es el aula virtual. En las aulas virtuales, el estudiantado, además de tener los recursos docentes y los calendarios de actividades, tienen herramientas de comunicación como chats, tablones de notificaciones o foros de debate.

Los chats tienden a utilizarse para comunicaciones breves entre un subconjunto del estudiantado y el tablón permite al profesorado informar y enviar indicaciones al alumnado. El canal que nos interesa en este artículo es el foro de debate virtual. Éste ofrece un canal bidireccional de comunicación entre el estudiantado y el profesorado, que pueden crear hilos de debate sobre un tema, y responder a estos hilos. Los foros de debate son comúnmente utilizados para promover la participación del estudiantado en un tema concreto, como dudas sobre el temario, o deliberación sobre alguna actividad de la asignatura. Precisamente por su capacidad de ofrecer un medio para expresarse y discutir sobre un tema, los foros de debate también se utilizan en actividades evaluables.

La evaluación de la participación en foros de debate es un proceso complejo que requiere de mecanismos especializados para ayudar en la calificación de los estudiantes. Entre las soluciones existentes, se encuentran herramientas que permiten la visualización de los foros de debate como grafos de colaboración [3]. Los grafos de colaboración permiten visualizar fácilmente el entorno discursivo del foro, ayudando al personal docente en el proceso de evaluación y, en última instancia, en la calificación. El uso de los grafos de colaboración para la evaluación no se limita a su visualización, sino que también se puede recurrir a las propiedades típicas de los grafos para guiar la evaluación.

En este artículo presentamos una propuesta para ayudar en la evaluación de la participación en foros de debates virtuales mediante un recomendador. Este recomendador toma como entrada un grafo de colaboración y propone una calificación para cada uno de los estudiantes. Para ello hace uso de mecanismos de aprendizaje automático (también conocido como *Machine Learning*, ML), de manera que aprende a predecir la calificación de los estudiantes en función de las características de los grafos de colaboración. Para entrenar el recomendador se utilizan los datos históricos de debates evaluados de semestres anteriores, es decir, los grafos de colaboración de cada foro de debate y la calificación de los estudiantes que participaron en dichos debates. Se analizan dos arquitecturas de aprendizaje automático, una basada en modelos que usan métricas sencillas del grafo y otra basada en redes neuronales de grafos. En los resultados experimentales se observa que las redes neuronales de grafos obtienen mejores resultados y son capaces de predecir la calificación con poco error. Por supuesto, esta propuesta de calificación es una recomendación que ha de ser validada por el profesorado, pero ofrece una ayuda basada en datos históricos, permitiendo un proceso de evaluación consistente.

El resto del artículo está organizado de la siguiente manera. La sección 2 describe el trabajo relacionado. Las secciones 3 y 4 presentan la propuesta y su validación mediante un experimento, respectivamente. Finalmente, la sección 5 incluye una discusión de la propuesta y la sección 6 presenta las conclusiones y el trabajo futuro pendiente.

2. Trabajo relacionado

La proliferación de la educación en línea ha favorecido la creación de nuevas herramientas para apoyar el proceso de enseñanza-evaluación-aprendizaje. Algunas se centran en mejorar el proceso de distribución de contenidos, otras en la mejora del acompañamiento o en el análisis y mejora de los procesos de retorno personalizado, pero también ofrecen una oportunidad para ayudar al profesorado en los procesos de evaluación.

Este tipo de educación realizada mediante sistemas de gestión del aprendizaje (también conocido por sus siglas en inglés para *Learning Management Systems*, LMS), generan una gran cantidad de datos, almacenando datos de navegación, comunicación y académicos del estudiantado. Todo esto ha permitido desarrollar diferentes herramientas analíticas de aprendizaje (o *Learning Analytics*, LA) y de minería de datos (o *Educational Data Mining*, EDM).

Estas segundas ofrecen una gran potencialidad ya que se concentran en el descubrimiento automatizado de información a partir de los datos. Por ejemplo, a partir de datos académicos se puede identificar estudiantes en riesgo y diseñar intervenciones para motivarlos y reducir las tasas de suspensos [8, 6]. Además, estos datos académicos combinados con datos de navegación y comunicación pueden identificar estudiantes en riesgo de abandonar antes que suceda [9, 11].

Además de predecir niveles de riesgo, los datos de comunicación se han utilizado en otras propuestas como la detección del estado de ánimo del estudiantado [4] o bien en la respuesta automática a consultas del estudiantado mediante bots [2]. Aunque estos sistemas ayudan al profesorado en el día a día dentro del aula, existen pocos trabajos donde se utilicen como apoyo en la evaluación de la calidad de la discusiones o actividad de los alumnos.

La tarea de evaluar la actividad en un foro de debate no es fácil para el profesorado ya que se debe tener en cuenta diferentes factores como el número de intervenciones o el contenido de las aportaciones. Por eso, se han utilizado diferentes técnicas dependiendo del objetivo de la evaluación. Por una parte, en el caso de la calidad del contenido, la minería de textos (o *text mining*) es la técnica más utilizada. Analizar el contenido del texto [5] o las palabras clave utilizadas [1] puede facilitar al profesorado la evaluación de las intervenciones. Por otra parte, en el caso de evaluar la cantidad de aportaciones, se han utilizado métricas como el número de mensajes o la interrelación con los otros estudiantes mediante análisis con redes sociales [10]. No obstante, en todos los casos, estas propuestas son a nivel de visualización y no a nivel de un sistema de recomendación.

3. Propuesta

Este trabajo presupone un escenario típico en la evaluación de discusiones en foros de debate virtual (véase figura 1). Como puede observarse, un LMS, como por ejemplo CANVAS, SAKAI o MOODLE, ofrece un conjunto de herramientas para la realización de tareas en entornos virtuales, siendo los foros virtuales una de ellas. En estos foros, el estudiantado participa y pue-

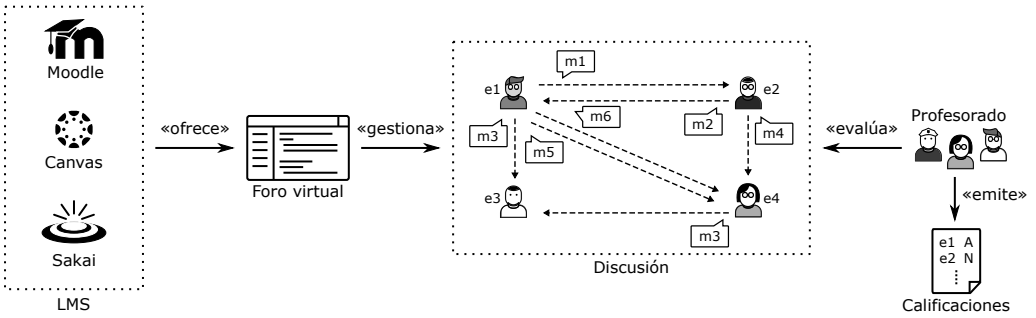


Figura 1: Escenario típico de evaluación de discusiones en foros de debate virtual.

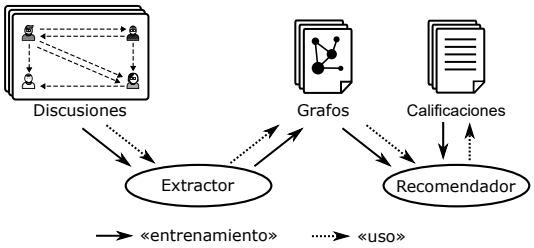


Figura 2: Recomendador para la evaluación a partir de grafos de colaboración.

de discutir diversos temas. Cuando la actividad que se realiza en el foro forma parte de una tarea evaluable, el equipo de profesores debe evaluar la calidad de la discusión y la contribución de cada estudiante. Finalmente, el profesorado emite una calificación para cada estudiante. El proceso de evaluación y calificación es una tarea compleja que requiere comprobar múltiples aspectos de la discusión (por ejemplo, número de mensajes enviados por estudiante, temas tratados o subgrupos de discusión creados dentro del foro, etc.).

Para ofrecer una ayuda en el proceso de evaluación, se propone un recomendador para asistir al profesorado durante la calificación de los grupos de estudiantes que participan en un foro de debate virtual. La figura 2 muestra, de manera esquemática, el diseño del recomendador. El recomendador trabaja con grafos de colaboración, que representan la discusión del estudiantado en el foro, y se entrena con los datos históricos de semestres anteriores, que incluyen la calificación del estudiantado y los grafos de colaboración. Como puede observarse, los grafos de colaboración se extraen, mediante una herramienta (ver *Extractor*), de las discusiones surgidas en los foros virtuales de la asignatura. Durante el entrenamiento (ver flechas negras), los conjuntos de grafos y las calificaciones de ediciones anteriores de la asignatura se utilizan para entrenar el recomendador (ver *Recomendador*). Una vez entrenado, se podrá utilizar para recomendar calificaciones en nuevos grafos de colaboración y así asistir al profesorado (ver flechas punteadas).

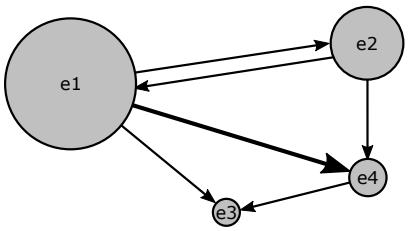
A continuación, describimos los grafos de colaboración y su extracción, así como el proceso de entrenamiento del recomendador.

3.1. Grafos de colaboración

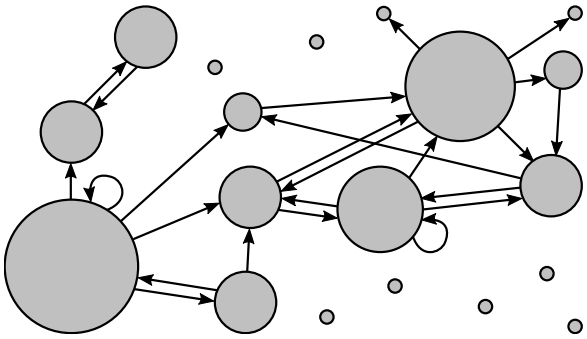
Los grafos de colaboración son una manera de representar una discusión entre el estudiantado en un foro de debate virtual. Un foro de debate virtual está compuesto por los actores de la discusión, que en nuestro caso es el estudiantado; y las interacciones entre los actores, que son sus mensajes/contribuciones.

Un grafo de colaboración es un grafo dirigido que está formado por (1) vértices, que representan a los actores de la colaboración y (2) aristas, que representan las interacciones entre los actores. Existe una arista entre dos estudiantes si uno de ellos ha enviado un mensaje a otro (ya sea iniciando una discusión o respondiendo a un mensaje anterior), y la arista se dirige del estudiante que crea el mensaje hacia el estudiante que recibe el mensaje. Dado que un estudiante puede corregir o clarificar una aportación propia, es posible que el grafo contenga aristas con mismo origen y destino. Nótese que, al ser un foro, los mensajes son públicos y, por tanto, cualquier estudiante puede responder a cualquier otro estudiante, dando lugar a nuevas aristas. Los vértices del grafo tienen las siguientes propiedades: (1) etiqueta, que corresponde al email y/o nombre del estudiante y (2) número total de mensajes enviados por el estudiante. El número total de mensajes se usa para establecer el tamaño del nodo en el momento de visualizar el grafo. De esta forma, el tamaño del nodo es proporcional al número de mensajes enviados. Por otro lado, las aristas del grafo de colaboración tienen un peso, que indica el número de mensajes entre los dos estudiantes involucrados. De forma parecida al tamaño de los nodos, el peso de las aristas se utiliza para establecer su grosor, que es directamente proporcional al peso, en el momento de visualizar el grafo.

La figura 3a muestra un ejemplo de grafo de colaboración basado en la discusión mostrada en la figura 1, mientras que la figura 3b muestra un grafo de colabo-



(a) Grafo para la discusión mostrada en figura 1



(b) Ejemplo de grafo real extraído de una de las discusiones utilizadas para entrenar el recomendador.

Figura 3: Ejemplos de grafo de colaboración.

ración generado a partir de un debate real (los nodos se muestran sin etiquetas para proteger la privacidad del estudiantado). Como puede observarse, la visualización del grafo de colaboración ya es por sí misma una ayuda para comprender la colaboración. Por ejemplo, es fácil reconocer a los estudiantes que más han participado y las principales interacciones (o grupos de estudiantes discutiendo) del debate.

La extracción de los grafos de colaboración a partir de las discusiones de un foro no es una tarea trivial. Además del conocimiento técnico sobre el LMS donde está alojado el foro, el proceso requiere navegar por el conjunto de mensajes, extraer las relaciones de dependencia (esto es, identificar si un mensaje responde a otro) e identificar su autor. Para llevar este proceso utilizamos la herramienta FLYZER[1] [3], que es una extensión para el navegador Web Google Chrome capaz de analizar la web de un foro virtual y extraer el correspondiente grafo.

3.2. Entrenamiento del recomendador

Esta sección describe el problema de recomendar calificaciones y la solución propuesta. En primer lugar se presenta una formalización del problema ya que una cuestión importante es cómo codificar el grafo de colaboración para que pueda ser procesado por los algoritmos de aprendizaje. A continuación, se describen las soluciones tecnológicas desarrolladas. En particular, consideramos la aplicación de dos tipos de algoritmos de aprendizaje: modelos basados en métricas sencillas del grafo y redes neuronales de grafos.

Formalización del problema. Dado un grafo de colaboración, el objetivo del recomendador es asignar una calificación a cada uno de los estudiantes. Más formalmente, un grafo de colaboración es un multigrafo $G = (V, E, f)$, donde V representa el conjunto de vértices, E las aristas y $f : E \longrightarrow V \times V$ indica el nodo origen y destino para cada arista. Asumimos que cada nodo del grafo tiene asociado una calificación, es decir, $\mu : V \longrightarrow M$ donde $M = \{\text{SB, NO, A, SU}\}$ (sobresaliente, notable, aprobado y suspenso, respectivamente). Así pues, el problema se define de la siguiente manera: dado un vértice $v \in V$ de un grafo de colaboración, el sistema debe asignarle una calificación. De este modo, tenemos que resolver un problema de clasificación de nodos [13].

Modelos de aprendizaje automático y métricas de grafos. Como primera solución tecnológica hemos implementado varios modelos de aprendizaje basados en la codificación de los grafos como vectores. Los vectores se construyen extrayendo características individuales de los vértices. De esta manera, dado un vértice $v \in V$, se extraen varias métricas según su posición en el grafo de colaboración generando un vector de características. Dicho vector se utiliza como entrada de un modelo de ML entrenado para que asigne una calificación a cada vértice. Las métricas que se han extraído en este trabajo han sido las siguientes, las cuales han sido seleccionadas por su potencial para representar la colaboración:

- Grado de entrada. Número de mensajes que el estudiante recibe.
- Grado de salida. Número de mensajes que el estudiante escribe.
- Número de vértices distintos entrantes. Número de personas de las que recibe al menos un mensaje.
- Número de vértices distintos salientes. Número de personas a las que escribe al menos un mensaje.
- Coeficiente de *clustering* local. Cuantifica lo cerca que están sus vecinos de ser un grafo completo, es decir, cuantifica si sus vecinos se envían mensajes entre ellos. En grafos dirigidos, el coeficiente de clustering local de un nodo n se calcula dividiendo el número de aristas distintas que conectan los vecinos de n entre el número máximo de aristas que podría haber entre los vecinos de n. La figura 4 muestra tres ejemplos de clustering local.

[1] https://github.com/jlcanovas/flyzer

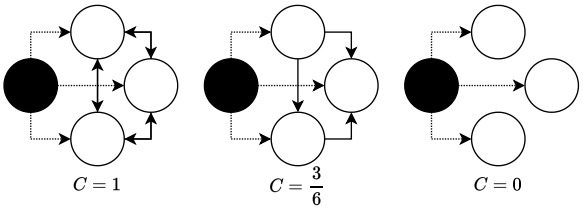


Figura 4: Coeficientes de clustering local de un nodo (representado en negro) en tres grafos diferentes.

Así pues, para cada vértice se extraen estas cinco métricas generando un vector de características de cinco dimensiones. Dicho vector se introduce como entrada a un modelo de ML. En este trabajo, hemos considerado los siguientes modelos de ML: (1) modelos bayesianos, (2) máquinas de soporte vectorial (o *Support Vector Machine*, SVM) y (3) regresión logística. La decición se seleccionar estos modelos se debe a que son los modelos de ML que tradicionalmente se aplican.

Redes neuronales de grafos. Además de los modelos de ML tradicionales comentados en la sección anterior, hemos considerado redes neuronales de grafos (más conocidas en inglés como *Graph Neural Networks*, GNN) [12], que reciben como entrada un grafo y por tanto pueden aprender de la estructura del mismo. En particular, una GNN es un tipo de red neuronal que recibe como entrada un grafo y genera representaciones vectoriales de sus vértices usando información de sus vértices vecinos (esto es, explotando la estructura del grafo). Más concretamente, dado un nodo $v \in V$ y una representación vectorial inicial x_v, una GNN de L capas calcula la representación final del nodo de esta manera:

$$h_v^0 = x_v \tag{1}$$

$$h_v^l = g_l\left(h_v^{l-1}, \left\{h_w^{l-1} : w \in \mathcal{N}(v)\right\}\right) \in \mathrm{R}^d \tag{2}$$

para cada $l = 1, \ldots, L$, $N(v) = \{w \in V | \exists e$ tal que $f(e) = (w, v)\}$ (vecindario de v) y g_l es una función no lineal que calcula h_v^l usando los vectores h_v^{l-1} y $\left\{h_w^{l-1} : w \in \mathcal{N}(v)\right\}$. Por consiguiente, el vector final asociado a un vértice tendrá información de todos los vecinos que estén a L pasos de él. Finalmente, el vector resultante se introduce como entrada a una capa lineal seguida de la activación *Softmax* para generar la recomendación.

$$o_v = \text{Softmax}(W_f h_v^L),$$

donde W_f es la capa final lineal representada por una matriz de dimensiones $4 \times d$ y o_v es un vector de dimensión 4 (número de notas asignables). La componente i-ésima de este último vector indica la probabilidad de asignarle al nodo v la nota i. A la hora de recomendar, al estudiante v se le asigna la nota cuya probabilidad asociada sea mayor.

En nuestro problema, usamos dos capas que siguen el operador convolucional usado en [7] intercaladas con activaciones ReLU (esto define g_l de la Ecuación 2), inicializamos a unos los vectores iniciales (x_v en la Ecuación 1) y fijamos la dimensión interna a 32, es decir $d = 32$.

4. Experimento y validación

4.1. Contexto

Para validar nuestra propuesta, hemos realizado un experimento con datos reales de un foro de debate virtual. En particular, el experimento se ha realizado en una asignatura de Ingeniería de Requisitos del Grado de Informática de la Universitat Oberta de Catalunya. Se trata de una asignatura optativa que se imparte durante el primer semestre de cada curso académico y que el estudiantado puede matricularse a partir del tercer año. La asignatura presupone conocimientos básicos en ingeniería del software, ya que el estudiantado debe haber cursado la asignatura de Ingeniería del Software previamente, y profundiza en la etapa del ciclo de vida de desarrollo de software que se dedica a la obtención, gestión, documentación, validación y verificación de requisitos.

El modelo de evaluación de la asignatura se basa en la evaluación continua y está compuesta por cuatro pruebas de evaluación de carácter obligatorio durante el semestre. Las pruebas están diseñadas para permitir al estudiantado tener una visión completa de todas las fases de la ingeniería de requisitos. De esta manera, todas las pruebas se articulan alrededor de un mismo caso de estudio y cada una de ellas trabaja una tarea específica de la ingeniería de requisitos. En concreto, la primera prueba se centra en la obtención de requisitos; la segunda trabaja la gestión de requisitos; la tercera trabaja la documentación de requisitos; y la cuarta repasa la validación y verificación de requisitos. Mientras que el caso de estudio es diferente cada semestre, la estructura y elementos de evaluación de las pruebas son similares, dotando de originalidad a la solución pero facilitando la evaluación.

Este experimento se desarrolla en la segunda prueba, centrada en gestión de requisitos, donde los estudiantes deben trabajar colaborativamente en el foro de debate del aula virtual. Durante la prueba, los estudiantes han de discutir y trabajar en tres tareas: (1) acordar una visión de producto, (2) proponer y priorizar requisitos, y (3) seleccionar y estimar requisitos. La entrega la realiza uno de los estudiantes del grupo, que ha de entregar la propuesta de solución a las tareas. Para facilitar la participación, el tamaño de cada grupo es de entre 6 y

10 alumnos, y la formación de los grupos se realiza por iniciativa propia de los estudiantes.

4.2. Descripción del experimento

A continuación, describimos los detalles del experimento explicando el tratamiento de los datos, los modelos usados y la métrica de evaluación.

Datos. Para la recolección de datos, consideramos los tres últimos semestres de la asignatura de Ingeniería de Requisitos. En ese período de tiempo participaron 503 estudiantes en el desarrollo de la segunda prueba de evaluación, creándose 75 grupos de discusión en total. De esta forma, el proceso de extracción de grafos descrito en la sección 3.1 fue aplicado al finalizar dichas discusiones, extrayéndose un total de 75 grafos.

También se obtuvo la calificación de cada uno de los estudiantes en esta actividad, para poder entrenar al recomendador, añadiendo dicha calificación a los respectivos nodos de los grafos de colaboración. Las calificaciones asignadas están en el rango: suspenso (SU), aprobado (A), notable (NO) o sobresaliente (SB). Al ser una prueba de evaluación intermedia y no la nota final de la asignatura, este rango no incluye la calificación de matrícula de honor.

Sin pérdida de generalidad, se puede asumir que tenemos un único grafo gigante que incluye los 75 grafos como componentes conexas. Del grafo gigante, el conjunto de vértices se ha dividido en entrenamiento/validación/prueba siguiendo unas proporciones 0.7/0.1/0.2. Los vértices del conjunto de entrenamiento se utilizan para entrenar los modelos, los vértices del conjunto de validación para seleccionar las mejores configuraciones de los modelos y, finalmente, los vértices del conjunto de prueba se utilizan para dar una estimación de cuán bien generalizan los modelos entrenados y comparar las aproximaciones.

Modelos. Se han considerado varios modelos: (1) aquellos que reciben como entrada los vectores de características, que son modelos bayesianos, SVM y regresión logística (ver segundo sub-apartado de la sección 3.2); (2) una red neuronal de grafos (ver tercer sub-apartado de la sección 3.2); y (3) un modelo *baseline* para comparar el resto de modelos. El modelo *baseline* es un modelo ingenuo que asigna una calificación a cada estudiante de manera aleatoria y siguiendo las proporciones del conjunto de entrenamiento. Es decir, la probabilidad de cada calificación viene dada por la proporción de dicha calificación en el conjunto de entrenamiento.

Métrica de evaluación. Como métrica de evaluación de la efectividad de los modelos en la recomendación de calificaciones, hemos utilizado *Macro Average Mean Absolute Error* (MAMAE). Esta métrica se define como la media del error medio absoluto de cada clase y está pensada para conjuntos de datos no balanceados ya que da la misma importancia a cada clase. El gráfico mostrado en la figura 5 muestra que las clases están desbalanceadas lo que justifica el uso de dicha métrica. Para calcular esta métrica, a cada calificación se le asigna a un entero. En nuestro caso a SB se le asigna el valor 3, a NO el valor 2, a A el valor 1 y a SU el valor 0.

	NOTA REAL	PREDICCIÓN	ERROR ABSOLUTO
Alumno 1	SB	NO	$\|3-2\| = 1$
Alumno 2	NO	SU	$\|2-0\| = 2$
Alumno 3	A	A	$\|1-1\| = 0$
Alumno 4	SU	A	$\|0-1\| = 1$
Alumno 5	SB	A	$\|3-1\| = 2$

Cuadro 1: Ejemplo de cálculo del error medio.

NOTA	ALUMNOS	MAE
SB	1, 5	MAE-SB = $\frac{1+2}{2} = 1{,}5$
NO	2	MAE-NO = 2
A	3	MAE-A = 0
SU	4	MAE-SU = 1
		MAMAE = $\frac{1{,}5+2+0+1}{4} = 1{,}125$

Cuadro 2: Ejemplo de cálculo del MAMAE para los alumnos mostrados en el cuadro 1.

A modo de ilustración consideremos que tenemos cinco alumnos. Sus calificaciones reales y la predicción se muestran en el cuadro 1, junto con el error absoluto de cada ejemplo. Como podemos observar, el error absoluto cuenta el número de "saltos" entre calificaciones. Por ejemplo, en el caso del Alumno 2, el error absoluto es 2 porque hay una variación de dos notas entre la nota real y la predicción. El cuadro 2 muestra, para cada calificación, su error medio absoluto asociado (MAE-*Nota*). Dichos errores medios son los que se emplean para calcular el MAMAE. Un MAMAE de 0 indica una predicción perfecta por parte del modelo. Así pues, a la hora de comparar modelos, se considerará mejor al modelo que tenga menos MAMAE en el conjunto de prueba.

4.3. Resultados

Una vez entrenados los modelos, evaluamos los mismos usando MAMAE sobre el conjunto de prueba. El cuadro 3 muestra los resultados para cada uno de los modelos, detallando el error medio de cada clase. Podemos extraer las siguientes observaciones.

Identificación de suspensos. Los modelos SVM y regresión logística no identifican bien a los suspensos

MODELO	MAE-SU	MAE-A	MAE-NO	MAE-SB	MAMAE
Baseline	1.5000	1.3750	0.5882	0.7727	1.0590
GNN	0.0000	0.7500	0.5294	0.9091	**0.5471**
Modelo bayesiano	0.0000	0.6250	0.9118	1.0455	0.6456
SVM	1.5000	0.4375	0.7941	0.7955	0.8818
Regresión logística	1.0000	0.8750	0.8235	0.8409	0.8849

Cuadro 3: Error medio absoluto de cada clase (MAE-*nota*) y la media de los mismos (MAMAE).

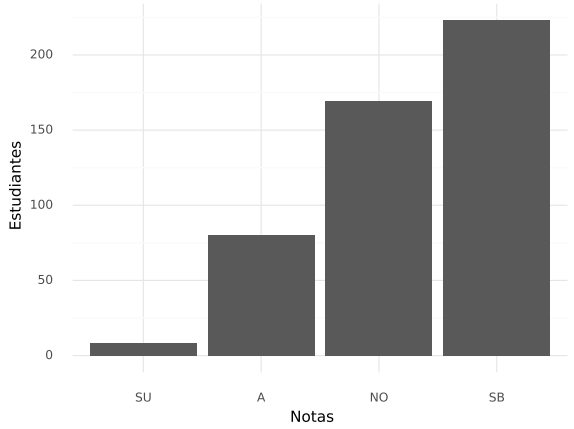


Figura 5: Distribución de calificaciones.

lo cual penaliza su MAMAE. Esto se debe a que estos modelos no son capaces de superponerse al desbalanceo presente en el conjunto de datos. Por esta misma razón se observa que estos modelos tienen menor MAE-SB (SB es la clase mayoritaria, véase figura 5) en comparación con GNN y modelos bayesianos. Por otro lado, las redes de grafos y los modelos bayesianos son capaces de identificar suspensos de manera perfecta.

Todos los modelos superan al clasificador ingenuo. Esto indica que es posible predecir la calificación de un estudiante únicamente usando la estructura del grafo de colaboración. Así pues, una potencial vía futura es considerar más características del grafo de colaboración como, por ejemplo, el contenido de los mensajes que se envían entre los alumnos (usando procesamiento del lenguaje natural).

El modelo que obtiene mejores resultados es la red neuronal de grafos. Los resultados de nuestro experimento tienen un MAMAE de $0{,}5471$. El resultado obtenido se puede interpretar como que el modelo típicamente comete un error inferior al salto entre dos calificaciones. En este sentido, la red neuronal de grafos entrenada demuestra ser capaz de capturar bien patrones de intercambio de mensajes dentro de los grafos de colaboración.

5. Discusión

En esta sección discutimos el potencial de aplicación de nuestra propuesta, el impacto que podría tener en la metodología docente, y sus limitaciones.

Optimización del tiempo de corrección de actividades. La tarea de revisar y evaluar las aportaciones de cada estudiante en una discusión es compleja y está directamente ligada al número de mensajes y estudiantes involucrados. Además, si el profesorado ha de evaluar diferentes grupos de estudiantes, el esfuerzo es muy superior. Con la ayuda de nuestra propuesta, la tarea de evaluar se ve facilitada y se puede conseguir una optimización en los tiempos de corrección. En particular, en su estado actual nuestro modelo puede resultar bastante útil para realizar un primer filtrado en la tarea de calificación, ya que demuestra ser bastante preciso en la tarea de identificar suspensos (ver resultados de GNN en la tabla cuadro 3).

Consistencia en la evaluación. Como el recomendador se entrena según el histórico de evaluaciones, las propuestas de evaluación están alineadas con el comportamiento típico del alumnado en las discusiones. Es más, el recomendador puede ayudar a mantener la consistencia a la hora de calificar, ya que es un dato adicional que el profesor puede tener en cuenta a la hora de realizar esta tarea, que puede verse influida por aspectos subjetivos (p. ej., cansancio tras revisar una gran cantidad de trabajos). En este mismo contexto, el recomendador también puede resultar útil como guía cuando hay varios profesores involucrados en la corrección. De esta manera, es más fácil que éstos alineen sus calificaciones a un criterio común.

Limitaciones. Como todo modelo de predicción, nuestra propuesta tiene algunas limitaciones. Primero, el modelo produce una predicción y, por lo tanto, como predicción no podemos asegurar que la calificación asignada sea la correcta. El profesorado debe revisar las calificaciones antes de publicarlas al estudiantado. Segundo, el modelo es susceptible a los cambios docentes en la asignatura. Es decir, cualquier cambio en la tipología de actividad, duración de la actividad o el modelo de evaluación en general de la asignatura puede invalidar el modelo entrenado. Finalmente, también se ha de tener en cuenta que solo se ha utilizado la

estructura del grafo, mientras que el profesor también evalúa el contenido de los mensajes. De esta forma, es esperable que utilizando esta información se pudiera mejorar el modelo.

6. Conclusiones y trabajo futuro

En este artículo hemos presentado un sistema para ayudar en la evaluación de la participación en las discusiones surgidas en foros de debates virtuales mediante el uso de un recomendador. El recomendador toma como entrada el grafo de colaboración de una discusión y propone una calificación para cada uno de los participantes. Nuestra propuesta hace uso de mecanismos de aprendizaje automático para, una vez se ha entrenado con datos históricos de la evaluación de semestres anteriores, aprender a predecir la calificación del estudiantado. El recomendador ha sido validado en un experimento con lo datos de la asignatura de Ingeniería de Requisitos del Grado en Informática de la Universitat Oberta de Catalunya, demostrando que es capaz de predecir la evaluación de manera eficaz con un error medio inferior al salto entre dos calificaciones.

Como trabajo futuro, queremos continuar desarrollando nuestra propuesta para considerar características adicionales en el entrenamiento, como por ejemplo la calidad de las contribuciones del estudiantado en las discusiones. Además, este trabajo se ha presentado un experimento sin una aplicación práctica del mismo. Por lo tanto, nos interesa aplicar el modelo en un entorno de aprendizaje real valorando la opinión del profesorado sobre la utilidad del recomendador. Finalmente, nos gustaría extrapolar esta propuesta a otras asignaturas con características similares.

Referencias

[1] Yvette Awuor y Robert Oboko. Automatic assessment of online discussions using text mining. *International Journal of Machine Learning and Applications*, 1(1):7, 2012.

[2] Yassine Benjelloun Touimi, Abdeladim Hadioui, Noureddine El Faddouli y Samir Bennani. Intelligent chatbot-LDA recommender system. *International Journal of Emerging Technologies in Learning (iJET)*, 15(20):4–20, 2020.

[3] Javier Luis Cánovas Izquierdo, Robert Clarisó y David Bañeres. Una herramienta para la evaluación de debates en aulas virtuales. *Actas de las Jornadas sobre Enseñanza Universitaria de la Informática*, (4):385–388, 2019.

[4] Karim Elia Fraoua, Jean-Marc Leblanc y Amos David. Use of an Emotional Chatbot for the Analysis of a Discussion Forum for the Improvement of an E-Learning Platform. En *International Conference on Learning and Collaboration Technologies. Human and Technology Ecosystems*, pp. 25–35. Springer, 2020.

[5] Sergio García-Molina, Carlos Alario-Hoyos, Pedro Manuel Moreno-Marcos, Pedro J. Muñoz-Merino, Iria Estévez-Ayres y Carlos Delgado Kloos. An algorithm and a tool for the automatic grading of MOOC learners from their contributions in the discussion forum. *Applied Sciences*, 11(1):95, 2020.

[6] Ana Elena Guerrero Roldán, M. Elena Rodríguez, David Bañeres, Cristina Pérez Solà, Javier Panadero y Abdulkadir Karadeniz. Hacia un sistema de detección temprana de estudiantes en riesgo en entornos de enseñanza-aprendizaje en línea. *Actas de las Jornadas sobre Enseñanza Universitaria de la Informática*, (5):37–44, 2020.

[7] Thomas N. Kipf y Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[8] Leah P. Macfadyen y Shane Dawson. Mining LMS data to develop an "early warning system" for educators: A proof of concept. *Computers & education*, 54(2):588–599, 2010.

[9] Carlos Márquez-Vera, Alberto Cano, Cristobal Romero, Amin Yousef Mohammad Noaman, Habib Mousa Fardoun y Sebastian Ventura. Early dropout prediction using data mining: a case study with high school students. *Expert Systems*, 33(1):107–124, 2016.

[10] Bernardo Pereira Nunes, Matthew Tyler-Jones, Gilda H.B. de Campos, Sean W.M. Siqueira y Marco A. Casanova. Fat: A real-time (f)orum (a)ssessment (t)ool to assist tutors with discussion forums assessment. En *Symposium on Applied Computing*, volumen 15, 2015.

[11] M. Elena Rodríguez, David Bañeres y Ana Elena Guerrero-Roldán. Hacia un sistema de detección temprana del riesgo de abandono en entornos en línea. *Actas de las Jornadas sobre Enseñanza Universitaria de la Informática*, (7):207–2014, 2022.

[12] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner y Gabriele Monfardini. The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80, 2008.

[13] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang y S. Yu Philip. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24, 2020.