



ESCUELA DE INGENIERÍA
FACULTAD DE INGENIERÍA

Changing the way we recommend videogames

Fernando Duarte, Rodrigo López, and Lucas Rodríguez,

Abstract—La industria de los videojuegos ha crecido enormemente a lo largo de los años, esto ha creado un gran interés en innovar en las formas de llegar a los consumidores, por supuesto los investigadores en sistemas recomendadores no se han quedado atrás, produciendo diversos sistemas que ocupando algoritmos del estado del arte recomiendan videojuegos.

Explorando distintos datasets de steam, nos podemos dar cuenta que el enfoque de venta de videojuegos no esta orientado hacia el consumidor, esto dado que muchos consumidores tienen juegos que nunca han jugado, es por eso que nosotros tomamos diversos algoritmos del estado del arte como lo son deepFM y wideDeep, y realizamos una recomendación basada en las horas de juego de cada usuario para así lograr recomendar juegos que realmente le produzcan valor al consumidor.

Index Terms—recommend, videogames, hours played, deep learning

I. INTRODUCCIÓN

Los videojuegos son un mundo que no para de crecer, año tras año se lanzan mas juegos lo que incentiva a esta industria a seguir creciendo. Esto sin duda lleva a que la inversión existente para innovaciones en este contexto siga aumentando día a día, por lo que investigar y ser agente de cambio en esta industria puede ser muy beneficioso.

Es importante reconocer que el beneficio a la industria no puede ser la única métrica a la hora de crear innovación, el consumidor también debiese ser considerado ya que la industria debe querer generar valor para el. En esto muchos sistemas recomendadores fallan ya que consideran la compra de un articulo como feedback implícito de que ese articulo le gusta al usuario, gracias a datasets de steam se puede observar que esto quizá no es del todo cierto, analizando el dataset steam 200k y steam videogames and bundle data, se puede ver que cerca del 40% de los juegos comprados no son jugados, esto hace pensar que quizá que el juego sea comprado no es la mejor métrica para señalar que un juego le gusta al usuario, Es por esto que se presenta una implementación de algoritmos del estado del arte como lo son deepFM y wideDeep, tomando en cuenta la cantidad de horas de juego como señal de si realmente un juego le gusta o no al usuario.

II. METODOLOGÍA

Para recomendar se escogieron dos dataset de steam, Steam Video Games: Recommend video games from 200k steam user interactions y Steam Video Game and Bundle Data, este ultimo contenía información de contexto por lo que para los algoritmos de deep learning se les agrego esta información adicional. los algoritmos utilizados como baseline son de recomendacion no personalizada, most popular. De collaborative filtering, global average y item KNN. estos tres son algoritmos bastante simples pero que igualmente sirven para tener un baseline mas completo. luego también se agrego un algoritmo de factorización matricial llamado BPR. con esta gama de algoritmos tenemos una base bastante fuerte para testear algoritmos mas avanzados, los algoritmos propuestos son todos de deep learning, estos son multiVae, multiDae, deepFM, wideDeep, DNN y DIN, todos estos algoritmos pertenecen al estado del arte por lo que el ejercicio de implementarlos y evaluarlos es muy beneficioso para una exploración de opciones a la hora de recomendar.

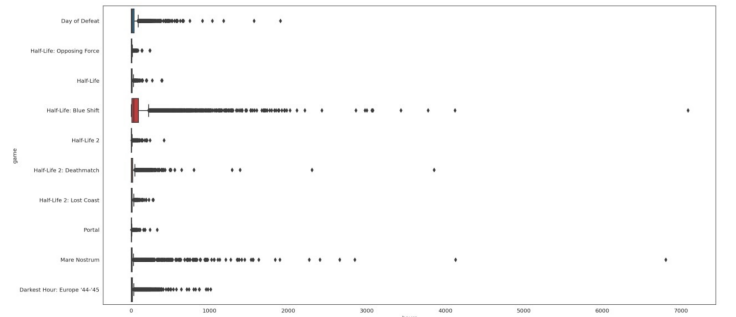


Fig. 1. BoxPlot de horas de juego en juegos mas populares.

Para testear estos algoritmos se ocuparon las métricas de NDCG, precisión y recall, estos algoritmos fueron testeados para distintos criterios de aceptación del usuario, es decir que se tomaron distintas medidas para decir si el usuario le gusto o no un juego. estas medidas fueron si el numero de horas jugadas por el usuario fueron mayor a un valor

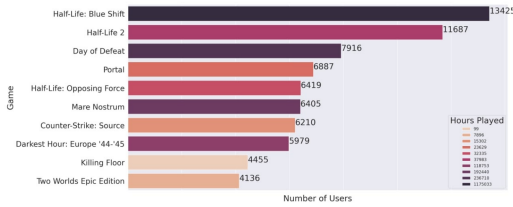


Fig. 2. Cantidad de compras de juegos mas populares.

entonces se considero como un juego positivo. este valor fue testeado para 0, 5 y 10 horas. también para los algoritmos que lo permiten se ocupo negative sampling, usando los juegos que no pasaron este threshold como negativos. También se realizo cross-validation para obtener métricas promedio en los resultados de los modelos.

Debido a la cantidad de modelos y el tiempo que demoran en ejecutarse no se pudieron ejecutar todas las pruebas para todos los modelos. pero si se realizaron para los modelos de interés que son deepFM, wideDeep, DIN y DNN.

III. ALGORITMOS

En esta sección se explicaran los principales algoritmos utilizados en este estudio, en primer lugar se ocupo un red neuronal profunda fernandoblueno se si dos capas ya es profunda, la cual posee dos capas ocultas, de 32 y 16 neuronas respectivamente, y ocupando la sigmoide como función de activación. esta red neuronal tiene la particularidad de que puede comprender las relaciones no lineales entre los items y usuarios, permitiendo ajustarlos de mejor manera que una maquina de factorización.

Luego se probó DeepFM, la cual también posee dos capas ocultas de 32 y 16 neuronas respectivamente y la función de activación sigmoide. a diferencia de DNN este modelo puede capturar tanto las relaciones lineales tanto como las no lineales por lo que puede obtener relaciones distintas a las obtenidas anteriormente. se esperaría que dado esta ventaja pueda obtener un mejor performance que una red neuronal profunda.

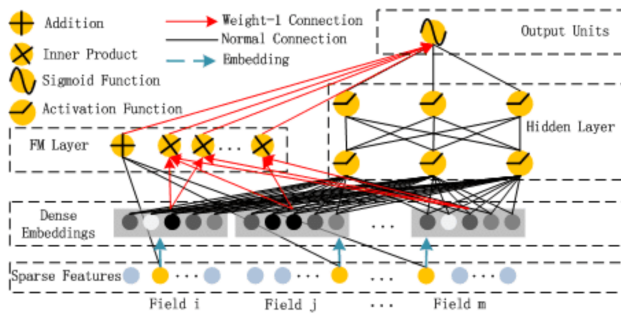


Fig. 3. Arquitectura DeepFM.

Luego testeamos el modelo wideDeep, este al igual que los otros dos posee dos capas ocultas de 32 y 16 neuronas respectivamente y la función de activación sigmoide y a diferencia de las anteriores tiene un lado wide que consiste en

pasar directamente el input a la función de activación lo que permite capturar relaciones diferentes a los modelos anteriores, al tener este componente wide también se espera que tenga un mejor performance que DNN.

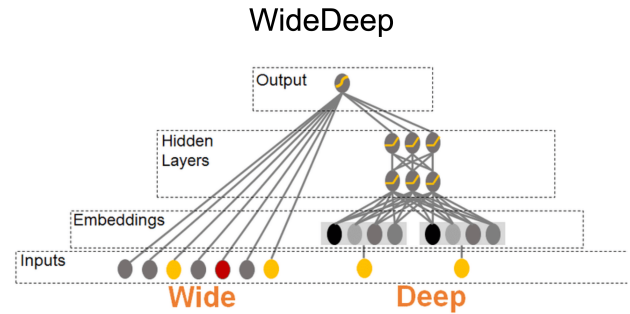


Fig. 4. Arquitectura WideDeep.

También se uso DIN el cual es un modelo con atención, con una arquitectura similar a las anteriores con dos capas ocultas de 32 y 16 neuronas y la función de activación sigmoide. y para la parte de atención también se ocuparon dos capas con 16 y 8 neuronas respectivamente. esta atención le permite enfocarse en lo que considera importante por lo que también puede capturar relaciones entre los usuarios y juegos nuevas no exploradas anteriormente. al no tener tanta información contextual se espera que esta atención no ayude enormemente al performance, ya que este tipo de arquitecturas suele ser usado en contextos donde la información contextual es mucho mayor.

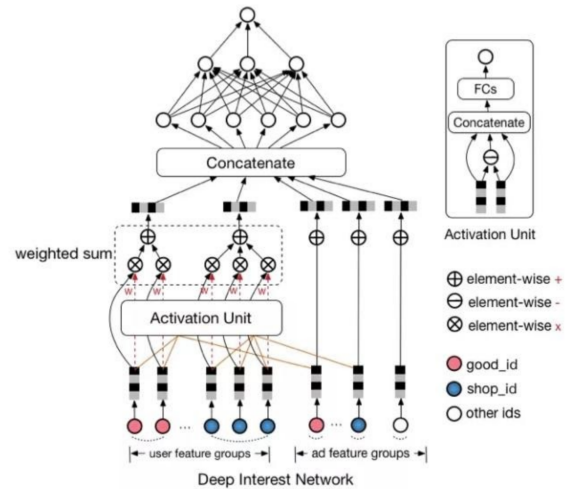


Fig. 5. Arquitectura DIN.

Por ultimo se testeamos dos autoencoders los cuales son MultiDae y MultiVae, los cuales cumplen la función de replicar en el output el input por lo que son capaces de completar la matriz item usuario y de esta manera con esta matriz completa luego se pueden generar recomendaciones, estos modelos fueron testeados con los items comprados por los usuarios por lo que están en desventaja al no poseer negative sampling. aun asi ambos son modelos del estado

del arte por lo que se espera poseen un buen rendimiento en general.

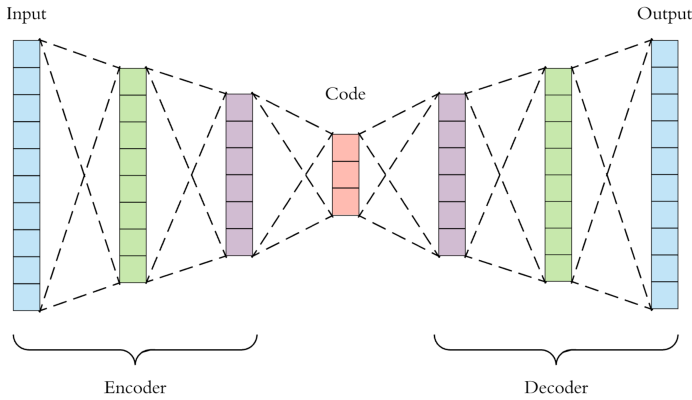


Fig. 6. Arquitectura MultiVAE MultiDAE.

Hay que recalcar que para el dataset de steam videogames and bundle data existe información contextual por que la comparación entre ambos datasets no es completamente idéntica. dado esto y que el dataset posee una mayor cantidad de datos se espera que se obtenga un mejor rendimiento para este dataset.

IV. RESULTADOS

A continuación, se presentan los resultados obtenidos por los distintos modelos variando los criterios para identificar si a un usuario le gusto un juego. Se utilizo un threshold de 0, 5 y 10 horas para ambos dataset. Se mostraran las métricas del mejor experimento el cual fue con 0 horas.

A. Steam 200K

	precision@10	recall@10	NDCG@10
Most Popular	0.081	0.2097	0.17
Global Average	0.0126	0.0236	0.0197
ItemKNN	0.0024	0.0117	0.0064
MultiVAE	0.037	0.4012	0.3469
MultiDAE	0.0477	0.4904	0.4287
DNN	0.2854	0.898	0.9679
DeepFM	0.2957	0.9263	0.9852
WideDeep	0.2781	0.8632	0.9365
BPR	0.08725	0.2124	0.176
DIN	0.2105	0.7018	0.6856

	precision@10	recall@10	NDCG@10
Most Popular	0.1148	0.5692	0.3621
Global Average	0.0489	0.2422	0.165
ItemKNN	0.0077	0.046	0.0284
MultiVAE	0.0126	0.6353	0.4147
MultiDAE	0.0125	0.6279	0.367
DNN	0.2092	0.6966	0.6828
DeepFM	0.2329	0.7784	0.7382
WideDeep	0.2126	0.7094	0.6922
BPR	0.1183	0.5783	0.3715
DIN	0.2105	0.6279	0.4122

V. CONCLUSIÓN

Como colusión se puede ver que a pesar de recomendar juegos siendo mas estrictos con el criterio de recomendación se pueden lograr resultados competitivos en las principales métricas de recomendación. Dado la gran cantidad de juegos existentes no existe problema en tratar de beneficiar al consumidor ya que lo que ocurre es que se le recomiendan juegos que le generaran mayor valor y aun así existirán suficientes juegos para recomendar como para continuar la demanda actual de compra, simplemente estos juegos serán mas probables de ser abiertos por el comprador por lo que estaríamos logrando un doble beneficio de ayudar a la industria y los jugadores.

Otro beneficio es que ya no se recomendaran juegos solo por hype, en la industria de los videojuegos existe una gran demanda por ciertos juegos que se hacen populares, muchas veces se compra porque cercanos lo compraron y no realmente considerando si el juego le gustara al comprador, es por esto que creemos que recomendando de esta manera podemos lograr que el comprador busque juegos menos comunes pero que realmente sean los que el quiere jugar.

Se pudo observar que la mejor separación para considerar si un juego le gusto o no al usuario es si es que este jugador lo abrió, ya que esta acción de instalar el juegos demuestra un interés superior que lograr una cantidad mínima de horas. Además existiendo juegos mas cortos que otros, al hacer el mínimo de horas un tiempo fijo se caía en el bias de decir que a los jugadores no le gustaban juegos solo por ser cortos.

Por otro lado se pudo ver como el aprendizaje profundo a ayudado a despegar a los sistemas recomendadores, con la gran facilidad de encontrar grandes cantidades de datos no queda duda que implementar estos modelos mas modernos es súper beneficioso ya que se puede triplicar el resultado en métricas de algoritmos tradicionales.

Por ultimo se observo que a pesar que el dataset steam videogames and bundle data tenia mas información y mayor numero de registro este rindió peor en las métricas, lo que

recalca la importancia de los sesgos propios de cada dataset y diferencias que tienen entre si, muchas veces existen supuestos como que un mayor dataset obtendrá mejores resultados, pero esto no se puede tomar a la ligera ya que, una buena limpieza del dataset, para obtener un dataset de calidad y la perdida de relevancia de artículos antiguos en este caso afecto mayormente que la cantidad de datos por lo que el dataset que se veía como mejor termino siendo el que obtuvo peores resultados.

REFERENCES

- Cheque, G., Guzman, J., Parra, D. (2019). Recommender systems for online video game platforms: the case of steam. In *Companion Proceedings of The 2019 World Wide Web Conference*, (pp. 763–771).
- de Castro Vieira, A. Brandao, W. C. Ga-eval: A neural network based approach to evaluate video games acceptance. Gong, J., Ye, Y., Stefanidis, K. (2019). A hybrid recommender system for steam games. In *International Workshop on Information Search*, (pp. 133–144). Springer.
- Hannula, R., Nikkila, A., Stefanidis, K. (2019). Gamerecs: Video games group recommendations. In *European Conference on Advances in Databases and Information*, (pp. 513–524). Springer.
- Huang, J. (2018). What can we recommend to game players?—implementing a system of analyzing game reviews. Master's thesis.
- Krasnianski, G. Kubasova, N. (2019). The saturation of the steam platform game market and the noticeability of the saturation by steam users. O'Neill, M., Vaziripour, E., Wu, J., Zappala, D. (2016). Condensing steam: Distilling the diversity of gamer behavior. In *Proceedings of the 2016 Internet Measurement Conference*, (pp. 81–95).
- Viljanen, M., Vahlo, J., Koponen, A., Pahikkala, T. (2020). Content based player and game interaction model for game recommendation in the cold start setting. *arXiv preprint arXiv:2009.08947*.
- Wang, D., Moh, M., Moh, T.-S. (2020). Using deep learning and steam user data for better video game recommendations. In *Proceedings of the 2020 ACM Southeast Conference*, (pp. 154–159).