

Análisis de sentimientos en comentarios de Among Us

Mauricio Alvarado

22 de febrero de 2023

1. Motivación

Among Us es un videojuego multijugador en línea desarrollado por la compañía estadounidense *InnerSloth*. Fue lanzado en 2018 con poco recibimiento. No obstante, el juego alcanzó una enorme popularidad en 2020 durante la pandemia de COVID-19, convirtiéndose en uno de los juegos más jugados del año.

La historia de *Among Us* se desarrolla en una nave espacial, donde los jugadores asumen el papel de miembros de la tripulación o impostores que intentan sabotear la misión. Los jugadores trabajan juntos para mantener la nave en funcionamiento mientras completan tareas, pero algunos jugadores son impostores cuyo objetivo es sabotear y matar a los demás jugadores sin ser descubiertos. A medida que los jugadores completan tareas, también investigan los asesinatos y acusan a otros jugadores de ser los impostores. Estos deben utilizar la estrategia y la comunicación para determinar quién es el impostor y votar para expulsarlo de la nave.

Pese a que el juego aún cuenta con mucha popularidad y recibió muchas mejoras de parte de los desarrolladores, durante los primeros meses de la pandemia el juego contaba con muchos errores y caídas en sus servidores debido a la alta demanda. Esto llevó a muchas personas a calificar el juego de manera negativa. Actualmente cuenta con una calificación de 3.8 en *Google Play Store*, lo cual lo posiciona muy por debajo de muchos juegos de igual o menor popularidad. Por ello, lo que motiva a este proyecto es poder clasificar los comentarios de un juego muy popular y de mis favoritos del 2020, que actualmente ha solucionado gran parte de sus problemas, pero que aún cuenta con una calificación muy baja.

Se tomará en cuenta los cinco mil comentarios más actuales en castellano y en Perú. La última actualización es considerando los comentarios hasta la fecha de 22 de febrero del 2023.

2. Hechos estilizados

.....

3. Metodología

3.1. Modelos

Se consideraron dos algoritmos de *machine learning* (*ML*) y dos arquitecturas neuronales de *deep learning* (*DL*). Cada uno corresponde a un modelo, por lo que se estimó 4 modelos diferentes.

Los dos algoritmos considerados para *ML* son los de Regresión Logística y de XGBoost. El primero fue elegido por ser uno de los algoritmos más usados para propuestas simples de clasificación. De hecho, es uno de los primeros algoritmos que se aprenden en diversos cursos a lo largo del mundo. El segundo fue elegido por ser uno de los algoritmos preferidos por muchos *data scientists* por su gran rendimiento en términos de eficiencia, ajuste y aprendizaje¹.

Para ambos algoritmos se estimó la mejor versión de estos basado en la selección de los hiperparámetros que maximicen el ajuste con los datos de entrenamiento. Se usó el método *GridSearchCV* que recoge las diferentes combinaciones de hiperparámetros definidos con anterioridad y estima el modelo con 5-Folds mediante *cross-validation*². Para el caso de Regresión Logística se consideró como parámetros de regularización a 0.001, 0.01, 0.1, 1, 10 y 100. Para el caso del algoritmo de XGBoost se consideró 100, 200 y 500 árboles de decisión; 3, 5 y 7 de profundidad máxima en cada uno de los nodos; y 0.1, 0.01 y 0.001 de *learning rate* en cada estimación.

Para el caso los algoritmos de *DL*, se propuso dos arquitecturas neuronales que cuentan con capas de *embedding*, *convolucionales*, *dropout*, *pooling* y *densas*. La primera estructura será llamada “intermedia” y la segunda será llamada “compleja”. La primera consiste de

3.2. Comparativa entre modelos

4. Selección de modelo

5. Resultados

6. Conclusiones

¹Para más detalle sobre sus ventajas y razones de su gran popularidad, se puede revisar lo siguiente: <https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/>.

²Consiste en separar las observaciones de entrenamiento en cinco grupos, estimar con cuatro de estos y reservar el quinto como un *testing* para estimar el ajuste. Tras realizar este ejercicio con las cinco combinaciones posibles, se obtiene cinco métricas de ajuste diferentes que serán promediadas. El objetivo de este ejercicio es corroborar la inexistencia de *overfitting*, lo cual se debería reflejar en métricas de ajuste en los datos de *test* muy similares al de los de entrenamiento.