Minería de Datos
IIC2433

Word2vec y cierre

Vicente Domínguez

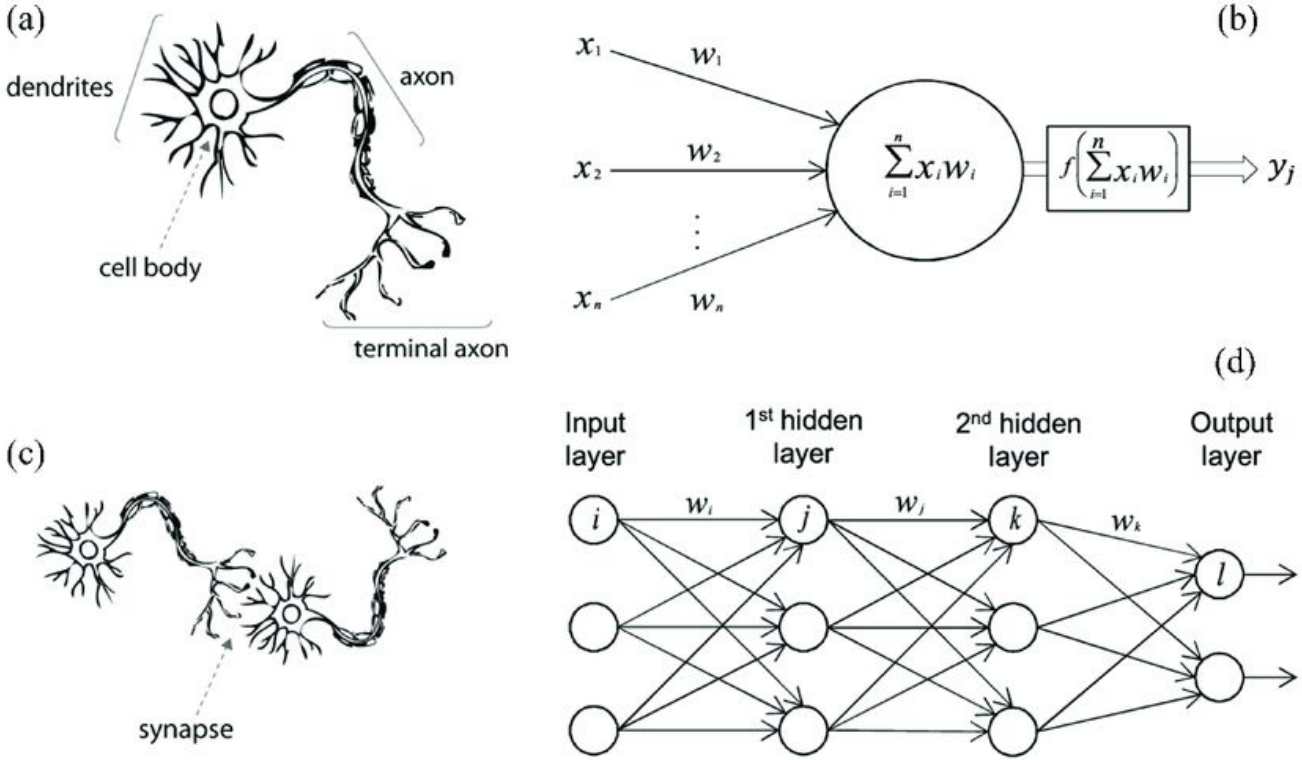
Modelos especializados

- Existen modelos especializados en tratar de capturar información relevante de los documentos de texto.
- Uno de los más conocidos es el modelo word2vec
- Para saber más de estos temas, recomiendo tomar el curso de Aprendizaje Profundo

word2vec

- Modelo basado en redes neuronales
- Trata de transformar las palabras de un conjunto de documentos en un vector de características, de ahí el nombre.
- Fue el origen de los modelos "2vec" (doc2vec, prod2vec, vec2vec)

Redes neuronales



Meng, Zhenzhu & Hu, Yating & Ancey, Christophe. (2020). Using a Data Driven Approach to Predict Waves Generated by Gravity Driven Mass Flows. Water. 12. 10.3390/w12020600.

word2vec

- Tal como en los métodos anteriores, se genera un vocabulario de todas las palabras existentes.
- Se genera una ventana de palabras, siendo el número de palabras que considera la oración.
- En esta ventana, palabra es transformada en un ***one hot encoding***

I	1	0	0	0	0
Like	0	1	0	0	0
watching	0	0	1	0	0
movie	0	0	0	1	0
enjoy	0	0	0	0	1

https://medium.com/towards-artificial-intelligence/an-intuitive-introduction-of-word2vec-by-building-a-word2vec-from-scratch-a1647e1c266c

word2vec

- La intuición detrás de este modelo es el hecho de que palabras utilizadas en contextos similares, deberían tener significado similar.

- Por ejemplo si tuviera las oraciones:
  - *I like watching a movie.*
  - *I enjoy watching a movie.*

- Intuitivamente entendemos que *like* y *enjoy* deben tener un significado similar.

- Ahora nos surge la duda ¿Cómo aprende este contexto?

word2vec

- Dado los vecinos de una palabra, trata de aprender un vector latente que lo represente, también conocidos como ***Embeddings***.
- Para cada palabra, genera un vector de entrada y uno de salida.
- La palabra objetivo la deja en el vector de salida, y el contexto o las palabras que la acompañan las deja en el vector de entrada.

Entrada

like	watching	movie
I	watching	movie
I	like	movie
I	like	watching
enjoy	watching	movie
I	watching	movie
I	enjoy	movie
I	enjoy	watching

I
Like
watching
movie
I
enjoy
watching
movie

Salida

https://medium.com/towards-artificial-intelligence/an-intuitive-introduction-of-word2vec-by-building-a-word2vec-from-scratch-a1647e1c266c

word2vec

- Cada uno de estos vectores es transformado en formato *one hot encoding*

Vectorized input

I	like	watching	movie	enjoy
0	1	1	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	0	0
0	0	1	1	1
1	0	1	1	0
1	0	0	1	1
1	0	1	0	1

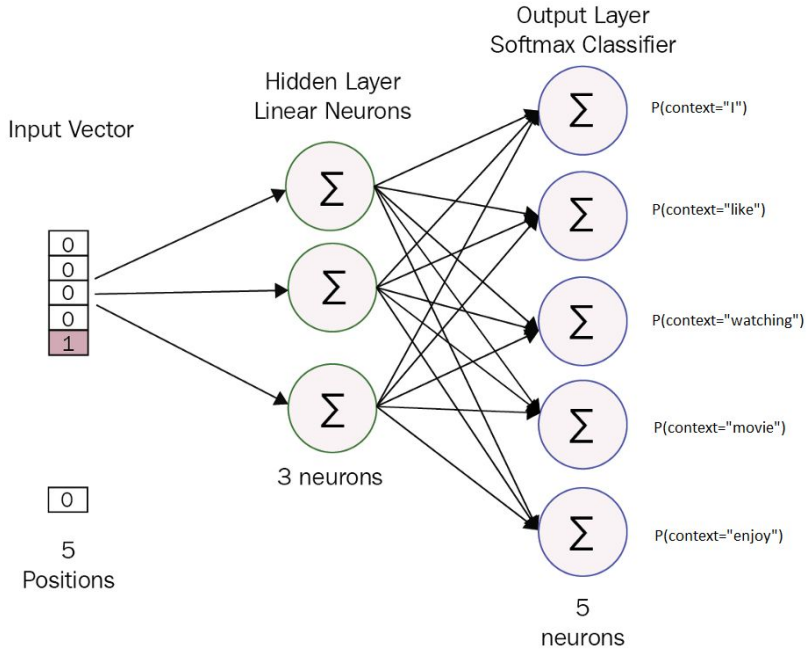
Output Vector

I	like	watching	movie	enjoy
1	0	0	0	0
0	1	0	0	0
0	0	1	0	0
0	0	0	1	0
1	0	0	0	0
0	0	0	0	1
0	0	1	0	0
0	0	0	1	0

https://medium.com/towards-artificial-intelligence/an-intuitive-introduction-of-word2vec-by-building-a-word2vec-from-scratch-a1647e1c266c

word2vec

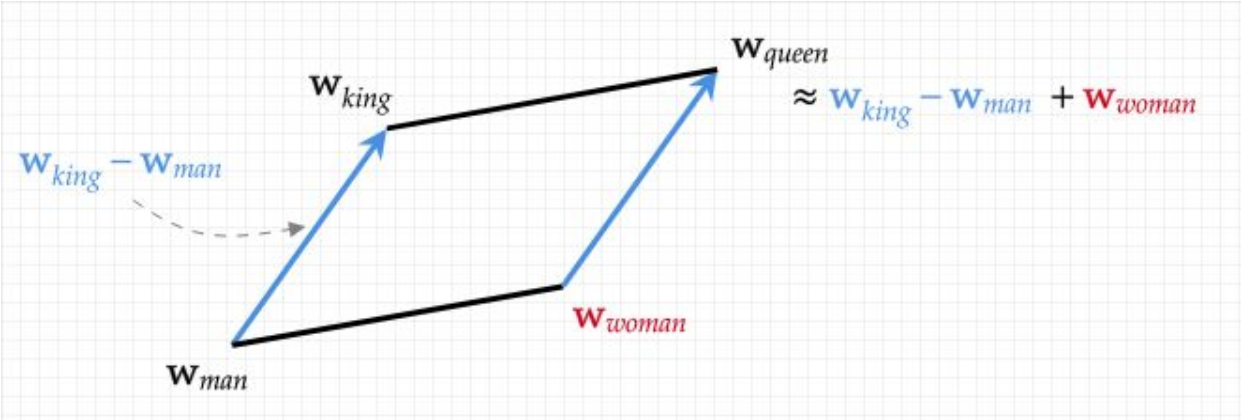
- Finalmente, la arquitectura del modelo es una red neuronal que trata de aprender un vector de *embedding* para cada palabra



https://medium.com/towards-artificial-intelligence/an-intuitive-introduction-of-word2vec-by-building-a-word2vec-from-scratch-a1647e1c266c

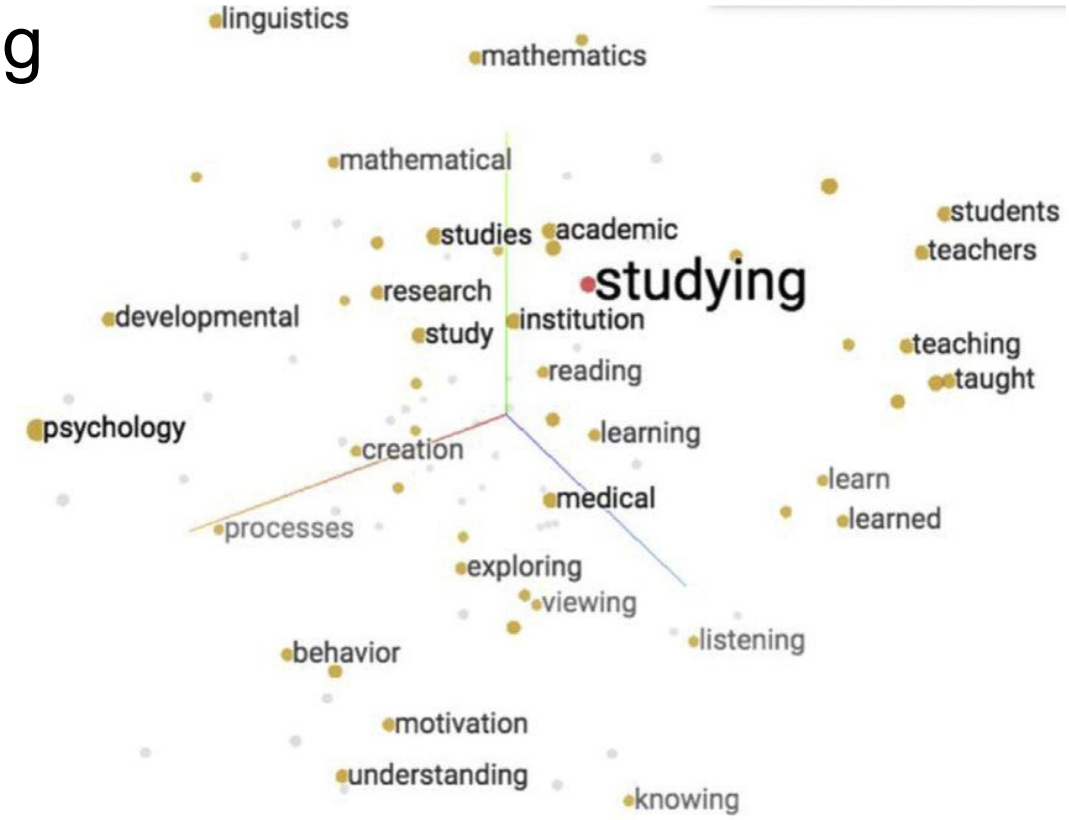
word2vec

- De los *embeddings* obtenidos, ocurren relaciones como la siguiente



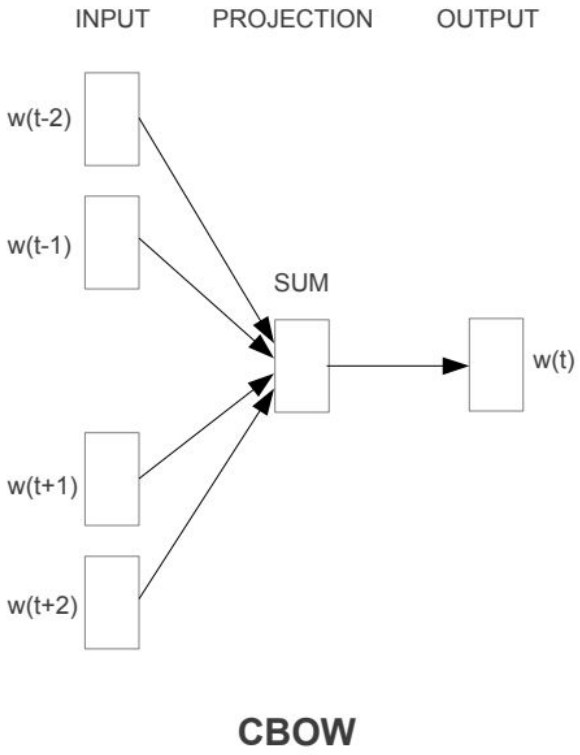
https://www.ed.ac.uk/informatics/news-events/stories/2019/king-man-woman-queen-the-hidden-algebraic-struct

Word Embedding

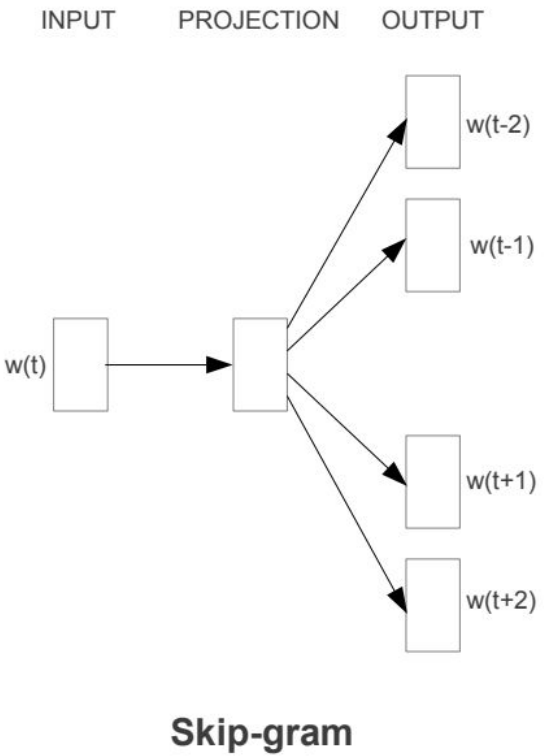


Kastrati, Zenun & Kurti, Arianit & Imran, Ali. (2020). WET: Word Embedding-Topic distribution vectors for MOOC video lectures dataset. Data in Brief. 28. 105090. 10.1016/j.dib.2019.105090.

word2vec



CBOW



Skip-gram

word2vec

Comentarios finales

- Modelo ampliamente utilizado en distintas áreas.
- Se puede utilizar con sus propios datos, o utilizar algún modelo pre entrenado.
- Existen modelos pre entrenados con conjuntos de datos masivos (todo wikipedia) incluso en español. https://github.com/dccuchile/spanish-word-embeddings
- Hay muchos otros modelos, sobre distintas áreas, pero van más allá de lo que podemos ver en este curso. Hay mucho por aprender aún 👀

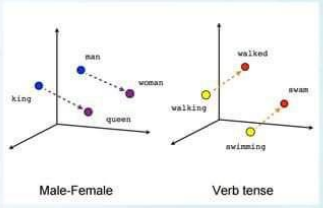
STOP DOING WORD EMBEDDINGS

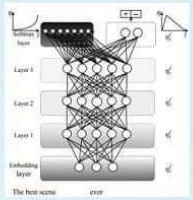
- WORDS WERE NOT SUPPOSED TO BE VECTORS
- HUNDREDS OF DIMENSIONS yet NO REAL-WORLD USE FOUND for putting WORDS in SPACE
- Wanted to anyway for a laugh? We had a tool for that: It was called HOMING PIGEONS

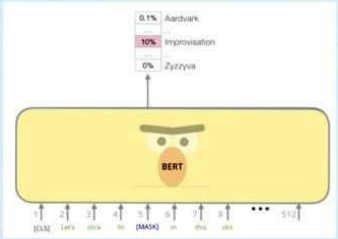
- "Yes please give me $W^O \in \mathbb{R}^{hd_v \times d_{\text{model}}}$ of something. Please give me $R_k = \{\mathbf{x}_k^{LM}, \overrightarrow{\mathbf{h}}_{k,j}^{LM}, \overleftarrow{\mathbf{h}}_{k,j}^{LM} \mid j = 1, \ldots, L\} = \{\mathbf{h}_{k,j}^{LM} \mid j = 0, \ldots, L\}$ of it" - Statements dreamed up by evil wizards

LOOK at what computer scientists have been demanding your Respect for all this time, with all the GPUs and corpora we built for them

(This is REAL embeddings, done by REAL computer scientists):







????? ??????? ?????????????

"Hello I would like tensor([0.8461, 1.2488, 0.2553, ..., -1.0341, 0.1078, -1.7277]) apples please"

They have played us for absolute fools

Tópicos Avanzados

- Las áreas de Aprendizaje de Máquinas y Minería de datos son de las que más se han desarrollado en el último tiempo.
- El aumento de la tecnología (hardware) ha permitido la implementación de modelos cada vez más complejos.
- Los resultados que han demostrado tener estos algoritmos han hecho que los científicos y la industria pongan sus ojos y esfuerzos en desarrollar más estas áreas.
- De aquí en adelante queda un gran camino por delante, pero quiero al menos contarles sobre 3 áreas que me han parecido relevantes en mi formación.
- Estas son:
    - Recommender Systems
    - Reinforcement Learning
    - Deep Learning

Sistemas Recomendadores

- Área que ha tenido un gran crecimiento en los últimos años debido al impacto de la personalización.
- Varias compañías lo tienen integrado como parte de su modelo de negocios.




Sistemas Recomendadores

- De forma simple, la idea es priorizar un conjunto de productos para un usuario, dentro de un mar de productos.
- A diferencia de todos los métodos que hemos visto hasta ahora, en los algoritmos de recomendación hay entidades que definen al problema: los usuarios y los productos.

	userId	movieId	rating	timestamp
0	1	1	4.0	964982703
1	1	3	4.0	964981247
2	1	6	4.0	964982224
3	1	47	5.0	964983815
4	1	50	5.0	964982931
5	1	70	3.0	964982400
6	1	101	5.0	964980868

https://www.codespeedy.com/build-recommender-systems-with-movielens-dataset-in-python/

Sistemas Recomendadores

- Más formalmente, una definición matemática del problema sería la siguiente (Adomavicius et al. 2007)

$$\forall c \in C, s'_c = \underset{s \in S}{\operatorname{argmax}}\, u(c, s)$$

$$u : C \times S \rightarrow R, \textit{ funcion de utilidad}$$

$$R : \textit{ conjunto recomendado de items}$$

$$C : \textit{ conjunto de usuarios}$$

$$S : \textit{ conjunto de items}$$

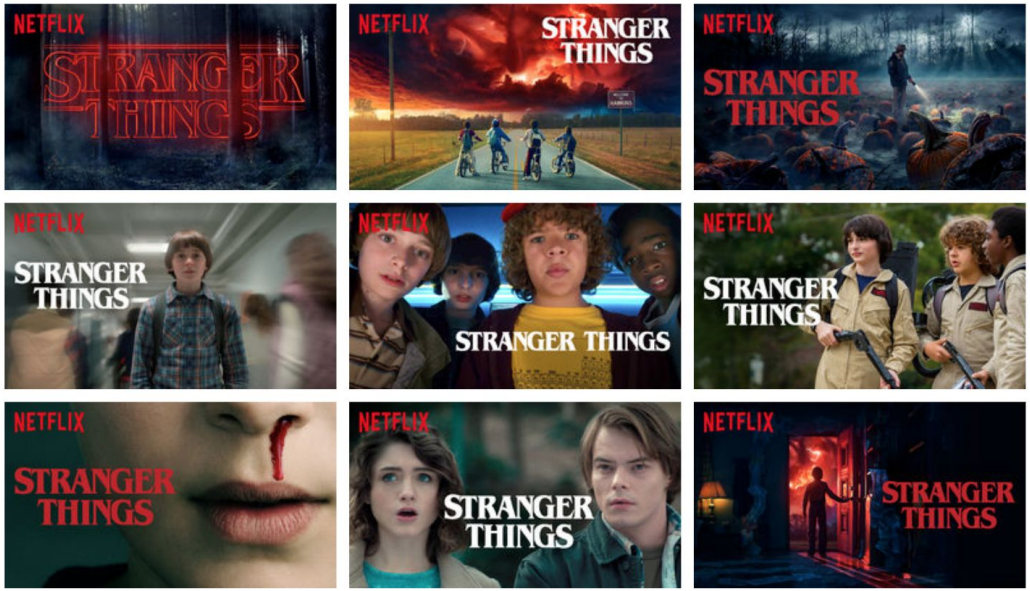
https://dparra.sitios.ing.uc.cl/classes/recsys-2017-2/clase1_recsysintro.pdf

Sistemas Recomendadores

- Dentro de los problemas típicos de esta área se encuentran: rating prediction y ranking prediction.
- Hay un curso completo que pueden tomar si están interesados en este tema. Sistemas Recomendadores - IIC3633
- También hay librerías con los algoritmos clásicos de predicción de rating y ranking. https://github.com/gasevi/pyreclab

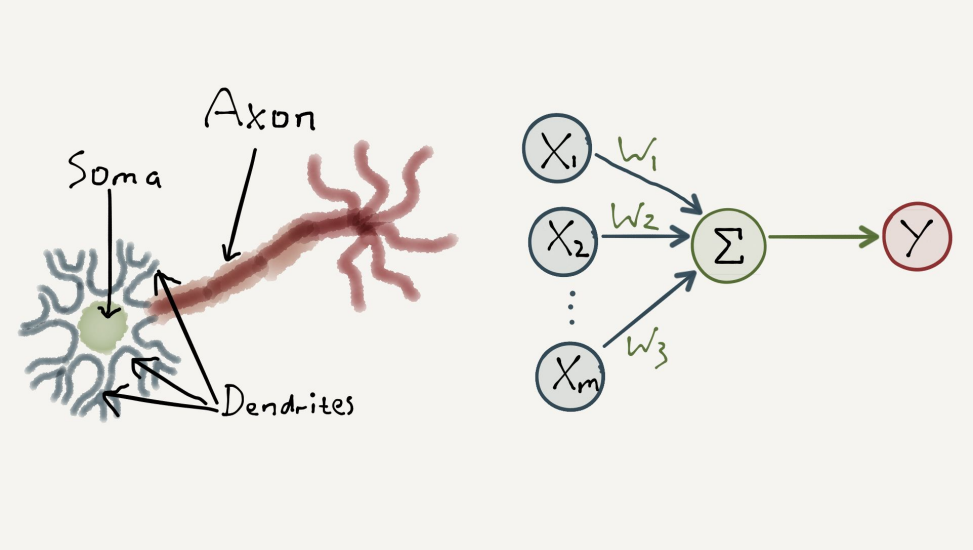
Sistemas Recomendadores (Aplicación)

- A finales del 2017 Netflix publicó un artículo donde explicaban que estaban llegando a un nivel de personalización tan granular que hasta decidían qué imagen mostrar en cada programa o película para cada usuario.
- Post del artículo



Deep Learning

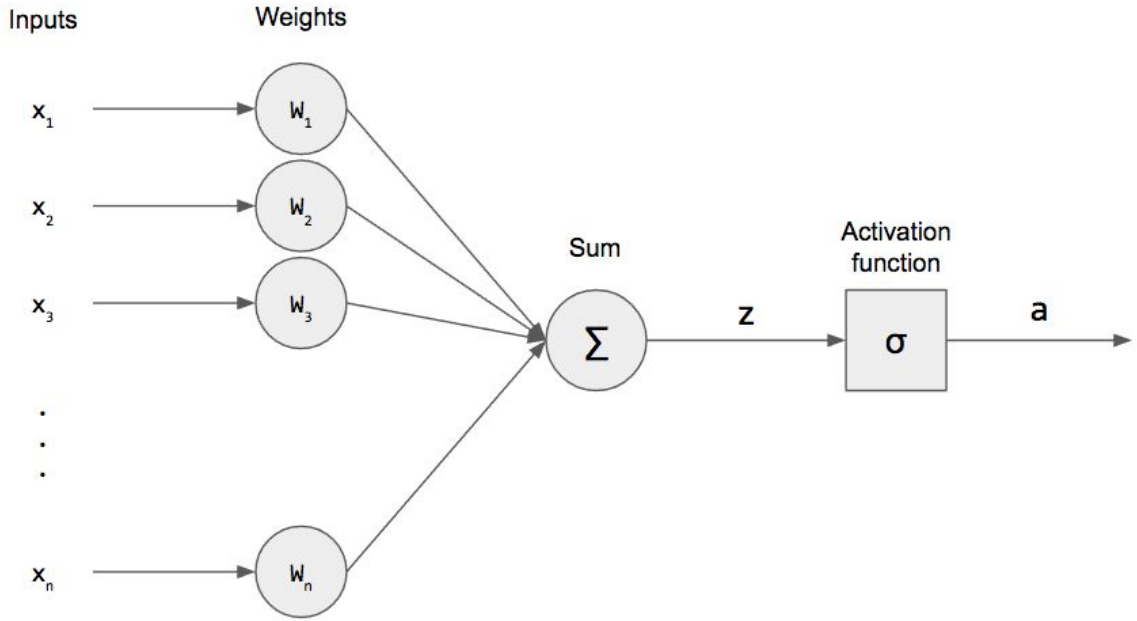
- Área de Machine Learning enfocada en el desarrollo de modelos de redes neuronales profundas.
- Estos modelos tratan de modelar y simular el comportamiento de las neuronas del cerebro humano, en el proceso de aprendizaje.



https://jontysinai.github.io/jekyll/update/2017/11/11/the-perceptron.html

Deep Learning

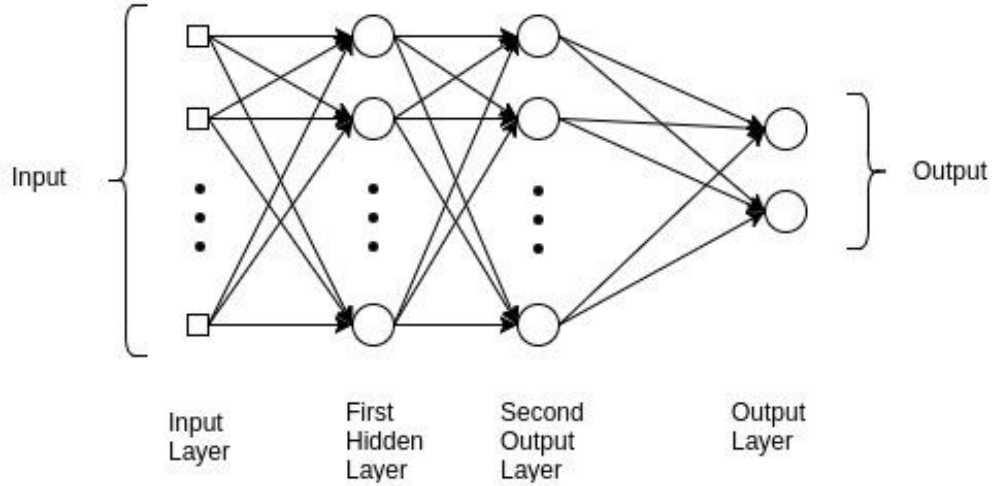
- La unidad básica es el perceptrón, la cual es muy similar a la regresión logística con unas cuantas variaciones.



https://pythonmachinelearning.pro/perceptrons-the-first-neural-networks/

Deep Learning

- Los modelos finales consisten en una red profunda de capas de perceptrones



https://medium.com/@AI_with_Kain/understanding-of-multilayer-perceptron-mlp-8f179c4a135f

Deep Learning

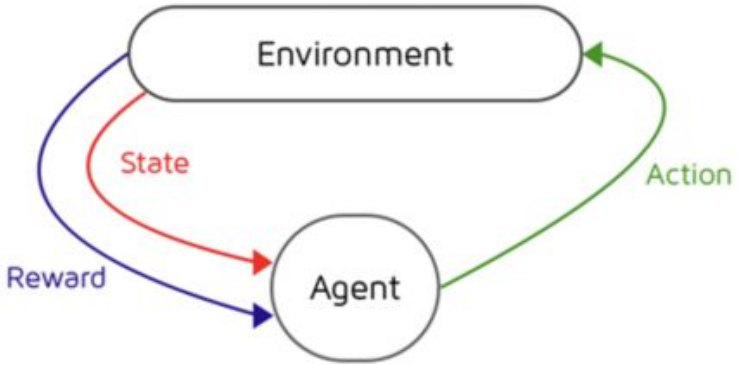
- Las ideas de las redes neuronales surgieron en los 90's, pero no pudieron ser implementadas debido a la diferencia entre la teoría y el hardware disponible.
- Gracias al desarrollo del poder de cómputo de las tarjetas gráficas, se pueden ejecutar estos modelos en paralelo en tiempos razonables.
- Actualmente nuevas tarjetas diseñadas para este tipo de tareas, son conocidas como TPUs (Tensor Processing Units).
- Si quieren saber más sobre estos temas, existe el curso de Aprendizaje Profundo - IIC3697.

Deep Learning (Aplicación)

- Una de las principales tareas que han mostrado grandes avances gracias al Deep Learning ha sido el reconocimiento facial.
- No solo eso, ya crear modelos que generen caras que nunca han existido de forma automática.
- Cada vez que ingresan a este sitio, encontrarán una persona diferente que no existe.
- El diario New York Times realizó un artículo sobre este modelo, explicando cuales son las intuiciones detrás de él. Pueden ver el artículo aquí.

Reinforcement Learning

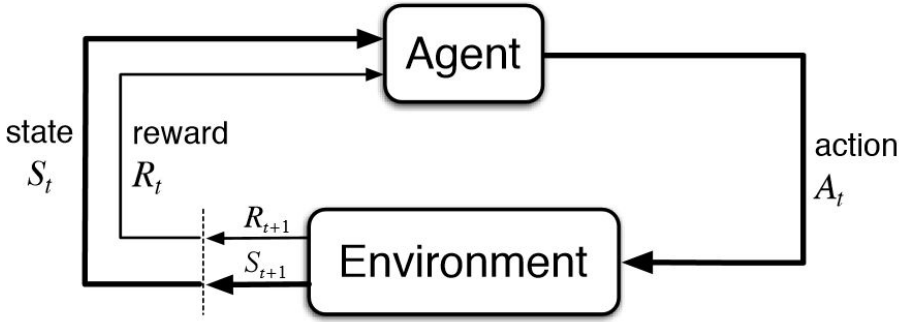
- Tipo de aprendizaje que surge de la psicología y la matemática.
- Se sustenta en el hecho de que los seres interactúan con un ambiente. Y de esas interacciones pueden surgir aprendizajes.
- Es un tipo de aprendizaje distinto a los vistos en el curso.



Reinforcement Learning

- Un conjunto de datos para este tipo de aprendizaje sería el siguiente

Timestep #	State	Action	Reward
1	S1	A1	-1
2	S2	A2	0
3	S3	A3	1
4	S4	A4	0



Reinforcement Learning

- Considere los siguientes datos

Veces jugado	70	10	30	50
% Éxito	55%	55%	25%	50%

¿Qué máquina deberíamos jugar ahora dada la evidencia?

Reinforcement Learning

- Ahora, considere los siguientes datos

Ítem recomendado	1	2	3	4
Veces recomendado	70	10	30	50
% Éxito	55%	55%	25%	50%

¿Qué máquina deberíamos jugar ahora dada la evidencia?

Reinforcement Learning

- Dado que tenemos acciones, y no siempre sabemos cual será la recompensa que nos dará el ambiente, surge de aquí el problema de **Explotación vs Exploración**

Exploración

Dada la evidencia, podemos decidir explorar y probar una máquina al azar.

Permite probar nuevas opciones y encontrar una jugada mejor que la evidencia.

Explotación

Podemos seguir la evidencia y jugar la máquina que mejor probabilidad de ganancia tiene hasta el momento.

Permite maximizar la ganancia del sistema.

Reinforcement Learning

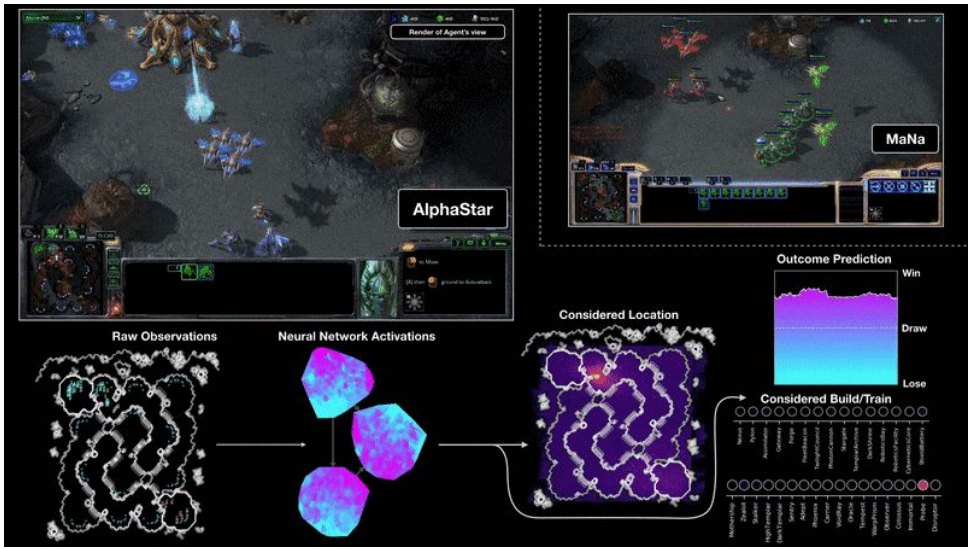
- No hay cursos disponibles en la UC para aprender más de este tema.
- Pero hay libros y cursos en línea, les dejo un link a un libro muy bueno acá.

Reinforcement Learning (Aplicación)

- Una de sus aplicaciones más conocidas es en el ámbito de juegos.
- Debido a que pueden ir aprendiendo de los errores, interactuando con el ambiente, no es necesario conocer todo el espacio de acciones. Funciona bien en juegos de estrategia.

Reinforcement Learning (Aplicación)

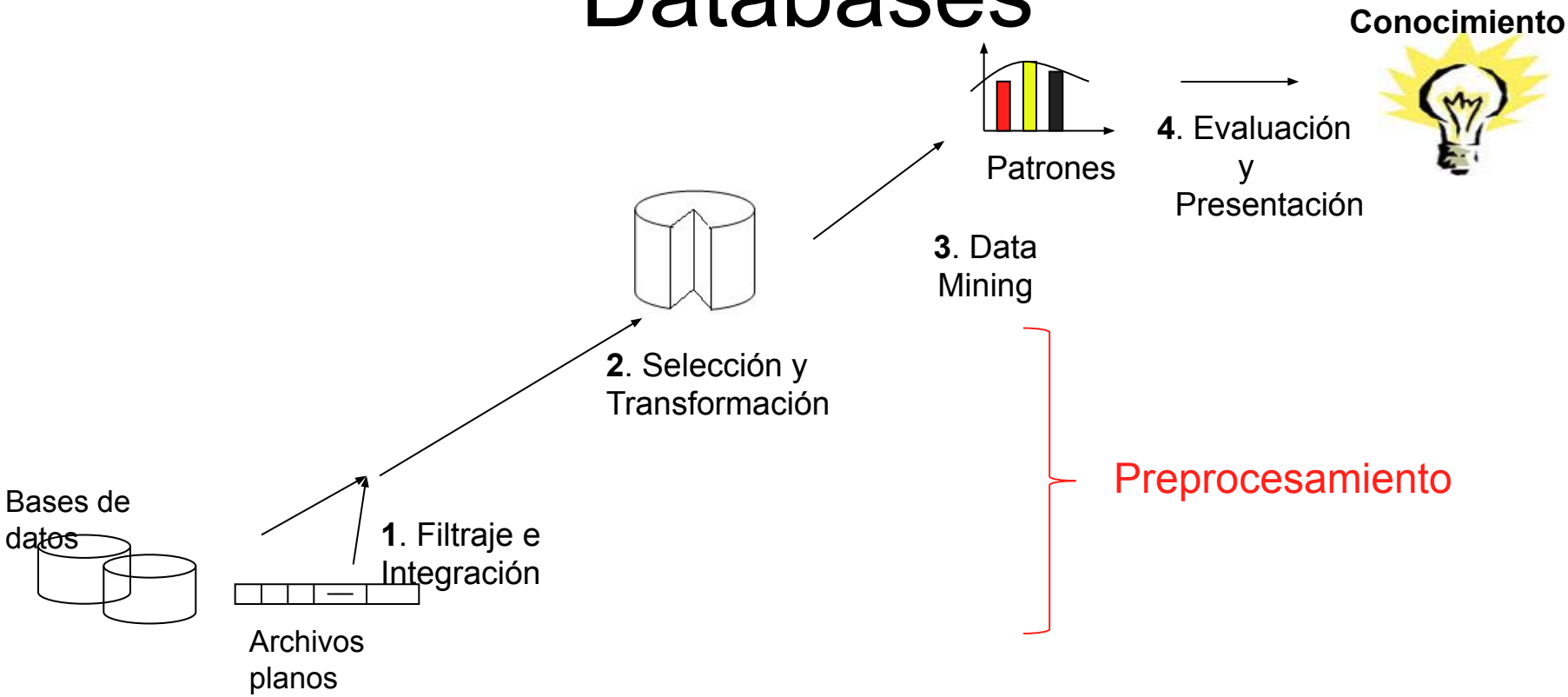
- El Go fue uno de los primeros juegos dominados por estos métodos
- Otro conocido ejemplo es el juego de estrategia StarCraft, donde un bot pudo ganarle a todos los jugadores profesionales contra los que se enfrentó
- Pueden leer un artículo sobre esto acá.



Cierre

- Durante el curso partimos desde la extracción de datos, hasta la evaluación de modelos.
- Más que aprender sobre matemática o estadística, lo más valioso es que entiendan la noción que hay detrás de todo.
- Lo importante es ser curioso, no quedarse con solo llegar y utilizar un modelo.

Knowledge Discovery in Databases



Habilidades

- Aparte de los contenidos vistos en el curso, espero que hayamos aprendido a:
  - Entender cómo funcionan los modelos de Minería de Datos
  - Poder programar algún modelo visto matemáticamente
  - Comprender y poder explicar el buen y mal desempeño de algún modelo de Minería de Datos
  - Tener la base para poder investigar en áreas más especializadas de Machine Learning
  - Poder modelar problemas y saber como poder aplicar minería de datos sobre ellos.

Aún no se acaba todo

- Ojo que aún queda el proyecto y la última tarea.

Nos vemos

- Nos veremos una última vez en las presentaciones de los proyectos.
- Recuerden que cuentan con sus ayudantes para lo que sea.
- Y siempre podrán escribirme después de que dejen de ser mis alumnos, al correo vidominguez@uc.cl
- ¡Mucho éxito!

Me: *uses machine learning*

Machine: *learns*

Me:

Referencias

- Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. Knowledge and Data Engineering, IEEE Transactions on, 17(6), 734-749.