uc3m | Universidad Carlos III de Madrid

OpenCourseWare

Procesamiento de Lenguaje Natural con Aprendizaje Profundo,

Máster en Ciencia y Tecnología Informática

Tema 4: Word Embeddings

Isabel Segura Bedmar

Objetivos

- Conocer el concepto de word embedding y comprender su utilidad en las redes neuronales.
- Conocer las arquitecturas básicas de Word2Vec.
- Entrenar modelos de word-embeddings.
- Utilizar modelos pre-entrenados de word embeddings en la inicialización de redes profundas (tales como CNN o RNN) aplicadas a tareas de PLN.

Índice

- Word Embeddings
- Por qué usar word embeddings
- Word2Vec
- Limitaciones de los word embeddings

El pasado…

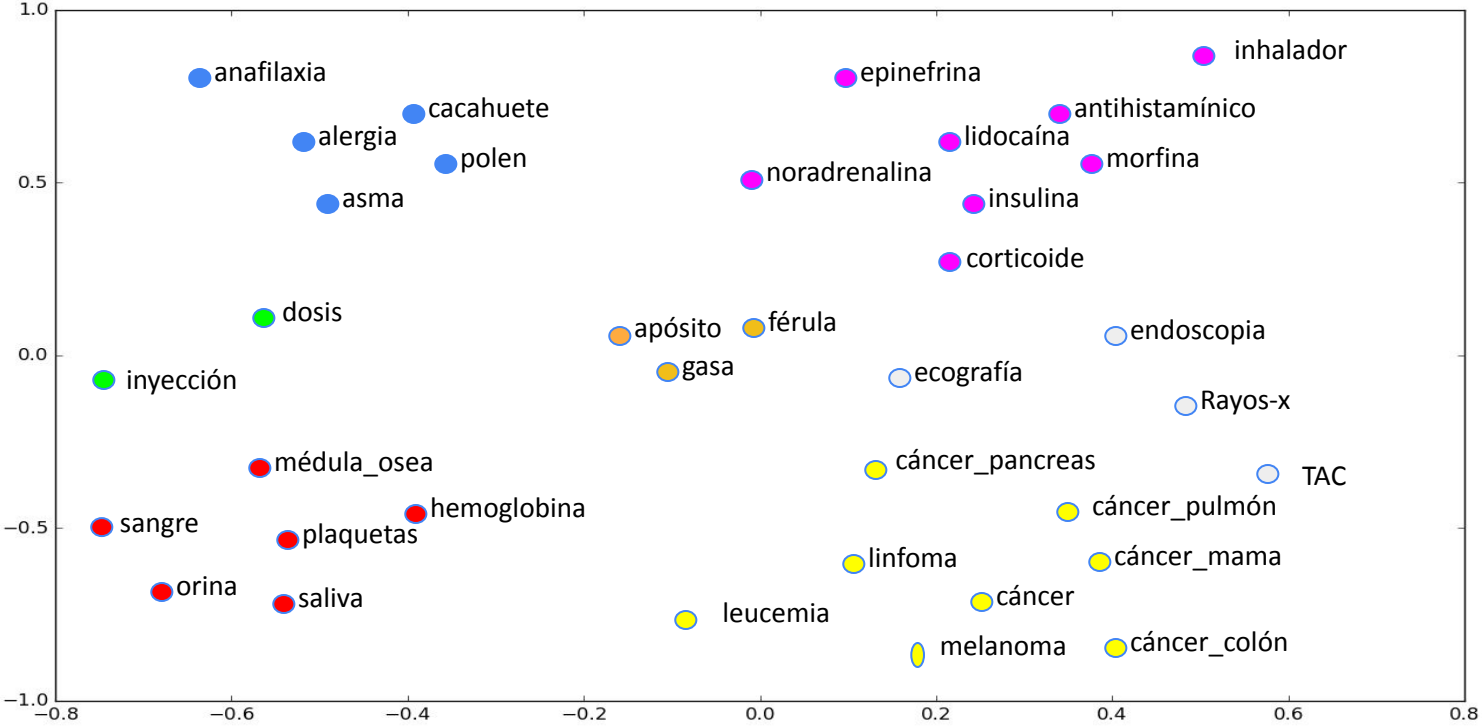
- **BoW** y **TF-IDF** son modelos eficaces y eficientes en la **representación de textos**.
- Sin embargo, **no capturan información semántica**:
  - "*edema de glotis*" e "*hinchazón de la laringe*" son sinónimos, pero tendrán vectores (en BoW o en tf-idf) completamente distintos.
- Investigación en PLN dirigida a desarrollar nuevos métodos que puedan representar información semántica.
- A partir de 2013, con el renacimiento de las redes neuronales artificiales, se han propuesto **nuevos enfoques** (Word2Vec, Glove, fastText, etc) capaces de aprender representaciones de palabras (**word embeddings**) en un espacio vectorial y capturar sus relaciones semánticas.

¿Qué es un modelo de word embeddings?

- Un **modelo de word embeddings** está formado por un **vocabulario** (conjunto de palabras distintas en una colección de textos) y sus respectivos **embeddings** (vectores).
- Es decir, cada **palabra** es representada por un **vector** de números reales, capaz de **codificar** el **significado** de la palabra y **capturar** sus **relaciones semánticas** con otras palabras del vocabulario.
- Necesita ser **entrenado** a partir de una **colección de textos** lo suficientemente **grande** para poder aprender buenas representaciones de las palabras.
    - Un ejemplo podría ser la colección de textos de Wikipedia en inglés (mil millones de palabras distintas).
- El uso de word embeddings muestra **buenos resultados** en muchas **aplicaciones PLN**.

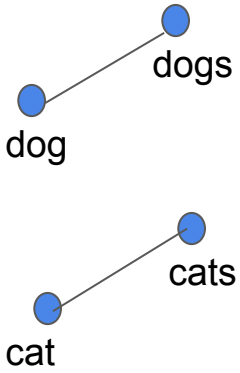
Word embeddings

- **Vectores cercanos** posiblemente representen palabras que son **sinónimos** o tienen algún tipo de **relación semántica**.



Word embeddings

- El estudio empírico* de algunos modelos (por ejemplo, el modelo Word2Vec entrenado con Google News) ha mostrado que no únicamente son capaces de capturar las relaciones de sinonimia, sino también otras relaciones.

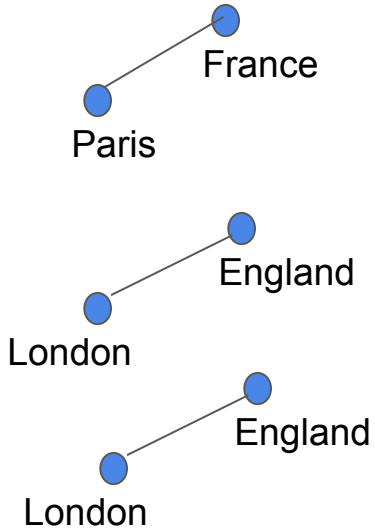


Singular-Plural de animales domésticos, misma distancia

*Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. In Proceedings of NIPS, 2013.

Word embeddings

- El estudio empírico* de algunos modelos (por ejemplo, el modelo Word2Vec entrenado con Google News) ha mostrado que no únicamente son capaces de capturar las relaciones de sinonimia, sino también otras relaciones.

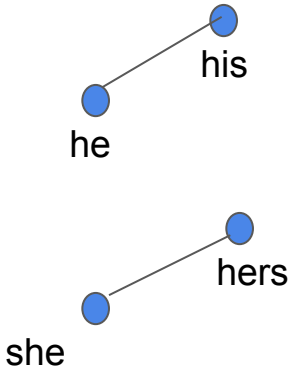


Capitales y Países misma distancia.

*Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. In Proceedings of NIPS, 2013.

Word embeddings

- El estudio empírico* de algunos modelos (por ejemplo, el modelo Word2Vec entrenado con Google News) ha mostrado que no únicamente son capaces de capturar las relaciones de sinonimia, sino también otras relaciones.

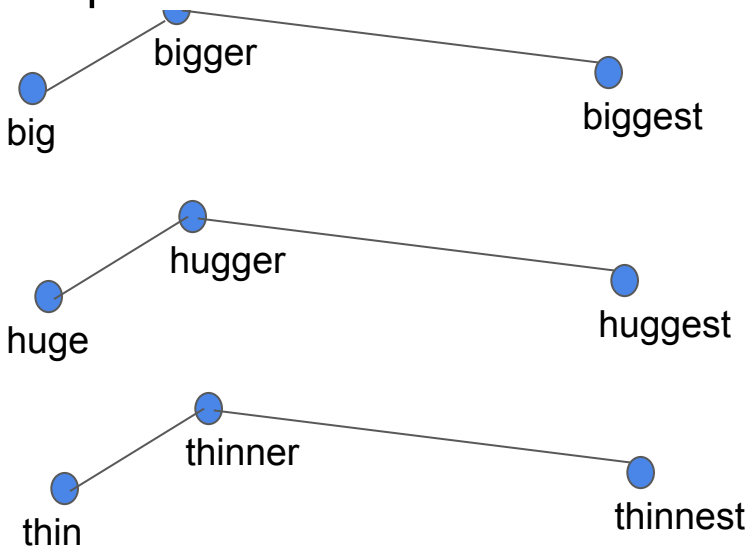


Misma distancia entre pronombre personal y posesivo

*Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. In Proceedings of NIPS, 2013.

Word embeddings

- El estudio empírico* de algunos modelos (por ejemplo, el modelo Word2Vec entrenado con Google News) ha mostrado que no únicamente son capaces de capturar las relaciones de sinonimia, sino también otras relaciones.



Misma distancia en adjetivos, comparativos y superlativos.

*Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. In Proceedings of NIPS, 2013.

Índice

- Word Embeddings
- **Por qué usar word embeddings**
- Word2Vec
- Limitaciones de los word embeddings

¿Por qué usar word embeddings?

- Al ser capaces de capturar información semántica, son un **enfoque apropiado** para la **representación de textos** como **entrada** a un algoritmos de **aprendizaje automático**. En particular, para la **inicialización** de las **redes neuronales**.
- En muchos casos, el uso de **word embeddings** en lugar de inicialización aleatoria consigue **mejores resultados** y **reduce** los tiempos de **entrenamiento**.

¿Por qué usar word embeddings?

- También incorporan **conocimiento externo** del mundo, que no tiene porque estar presente en el dataset de entrenamiento de la red. Por ejemplo:
  - Por ejemplo, tenemos una red para la tarea de **reconocimiento de entidades** en el dominio clínico (identificar nombres de enfermedades, medicamentos, etc).
  - Dicha red está siendo **entrenada** con un dataset donde la palabra **‘*aspirin*’** está **anotada** como **‘DRUG’**, pero no contiene a la palabra ‘*“ibuprofen”.*

¿Por qué usar word embeddings?

- En un modelo de word embeddgins que haya sido entrenado con textos clínicos, posiblemente las palabras '*aspirin*' y *'ibuprofen'* tengan **vectores cercanos** porque aparecen en oraciones similares:
  - ***Aspirin is a** nonsteroidal anti-inflammatory drug (NSAID)*
  - ***Ibuprofen**, a Nonsteroidal anti-inflammatory Drug, …*
  - ***Ibuprofen** is an anti inflammation medicine (a non steroidal anti inflammatory drug or NSAID).*
- Por tanto, si la red es inicializada con este modelo de word embeddings, es probable que pueda **inferir** que **'aspirin'** y **'ibuprofen'** están **relacionados**.
- Gracia a esto, la **red** podría identificar **'ibuprofen'** como **'DRUG'**, **aunque no esté** presente en el **dataset** de **entrenamiento**.

Índice

- Word Embeddings
- Por qué usar word embeddings
- **Word2Vec**
- Limitaciones de los word embeddings

Word2Vec

- Es uno de los enfoques más sencillos y utilizados para aprender word embeddings a partir de una colección de textos (se necesitan millones de palabras!!!).
- Propuesto en 2013, por Mikolov et al [1], [2], [3].
- **Word2Vec** es una red muy simple con **una única capa oculta**.
- Su **entrada** es una colección de **oraciones**. Cada oración ha sido pre-procesada para eliminar signos de puntuación y números. También es posible aplicar **lematización** (para reducir el vocabulario).
- La **salida** de la red es un **diccionario** donde cada **palabra** del vocabulario (conjunto de palabras diferentes en la colección de oraciones) está **asociada** a un **vector**.
- Todos los **vectores** tienen la **misma dimensión**. La dimensión óptima se determina de forma empírica y depende del dataset.
- Al principio la red inicializará los vectores de forma aleatoria, que serán ajustados durante el entrenamiento.

[1] *Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. In Proceedings of Workshop at ICLR, 2013.*

[2] *Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. In Proceedings of NIPS, 2013.*

[3] *Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic Regularities in Continuous Space Word Representations. In Proceedings of NAACL HLT, 2013.*

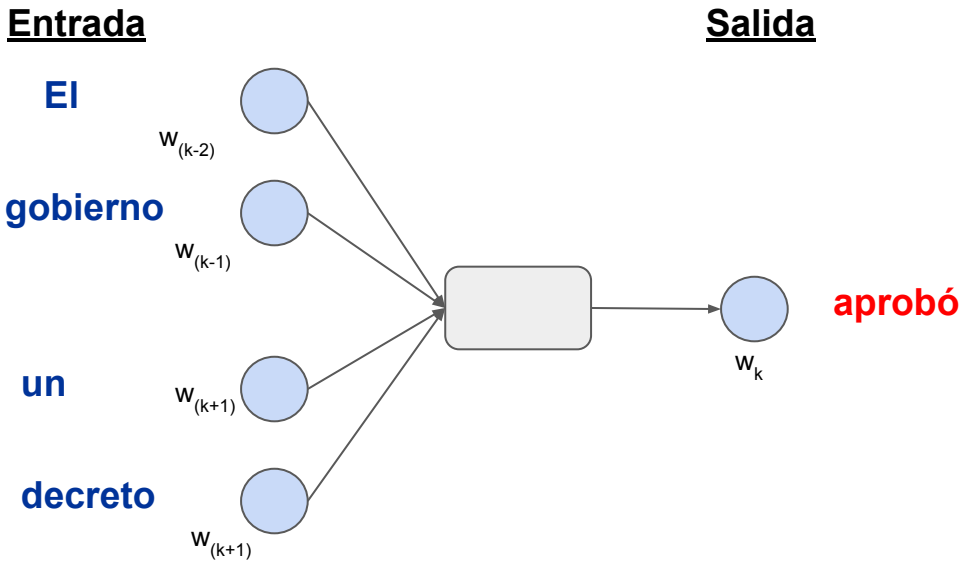
Word2Vec

- Propone dos arquitecturas (o estrategias) distintas para aprender los vectores:
  - **continuous bag-of-words** (CBOW): el objetivo de la red es **predecir una palabra a partir de su contexto**. Es decir, la red recibe una oración donde una de las palabras ha sido ocultada. Entonces debe aprender a inferir la palabra correcta. Para ello usará la información de las palabras del contexto.
  - **skip-grams**: el objetivo de la red e**s predecir el contexto más apropiado para una palabra de entrada**. Es decir, ahora la red recibe únicamente una palabra, y tiene que inferir las palabras que rodean a la palabra de entrada.

Word2Vec

- Una de las principales ventajas de **Word2Vec** es que **no requiere anotar un dataset de entrenamiento** (tarea muy costosa y que implica una gran cantidad de tiempo).
- En **CBOW**, será necesario tomar **contextos** de un determinado **tamaño** (por ejemplo, 5).
- Dado el contexto "*El gobierno **aprobó** un decreto*",
  - la entrada para CBOW serían las palabras: *El gobierno un decreto*,
  - *aprobó* será la salida esperada, es decir, la etiqueta para el contexto de entrada.

Word2Vec: CBOW



Word2Vec

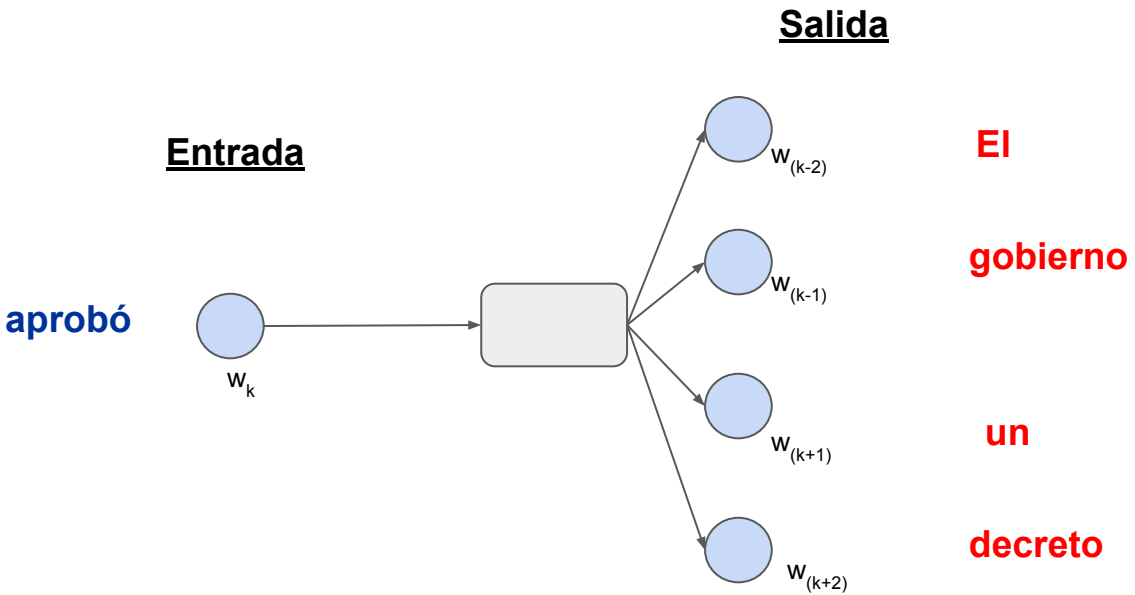
- Por el contrario, en el enfoque **skip-gram**, la entrada de la red será una única palabra, y la salida esperada, el contexto de palabras (de un determinado tamaño k) que rodean a la palabra de entrada.
- Tanto en CBOW como en skp-gram, cada oración puede producir varias instancias, siempre que el tamaño del contexto (k) sea menor que el número de palabras en la oración.

Word2Vec

- Por ejemplo, para CBOW, y k=3, algunos ejemplos de instancias son:

	Entrada	**Salida**
El banco de Inglaterra mantiene los tipos	*El de*	*banco*
El banco de Inglaterra mantiene los tipos	*banco Inglaterra*	*de*
El banco de Inglaterra mantiene los tipos	de mantiene	Inglaterra
El banco de Inglaterra mantiene los tipos	Inglaterra los	mantiene
El banco de Inglaterra mantiene los tipos	mantiene tipos	los

Word2Vec: skip-gram



Word2Vec

- CBOW es mucho más rápido que skip-gram, y también obtiene mejores resultados para palabras muy comunes.
- **skip-gram** puede aprender buenas representaciones con conjuntos de entrenamiento más pequeños que CBOW. Representa bien incluso palabras raras.
- Existen muchos modelos pre-entrenados sobre distintas colecciones, dominios e idiomas. Por ejemplo, visita el siguiente repositorio con más de 200 modelos.
- La librería spacy también incluye diferentes modelos de embeddings.
- La librería gensim también permite descargar modelos pre-entrenados para ser utilizados en la representación de los textos como entrada a cualquier algoritmo.

Índice

- Word Embeddings
- Por qué usar word embeddings
- Word2Vec
- **Limitaciones de los word embeddings**

Limitaciones de los modelos de word embeddings

- Para entrenar un modelo es necesario una gran cantidad de textos (> millón de palabras) y tiempo.
- No es posible representar sintagmas nominales o entidades multi-token, como por ejemplo “*Joe Biden*”, “*American Airlines*”, o “*Toyota Corolla GR Sport*”.
- Palabras como *‘bad’* y *‘good’*, aunque son antónimos, tiene vectores muy próximos (esto es porque suelen aparecer en contextos similares).
- Un único embedding por palabra. No es capaz de diferenciar palabras polisémicas (*mono: i) tipo de simio, ii) bonito, gracioso, iii) prenda de vestir)*.

Resumen

- Los word embeddings son un enfoque para representación de textos, capaz de capturar el significado y relaciones semánticas de cada palabra.
- Utilizar un modelo de word embeddings para inicializar los pesos de una red, suele obtener mejores resultados que la inicialización aleatoria.
- Utilizados en muchas aplicaciones de PLN con buenos resultados.
- Permiten integrar conocimiento externo que no está en el conjunto de entrenamiento.
- No son capaces de proporcionar vectores diferentes a significados diferentes de una misma palabra.

OpenCourseWare
Procesamiento de Lenguaje Natural con Aprendizaje Profundo,

Gracias!!!

https://github.com/isegura