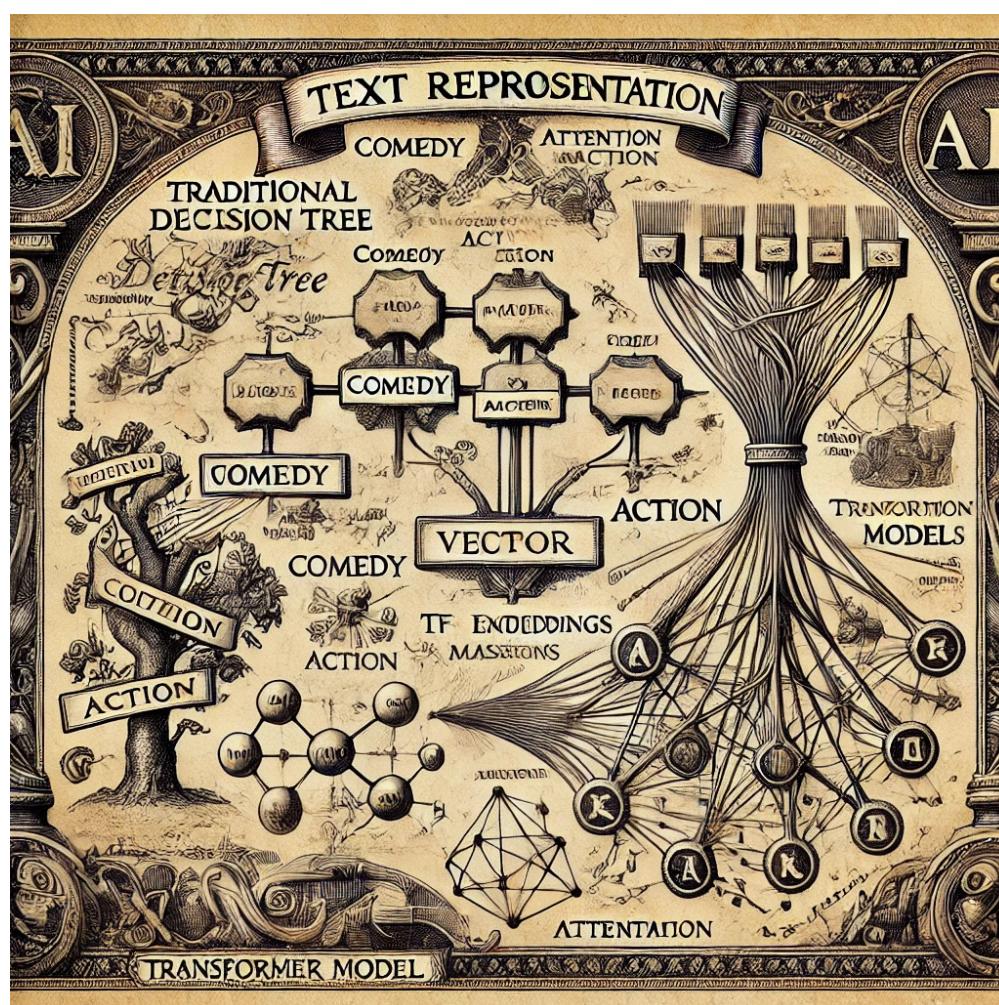


“El Oráculo Generativo le susurró a entonces a Shun que existía un reino más profundo y sutil, donde cada pensamiento, palabra o idea tomaba la forma de vectores luminosos en un espacio infinito. Era allí, en ese próximo destino, donde lo esperaba el mítico Embeddinator, capaz de transformar todo conocimiento en coordenadas precisas y transparentes. Intrigado, Shun emprendió el nuevo viaje hacia ese reino oculto, dispuesto a explorar los secretos de la vectorización y descubrir el verdadero lenguaje del cosmos.”

Representaciones en Aprendizaje Profundo y Procesamiento del Lenguaje Natural

x Matias Nuñez

En los últimos años, las tecnologías del lenguaje han experimentado un avance espectacular. Desde traductores automáticos que funcionan cada vez mejor hasta asistentes virtuales capaces de mantener conversaciones, estas herramientas están transformando nuestra relación con el lenguaje y la comunicación. Detrás de estos avances se encuentra un enfoque llamado "aprendizaje profundo" (deep learning), una rama de machine learning, que ha revolucionado cómo las máquinas procesan y generan lenguaje.



Para traductores, filólogos, lingüistas y profesionales de las ciencias sociales, comprender los fundamentos de estas tecnologías se ha vuelto cada vez más relevante. No se trata de convertirse en programadores, sino de entender cómo estas herramientas conceptualizan el lenguaje, qué pueden hacer realmente, y cuáles son sus limitaciones. Por otro lado, programar hoy es mucho mas facil que hace un par de años. Uno puede programar al estilo [vibe coding](#) describiendo el programa a hacer a la computadora a traves de lenguaje natural mediante prompts. Para entender como llegamos a esto, nesecitamos entender a fondo el concepto de embeddings o incrustaciones. En este texto vamos a explorar cómo los sistemas de inteligencia artificial actuales representan el lenguaje y qué nos dice esto sobre la naturaleza misma de la comunicación humana.

Sueñan las máquinas con helados de chocolate?

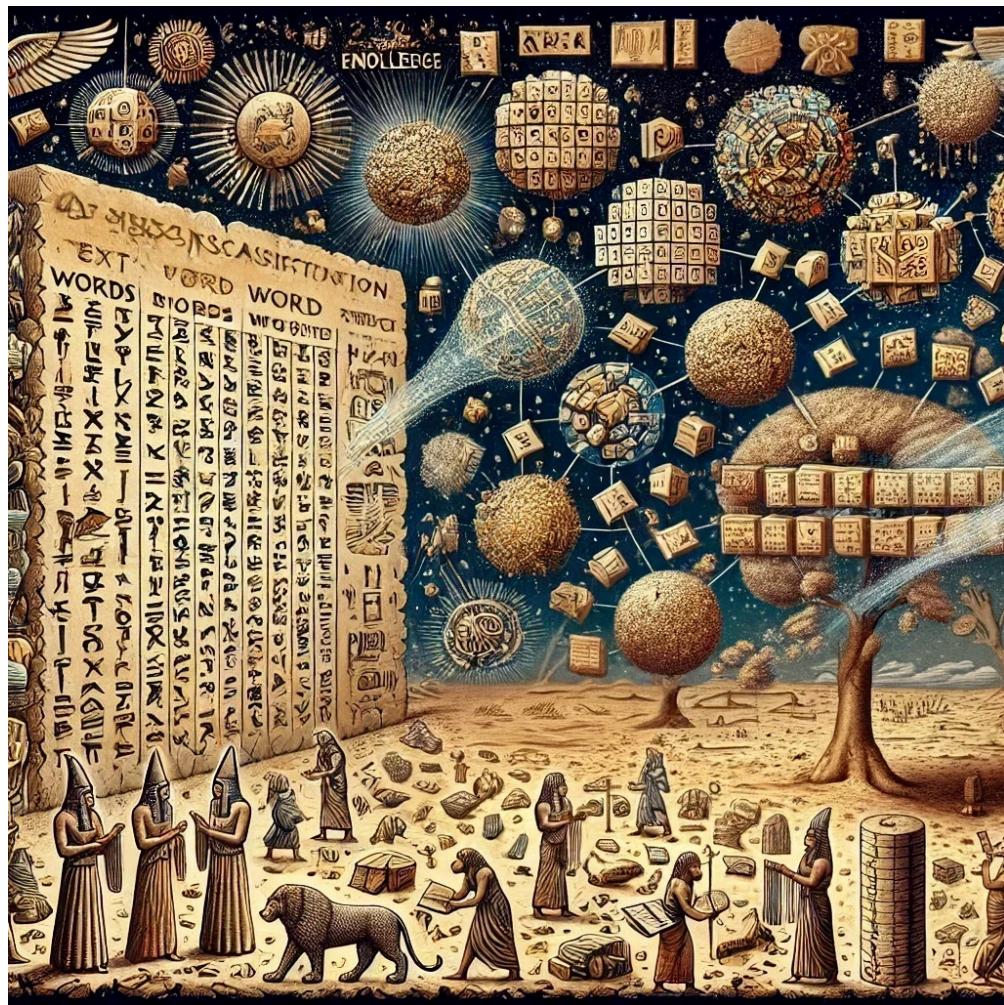
Cuando aprendemos un idioma, desarrollamos un sistema mental que relaciona palabras, significados y estructuras gramaticales. Las máquinas, sin embargo, no tienen acceso directo a la experiencia humana ni a nuestros conceptos. Entonces, ¿cómo procesan el lenguaje?

La respuesta está en las **representaciones numéricas**. Los sistemas de aprendizaje profundo transforman las palabras en secuencias de números (vectores) que capturan aspectos de su significado y uso. Esta transformación permite que las computadoras, que sólo entienden números, puedan indirectamente procesar información lingüística.

El paso natural en la evolución de la representación textual fue moverse desde vectores dispersos basados en fecuencias de aparicion de palabras (TF-IDF) hacia modelos neuronales capaces de generar **embeddings** densos. Estos embeddings son vectores de tamaño manejable (típicamente entre 100 y 768 dimensiones, según la arquitectura) donde la ubicación de cada palabra en este espacio multidimensional refleja su significado y relaciones semánticas.

Embeddings de palabras: Cartografiando el Significado

Los embeddings de palabras representan una de las innovaciones más fascinantes en el procesamiento del lenguaje natural. A diferencia de TF-IDF, estos modelos no se basan en simples conteos, sino que aprenden representaciones densas donde las relaciones semánticas se codifican como relaciones espaciales.

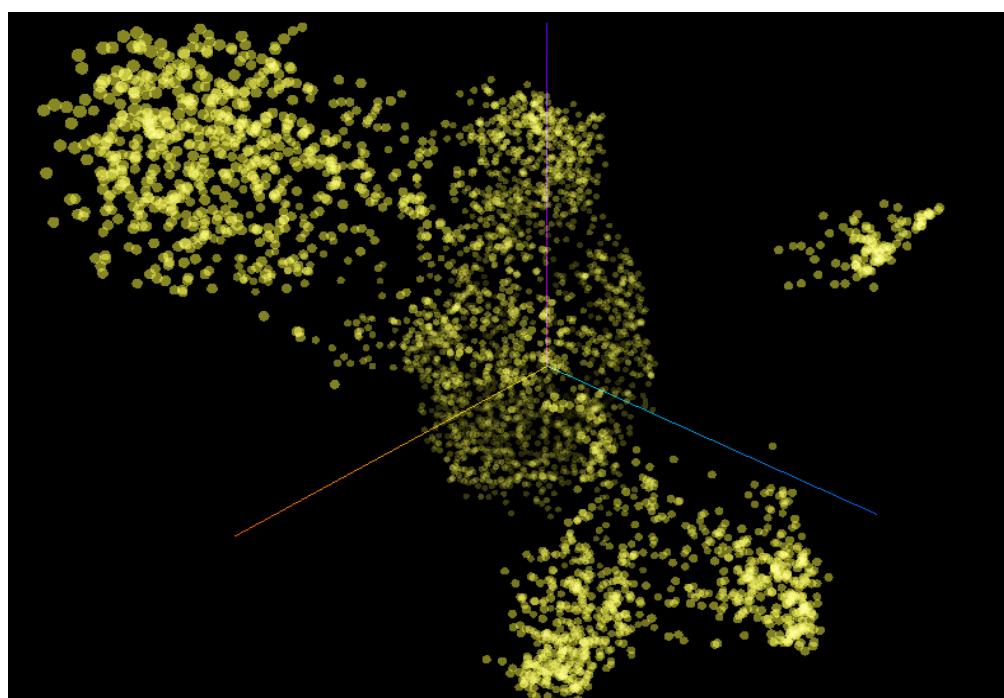


Hay muchas formas de hacer esto, muchas abstracciones posibles.

Una de las primeras ideas relacionadas, la idea de representaciones distribuidas, es atribuida a Geofry Hinton, uno de los dos premios nobel en fisica en 2024 por sus contribuciones en Inteligencia articial en los 80. Premio nobel muy polemico en el ambiente de la fisica, ya que el sector mas fundamentalista clama que esas contribuciones no son propiamente "fisica". Sin embargo usaron conceptos de la fisica estadistica , base de la termodinamica, desarrollada en la revolucion industrial, en principio por ingenieros que querian sacar la mayor cantidad de energia utilizable al carbon y llegaron a principios fisicos luego desarrollados teoricamente por la fisica estadistica de boltzman entre otros.

Quien quiera conocer mas sobre porque los embedding de palabras son poderosos, puede charlar sobre el paper [A Neural Probabilistic Language Model \(Bengio, et al. 2003\)](#) con un LLM .

Luego los embeddings de palabras propiamente dichos fueron desarrollados en ([Bengio et al. 2001](#); [Bengio et al. 2003](#)), antes de la revancha del aprendisaje produdsim en 2006 cuando las redes neuronales estaban fuera de moda, en el invierno de la IA, como se apoda al periodo hasta fines de los 80..donde el campo llego a un grado de maduracion habiendo entendido como entrenar a las redes neuronales, pero a la realizacion que no habia suficientes datos para hacerlas utiles. Luego de la adopcion masiva de internet, y con la aparicion de datos masivos, se retomo el interes por las redes neuronales, potenciado por las empresas que ofrecian servicios a cambio de usar los datos del usuario para entrenar sus modelos.



Embedding de la novela *El Quijote* de Cervantes.

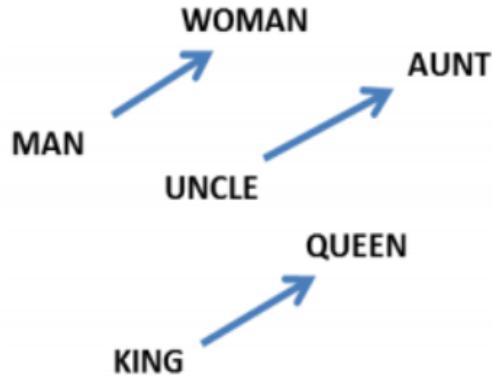
Modelos pioneros como Word2Vec o GloVe se basaron en una intuición lingüística fundamental: "**las palabras que aparecen en contextos similares tienden a tener significados similares**". Esta idea, propuesta hace décadas por los lingüistas John Rupert Firth y Zellig Harris (mentor de Noam Chomsky), cobró vida computacional cuando estos algoritmos comenzaron a analizar enormes corpus textuales, observando patrones de co-ocurrencia para posicionar palabras en un espacio vectorial continuo.

Por ejemplo, en el modelo [Word2Vec](#) entrenado en inglés, podríamos tener representaciones vectoriales como:

Vector("king") = (0.50, -0.23, 0.36, ...)
Vector("queen") = (0.48, -0.52, 0.40, ...)
Vector("man") = (0.37, -0.25, 0.21, ...)
Vector("woman") = (0.35, -0.54, 0.23, ...)

Lo extraordinario de estos embeddings es que capturan intuitivamente relaciones lingüísticas importantes. Palabras semánticamente similares como "doctor" y "physician" aparecen cerca en este espacio. Términos relacionados temáticamente como "hospital", "nurse" y "patient" forman agrupaciones naturales. Incluso emergen regiones específicas para diferentes categorías conceptuales: países, profesiones, animales, términos técnicos, etc.

Álgebra de Significados



From Mikolov *et al.*
(2013a)

Quizás lo más asombroso de estos embeddings es que las relaciones semánticas entre palabras se manifiestan como operaciones vectoriales regulares. El ejemplo clásico es la analogía:

$$\text{Vector("king")} - \text{Vector("man")} + \text{Vector("woman")} \approx \text{Vector("queen")}$$

Esta operación vectorial revela que el modelo ha capturado implícitamente la relación de género. De manera similar:

$$\text{Vector("Paris")} - \text{Vector("France")} + \text{Vector("Italy")} \approx \text{Vector("Rome")}$$

`Vector("walked") - Vector("walk") + Vector("run") ≈ Vector("ran")`

Estas relaciones no fueron programadas explícitamente, sino que emergieron durante el entrenamiento a partir de patrones estadísticos en los textos. Otras relaciones que podemos observar incluyen:

$$\text{Vector("good")} - \text{Vector("better")} \approx \text{Vector("bad")} - \text{Vector("worse")}$$

$$\text{Vector("quick")} - \text{Vector("quickly")} \approx \text{Vector("slow")} - \text{Vector("slowly")}$$

$$\text{Vector("water")} - \text{Vector("liquid")} \approx \text{Vector("air")} - \text{Vector("gas")}$$

Relationship	Example 1	Example 2	Example 3
France - Paris	Italy: Rome	Japan: Tokyo	Florida: Tallahassee
big - bigger	small: larger	cold: colder	quick: quicker
Miami - Florida	Baltimore: Maryland	Dallas: Texas	Kona: Hawaii
Einstein - scientist	Messi: midfielder	Mozart: violinist	Picasso: painter
Sarkozy - France	Berlusconi: Italy	Merkel: Germany	Koizumi: Japan
copper - Cu	zinc: Zn	gold: Au	uranium: plutonium
Berlusconi - Silvio	Sarkozy: Nicolas	Putin: Medvedev	Obama: Barack
Microsoft - Windows	Google: Android	IBM: Linux	Apple: iPhone
Microsoft - Ballmer	Google: Yahoo	IBM: McNealy	Apple: Jobs
Japan - sushi	Germany: bratwurst	France: tapas	USA: pizza

Relationship pairs in a word embedding. From Mikolov *et al.* (2013b).

1

Se puede intuir como esto puede pasar esto.

You've seen all the words that you understand before, but you haven't seen all the sentences that you understand before.

Para visualizar estas relaciones, podemos utilizar herramientas como el Embedding Projector de TensorFlow (disponible en <https://projector.tensorflow.org/>). Esta herramienta permite explorar interactivamente espacios de embeddings predefinidos, buscando palabras específicas y sus vecinos más cercanos. Por ejemplo, al buscar la palabra "action" en un espacio Word2Vec, encontraríamos vecinos semánticos como "movement", "activity", "reaction", "behavior" y "response", revelando cómo el modelo ha agrupado conceptos relacionados con la acción física y la respuesta. De manera similar, "explosion" podría estar cerca de términos como "blast", "detonation", "eruption", "catastrophic" e "impact", demostrando cómo los embeddings capturan campos semánticos de manera natural.

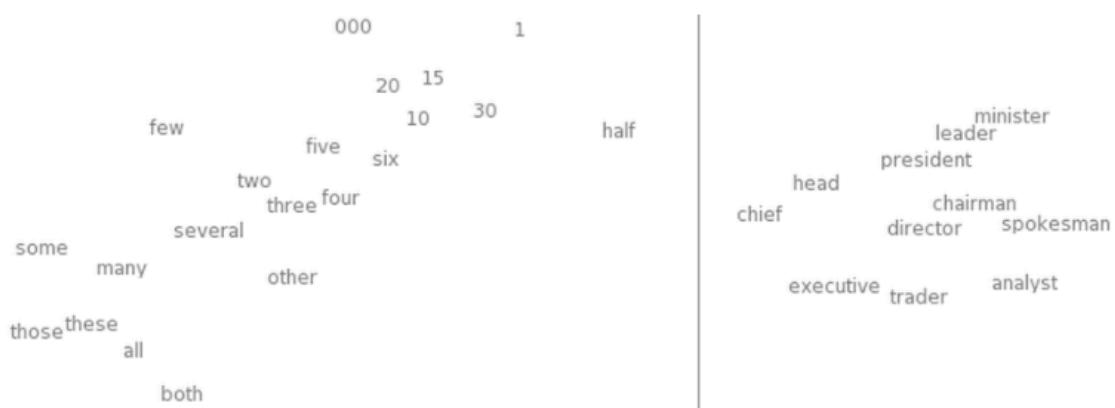
FRANCE	JESUS	XBOX	REDDISH	SCRATCHED	MEGABITS
AUSTRIA	GOD	AMIGA	GREENISH	NAILED	OCTETS
BELGIUM	SATI	PLAYSTATION	BLUISH	SMASHED	MB/S
GERMANY	CHRIST	MSX	PINKISH	PUNCHED	BIT/S
ITALY	SATAN	IPOD	PURPLISH	POPPED	BAUD
GREECE	KALI	SEGA	BROWNISH	CRIMPED	CARATS
SWEDEN	INDRA	PSNUMBER	GREYISH	SCRAPED	KBIT/S
NORWAY	VISHNU	HD	GRAYISH	SCREWED	MEGAHERTZ
EUROPE	ANANDA	DREAMCAST	WHITISH	SECTIONED	MEGAPIXELS
HUNGARY	PARVATI	GEFORCE	SILVERY	SLASHED	GBIT/S
SWITZERLAND	GRACE	CAPCOM	YELLOWISH	RIPPED	AMPERES

What words have embeddings closest to a given word?

From Collobert *et al.* (2011)

]

como el embedding es generalmente en espacios de muchas dimensiones, para tener una intuicion de la nube de puntos multidimensional, ya vimos que hay varias formas de hacerlo. PCA es una, que se centra en los ejes de inercia de la nube, desde la lupa de la fisica, o visto desde la estadistica como dispersion con respecto a la media lo que define la direccion, ambas coinciden. el metodo Fue ideado por bio matematicos, basado en la fisica de cuerpos de 3 dimensiones, la fisica que describe la distribucion de masa en cuerpos tridimensionales. Podemos usar algoritmos mas modernos, que serian como lupas muy potentes en una zona chica, muy local de la nube, y pierde vision periferica. t.-sne es uno, de la linea de pensamiento de machine learning que usan el concepto de grafos. entidades conectadas por lineas , mas o menos gruesas , o fuertes, segun la relacion entre esas entidades, o nodos. esto forma una red de nodos. Imaginan los datos en el espacio original conectados en una red, en un grafo. Se propone un grafo en 2 dimensiones, y se crea una funcion de costo, calculando la diferencia entre ambos grafos. se varia la posicion de los nodos del de baja dimensiones, y se calcula la distancia nuevamente, se mueven otro poco tal que disminuya la distancia, hasta que esta es minima. y llegamos a una interprtacion de los datos. Un proceso con el mismo espiritu del aprendisaje de una red neuronal.



t-SNE visualizations of word embeddings. Left: Number Region; Right: Jobs Region. From Turian *et al.* (2010), see complete image.

]

Para un chatbot o sistema de clasificación textual, esta propiedad es revolucionaria: ya no necesita reglas explícitas para saber que "filme" y "película" están relacionados, o que "adrenalina" y "acción" pertenecen a un campo semántico compartido. El sistema puede generalizar basándose en la posición relativa de las palabras en este espacio vectorial.

De Embeddings Estáticos a Contextuales

Los primeros modelos de embeddings como Word2Vec o GloVe generaban representaciones "estáticas" donde cada palabra recibía siempre el mismo vector, independientemente del contexto. Esto presentaba un problema fundamental con la polisemia: la palabra "banco" recibía un único vector que debía representar simultáneamente tanto a una institución financiera como a un asiento en el parque.

Un avance significativo llegó con modelos como ELMo y posteriormente con arquitecturas basadas en Transformers como BERT (lo veremos mas adelante). Estos modelos generan embeddings "contextuales" donde la representación de una palabra varía según el contexto en que aparece. Así, "banco" recibe un vector diferente en "deposité dinero en el banco" que en "me senté en un banco del parque".

Este nuevo upgrade revolucionario fue posible gracias a arquitecturas neuronales que procesaban oraciones completas, considerando las interrelaciones entre todas las palabras. En particular, los mecanismos de atención permitieron a estos modelos "enfocar" diferentes partes del contexto para desambiguar el significado de cada palabra.

Para un traductor o lingüista, esta evolución hacia representaciones contextuales resulta natural: sabemos que el significado de las palabras no existe aisladamente, sino que emerge de su uso en contexto. Los embeddings contextuales modelan computacionalmente esta realidad lingüística fundamental.

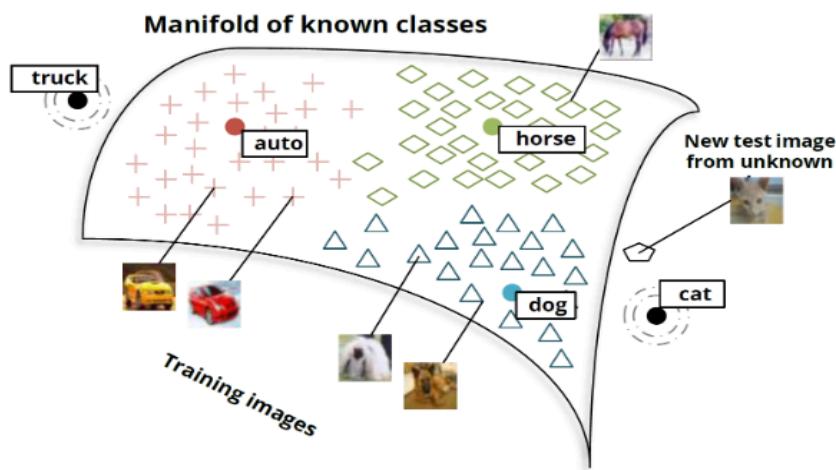
Representaciones Compartidas

Un desarrollo especialmente relevante para traductores y filólogos comparativos son los embeddings multilingües o cross-lingüísticos. Estos modelos ubican palabras de diferentes idiomas en un mismo espacio vectorial, de manera que traducciones equivalentes aparecen próximas entre sí.

Lo sorprendente de estos modelos es que, al alinear algunas traducciones conocidas (como "dog"/"perro" o "house"/"casa"), el sistema puede inferir traducciones para palabras que nunca se le enseñaron explícitamente. Esto ocurre porque si las palabras "dog" y "cat" están cerca en inglés, y "dog" se alinea con "perro", el modelo naturalmente ubicará "cat" cerca de "gato".

Este fenómeno sugiere que las lenguas, a pesar de sus diferencias superficiales, comparten estructuras semánticas profundas que los modelos computacionales pueden identificar. Las "formas" de distintos idiomas en el espacio semántico parecen ser topológicamente similares, de modo que al alinear algunos puntos clave, otras relaciones se alinean automáticamente.

Esta capacidad se extiende incluso más allá del lenguaje. Modelos multimodales pueden proyectar imágenes y texto en un espacio vectorial compartido, permitiendo búsquedas semánticas entre modalidades. Por ejemplo, podríamos buscar imágenes que correspondan conceptualmente a la frase "un animal corriendo en un prado", sin necesidad de que la imagen tenga etiquetas textuales específicas.



(Socher et al. (2013b))

]

Aplicaciones

Estos avances tienen profundas implicaciones para disciplinas como la traducción, la filología, la lingüística y las ciencias sociales.

Para traductores:

Los sistemas actuales de traducción automática no operan con reglas rígidas o traducciones palabra por palabra, sino que trabajan con representaciones semánticas profundas que capturan matices culturales y contextuales. Esta evolución explica por qué los sistemas actuales cometen muchos menos errores relacionados con la polisemia o la ambigüedad contextual.

Lejos de hacer obsoleta la traducción humana, estas herramientas están redefiniendo el papel del traductor como un mediador cultural y editor especializado que supervisa, adapta y refina el trabajo preliminar de los sistemas automáticos. Comprender cómo funcionan estos modelos permite a los traductores utilizarlos más eficazmente, identificando sus fortalezas y limitaciones en diferentes tipos de texto.

Para filólogos y lingüistas:

Estos modelos ofrecen nuevas herramientas para estudiar fenómenos lingüísticos a escala sin precedentes. Los investigadores pueden analizar cómo los significados han evolucionado a lo largo del tiempo entrenando modelos en textos de diferentes épocas [ver paper sobre el rol de la mujer a lo largo del siglo XX a través de embeddings](#), explorar diferencias semánticas sutiles entre idiomas mediante embeddings multilingües, o estudiar la adquisición del lenguaje comparando el aprendizaje humano con la evolución de las representaciones en los modelos durante el entrenamiento.

La capacidad de estos sistemas para capturar relaciones semánticas complejas sin instrucción explícita plantea fascinantes preguntas teóricas sobre la naturaleza distribucional del significado lingüístico y cómo las estructuras semánticas emergen de patrones estadísticos en el uso del lenguaje.

Para las ciencias sociales:

Los modelos de embeddings permiten analizar grandes corpus textuales para identificar patrones discursivos, cambios en percepción social, o sesgos en la comunicación. Los investigadores pueden detectar cómo cambian las asociaciones entre conceptos a lo largo del tiempo en periódicos o literatura, analizar diferencias en la representación de distintos grupos sociales en los medios, o estudiar la difusión de narrativas o marcos interpretativos en redes sociales.

Estas herramientas permiten un análisis cuantitativo y cualitativo del discurso a escalas previamente imposibles, ofreciendo nuevas perspectivas sobre fenómenos sociales y culturales reflejados en el lenguaje.

Ejercicios:

Utilizando la herramienta interactiva [Embedding Projector](#), realiza los siguientes ejercicios para explorar cómo funcionan las distancias euclidianas y coseno en la práctica.

Ejercicio 1:

- Busca en la herramienta palabras relacionadas con "king" (rey) y "queen" (reina).
- Identifica visualmente cuál es la palabra más cercana a "queen". ¿Se corresponde con lo que esperarías según el significado de las palabras?

Ejercicio 2: distancias euclidianas

- Selecciona las palabras "cat" y "dog". Anota sus coordenadas aproximadas (las primeras tres dimensiones son suficientes).
- Calcula manualmente la distancia euclíadiana entre ambas palabras utilizando la fórmula:

Distancia euclíadiana entre dos vectores (A, B) en 3 dimensiones:

$$\text{distancia euclíadiana} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}$$

Analogía semántica con embeddings:

$$\text{Vector}("king") - \text{Vector}("man") + \text{Vector}("woman") \approx \text{Vector}("queen")$$

- Compara esta distancia con la distancia entre "cat" y "car". ¿Cuál es menor? ¿Qué nos dice esto sobre el significado relativo de estas palabras?

Ejercicio 3: Midiendo similitudes mediante distancia coseno

- Nuevamente, usando las coordenadas aproximadas de las palabras "apple" (manzana), "banana" y "computer" (computadora), calcula manualmente la similitud coseno entre "apple" y "banana", y luego entre "apple" y "computer". Usa la fórmula siguiente para la similitud coseno entre dos vectores AA y BB:

Similitud coseno entre dos vectores $A = (x_1, y_1, z_1)$ y $B = (x_2, y_2, z_2)$:

$$\text{similitud coseno} = \frac{A \cdot B}{\|A\| \|B\|} = \frac{x_1 x_2 + y_1 y_2 + z_1 z_2}{\sqrt{x_1^2 + y_1^2 + z_1^2} \sqrt{x_2^2 + y_2^2 + z_2^2}}$$

- ¿Cuál par de palabras tiene mayor similitud? ¿Por qué crees que sucede esto?

Ejercicio 4: Analizando sinónimos

- Elige una palabra como "happy" (feliz) y encuentra tres palabras cercanas en el espacio de embeddings.
- Calcula las distancias euclíadianas y coseno entre "happy" y cada una de estas palabras cercanas.
- ¿Las palabras más cercanas según distancia euclíadiana coinciden con aquellas según similitud coseno? ¿Hay alguna diferencia notable?

Ejercicio 5: Analogías semánticas

- Intenta la analogía clásica: "king" - "man" + "woman". Visualmente, busca la palabra más cercana al resultado de esta operación en el embedding projector.
- Calcula las distancias (euclíadiana y coseno) entre tu resultado y las palabras candidatas cercanas. ¿La palabra "queen" es realmente la más cercana en ambos tipos de distancia? ¿Hay diferencias?

Ejercicio 6: Explorando campos semánticos

- Selecciona un tema como "music" (música) o "science" (ciencia). Identifica visualmente al menos cinco palabras relacionadas.
- Calcula las distancias euclíadianas y coseno entre estas palabras para formar una tabla comparativa.
- ¿Existe una agrupación clara por temática según estas medidas? ¿Qué conclusiones puedes extraer?