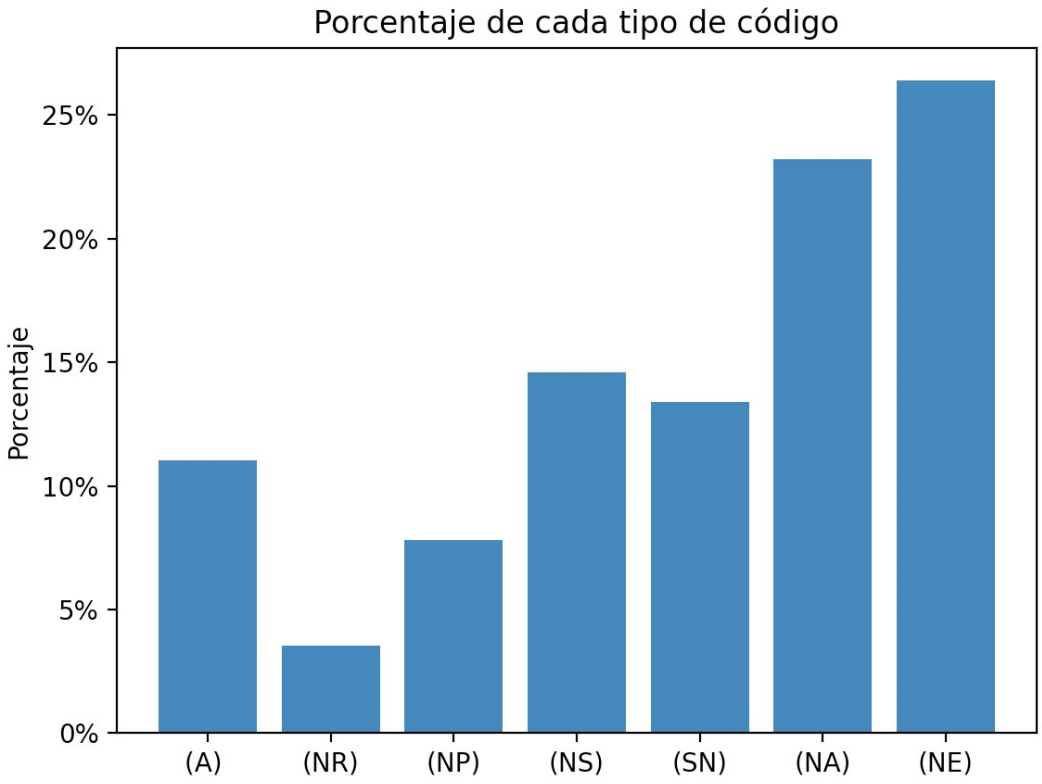
Introducción

Hemos desarrollado un modelo de red neuronal para predecir el motivo por el que un alumno obtiene evaluación negativa en una asignatura de Educación Secundaría.

En el IES Sierra Blanca de Marbella se aplica una metodología llamada Ventanas Panorámicas. Estas están constituidas por unos códigos que indicar el motivo de la evaluación negativa de un alumno en cada asignatura. El procedimiento habitual es que cada profesor reflexiones sobre la nota que ha entregado a cada alumno y dictamine este motivo a elegir entre siete posibilidades:

1: '(A) ABSENTISTA',
2: '(NR) NO SE RELACIONA',
3: '(NP) NO PUEDE',
4: '(NS) NO SE COMPORTA',
5: '(SN) SE NIEGA A TRABAJAR O ESTUDIAR EN EL AULA',
6: '(NA) NO ATIENDE',
7: '(NE) NO ESTUDIA Y NO TRABAJA EN CASA'

Veamos en las siguiente gráfica la frecuencia de cada código



El IES Sierra Blanca escolariza alumnos de un barrio humilde de Marbella. La ciudad es rica y sus trabajadores tienen ingresos por encima de la media nacional pero las familias no tienen grandes aspiraciones académicas para sus hijos. Los alumnos son bastante absentistas no legalmente sino en el sentido de que cada día faltan al centro entre el 10% y el 20% de forma habitual.

En cuanto al profesorado del centro es muy inestable. El claustro tiene ochenta profesores aproximadamente (cambia cada curso). Mas de dos tercios del profesorado cambia de destino cada

curso y los profesores nuevos tienen que aprender los protocolos del centro cada nuevo curso. Esto incide en una relativa falta de coherencia en los criterios que se aplican a cada procedimiento.

El entrenamiento de la red neuronal ha demostrado que si se entrena con los datos de un solo profesor puede predecir con un 99% de precisión pero si lo hacemos con el profesorado real del centro la precisión baja debido a la mencionada e inevitable falta de coherencia.

El modelo ha sido entrenado con los datos de 527 alumnos de ESO. Los datos de notas han sido desacoplados de los nombres y apellidos para cumplir la Ley de Protección de Datos y para evitar cualquier sesgo que pudieran inducir los apellidos de los alumnos.

Este proyecto es útil especialmente para ayudar a los profesores nuevos de cada curso a indicar el motivo que deben reflejar en las ventanas panorámicas.

Datos de entrenamiento

Para entrenar la red neuronal hemos usado los datos de 527 alumnos de ESO. Estos alumnos cursan 4 niveles diferentes:

137 alumnos de 1º ESO

148 alumnos de 2º ESO

127 alumnos de 3º ESO

125 alumnos de 4º ESO

Todos estos alumnos no cursan el mismo número de asignaturas porque en unos niveles hay 10 asignaturas y en otros 11. Esto nos permite crear datos artificiales combinando las 11 asignaturas para obtener 11 combinaciones diferentes de 10 asignaturas para cada alumno. Recordemos que solo las asignaturas con evaluación negativa tienen asociado un código de Ventana Panorámica. El algoritmos se alimenta cada vez con una asignatura suspensa, 9 asignaturas mas del mismo alumno y todas las faltas de asistencia del alumno. Con los datos que tenemos podemos alimentar el modelo 5802 veces.

Cada asignatura suspensa de un mismo alumno se alimenta al modelo por separado y obtiene un código diferente.

Tenemos mas asignaturas suspensas que 5802 pero no siempre los profesores han puesto el código de ventana panorámicas por errores o falta de información

El perfil de absentismo del alumno se define como la secuencia de faltas que ha tenido el alumno a lo largo del trimestre sin tener en cuenta si la falta ha sido o no justificada por entender que incide de la misma forma en el rendimiento académico. En total, los alumnos han acumulado 28695 horas de falta. Con las faltas creamos un array con valores 0 y 1 para cada alumno. Si el alumno no ha faltado nunca tendrá una secuencia de ceros de longitud igual al producto de 63 días lectivos por 6 horas que tiene cada día, 378 ceros. Si un alumno faltara siempre tendría una secuencia de 378 unos y los alumnos reales tienen una combinación de 378 ceros y unos.

Metodología

El código desarrollado implementa un modelo de red neuronal utilizando PyTorch para clasificación. Aquí está la descripción metodológica:

- Modelo de Aprendizaje Automático: Se incluyen 378 datos binarios de faltas y 10 datos de notas en los que la primera notas es la que corresponde a la asignatura cuyo código

buscamos. Se utiliza el modelo de perceptrón multicapa que incorpora una capa de convolución 1D (Conv1d) para las faltas, una capa lineal para las notas y una capa oculta.

- Entrenamiento: El modelo se entrena utilizando el optimizador SGD (Gradiente Descendente Estocástico) con una tasa de aprendizaje de 0.01. Se utiliza la función de pérdida de entropía cruzada (CrossEntropyLoss) para calcular la pérdida durante el entrenamiento.
- Dataloader: Se utiliza el DataLoader de PyTorch para cargar los datos. Se dividen los datos en conjuntos de entrenamiento y evaluación en una proporción de 80-20.
- Evaluación: Se evalúa el modelo en el conjunto de datos de evaluación. Se calcula el porcentaje de aciertos para cada clase y se muestra junto con el número total de intentos para esa clase.

Experimentación

Hemos probado otras bibliotecas de funciones para IA: TensorFlow y Scikit-learn. Hemos elegido PyTorch por parecer mas simple y fácil para comprender para docentes y alumnos de secundaria a los que va dirigido el proyecto. Por otro lado el hardware disponible una GPU RTX 2060 con 6Gb de memoria dedicada ha funcionado de forma optima con esta biblioteca gracias a la plataforma de computación paralela CUDA de Nvidia que funciona especialmente bien con PyTorch.

También hemos experimentado con los datos: hemos intentado usar datos de retraso en la llegada de los alumnos a clase. Con estos datos no hemos mejorado los resultados debido principalmente a que muchos profesores no reflejan estas incidencias en el registro. En este experimento los tiempos de entrenamiento de duplicaron por los que abandonamos esta vía.

Por otro lado hemos probado a no tener en cuenta las faltas para acelerar el entrenamiento del modelo pero han empeorado mucho la precisión del modelo.

Hemos usado la métrica accuracy en el código por las siguientes razones:

1. Problema de clasificación multi-clase: En el código, el problema de clasificación es multiclase, ya que se están clasificando diferentes tipos de comportamiento de los estudiantes. En este contexto, el accuracy proporciona una medida general de qué tan bien el modelo está clasificando todas las clases, lo cual es útil para evaluar el rendimiento global del modelo.

2. Distribución uniforme de clases: La distribución de clases en el conjunto de datos es más o menos uniforme, es decir, no hay un desequilibrio significativo entre las clases. El accuracy es una métrica razonablemente adecuada. Esto se debe a que cada clase contribuirá de manera similar al cálculo del accuracy, lo que permite una evaluación justa del rendimiento del modelo en todas las clases.
3. Simplicidad de interpretación: El accuracy es una métrica fácil de interpretar, ya que representa simplemente la proporción de predicciones correctas sobre el total de predicciones realizadas. Esto lo hace especialmente útil para la comunicación y la comprensión del rendimiento del modelo, teniendo en cuenta que este proyecto va dirigido a no especialistas.

Hemos usado métricas F1-score por ser fácil de aplicar pero no sensitivity, specificity porque tendríamos que haber alargado el código calculando la matriz de confusión para cada clase. Debemos tener en cuenta que es un objetivo primordial en este proyecto didactico simplificar al máximo el código para hacerlo accesible para profesores y alumnos de Enseñanza Secundaria.

Conclusiones

Ha quedado comprobado que podemos usar una red neuronal para predecir las Ventanas Panorámicas. Podemos ponerlo a disposición de cualquier profesor de Educación Secundaria Obligatoria o de cualquier padre de alumno para que puedan conocer su pronóstico.

Podríamos mejorar el algoritmo usando mas notas parciales de los alumnos que están disponibles a través del cuaderno de clase de Seneca (la WEB de la Consejería de Andalucia). Los continuos cambios en el diseño de dicho cuaderno han dificultado su uso en este proyecto. Hemos desarrollado una extensión para los navegadores Firefox y Chrome para facilitar el uso del cuaderno de clase de Seneca al margen de sus continuos cambios pero aun así no hemos sido capaces de usar los datos del cuaderno.

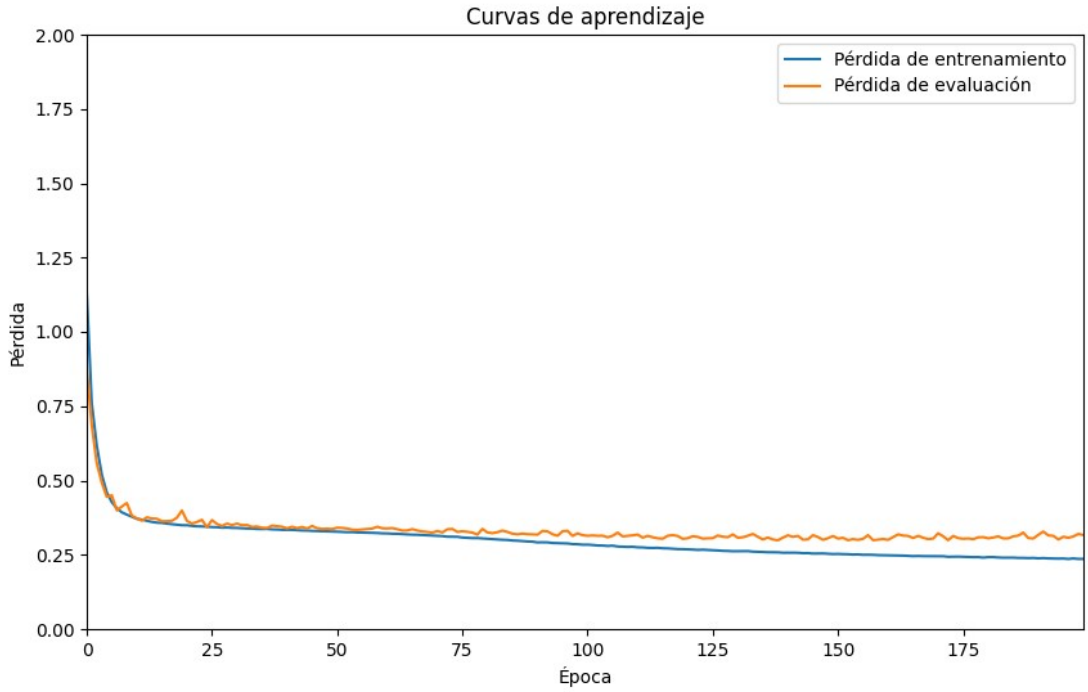
En cuanto a los datos de faltas y retrasos de los alumnos hemos observado vacíos en el registro de los datos. El procedimiento de registro actual no parece idóneo. Esperamos que en un futuro próximo se mejore el control de la asistencia de los alumnos con mejoras tecnológicas y de protocolo.

Finalmente hemos optimizado el entrenamiento con los siguientes paramentros.

- Fracción de datos para el entrenamiento: 0.8
- Fracción de datos para la evaluación: 0.2
- Tamaño de lote: 200
- Velocidad de aprendizaje: 0.1
- Neuronas por nota: 10
- Total de épocas: 200
- Máxima precisión: 0.85
- Época de máxima precisión: 175

- Total de datos: 107150

En total tenemos 10715 códigos que eran pocos para el entrenamiento, estos estaban relacionados con las faltas y las 9 notas del alumno. Al entrenar el algoritmo no obteníamos una buena convergencia por falta de datos pero hemos combinado las 9 notas de cada alumno para obtener 10 combinaciones y así tener el total de 107150 datos que hemos usado para el entrenamiento con óptimos resultados como se ve en la siguiente gráfica.



En la siguiente tabla observamos los resultados. Eventos seguido de un número indica cuantas veces aparece ese código en los datos a evaluar. Aciertos seguido de un % idica cuantas veces se ha acertado el código en la evaluación

Tabla de Resultados:

F1-score: 0.84

Tiempo total de entrenamiento en segundos: 721

Aciertos: 92%, eventos: 2452 (A) ABSENTISTA

Aciertos: 89%, eventos: 709 (NR) NO SE RELACIONA

Aciertos: 87%, eventos: 1674 (NP) NO PUEDE

Aciertos: 80%, eventos: 3127 (NS) NO SE COMPORTA

Aciertos: 79%, eventos: 2897 (SN) SE NIEGA A TRABAJAR O ESTUDIAR EN EL AULA

Aciertos: 79%, eventos: 4921 (NA) NO ATIENDE

Aciertos: 90%, eventos: 5650 (NE) NO ESTUDIA Y NO TRABAJA EN CASA

Acierto medio ponderado: 84%