Procesamiento de Lenguaje Natural

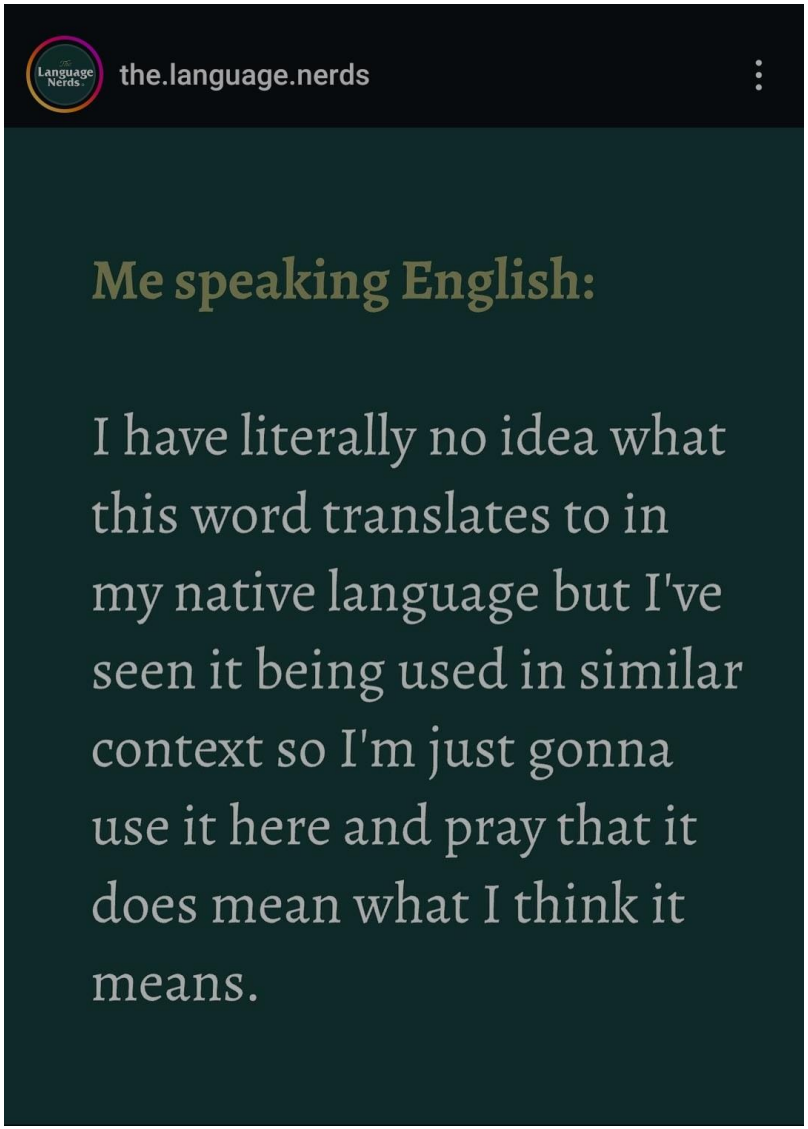
Vectores Semánticos

Mauricio Toledo-Acosta
mauricio.toledo@unison.mx

Departamento de Matemáticas
Universidad de Sonora

Section 1

Introducción


the.language.nerds
Me speaking English:
I have literally no idea what this word translates to in my native language but I've seen it being used in similar context so I'm just gonna use it here and pray that it does mean what I think it means.

Referencias

Chapter 6. Jurafsky, D., Martin, J. H. (2019). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*.

La Hipótesis Distribucional

Palabras que aparecen en contextos similares tienden a tener significados similares. Este vínculo entre la similitud en la distribución de las palabras y la similitud en su significado se denomina **hipótesis distribucional**.

Esta hipótesis fue formulada en los años 50 por lingüistas como Joos (1950), Harris (1954) y Firth (1957), que observaron que las palabras sinónimas (como oculista y oftamologo) tendían a aparecer en el mismo entorno (por ejemplo, cerca de palabras como ojo o examinado).

> *A word is characterized by the company it keeps*

Más información

Ejemplo

¿Qué es el **Ongchoi**?

Ejemplo

Veamos algunos contextos en los que ocurre **ongchoi**:

- **Ongchoi** is delicious sauteed with garlic.
- **Ongchoi** is superb over rice.
- ...**ongchoi** leaves with salty sauces...

Ahora, veamos algunos contextos de los contextos anteriores:

- ...spinach sauteed with garlic over rice...
- ...chard stems and leaves are delicious...
- ...collard greens and other salty leafy greens

Respuesta

La semántica vectorial

La **semántica vectorial** son modelos que buscan aprender representaciones del significado de las palabras directamente a partir de su distribución en los textos.

La idea de la semántica vectorial es representar una palabra como un punto en un espacio semántico multidimensional. Los vectores que representan palabras suelen denominarse **embeddings**, porque la palabra está incrustada en un espacio vectorial concreto.

La cercania de embeddings da cuenta de diversos fenómenos, además de la smilitud de palabras.

Ejemplo

¿Por qué es útil la semántica vectorial?

What's the meaning of life?
LIFE

Un buen modelo semántico debería decirnos que:

- Algunas palabras tienen significados similares (gato es similar a perro).
- Algunas palabras son sinónimas o antónimas (frío – caliente).
- Algunas palabras tienen connotaciones positivas (feliz) mientras que otras tienen connotaciones negativas (triste).
- Algunas palabras como comprar, vender y pagar ofrecen perspectivas diferentes sobre el mismo acontecimiento de compra subyacente (Si te compro algo, me lo has vendido, y te he pagado).

Similitud de Palabras

No todas las palabras tienen muchos sinónimos, sin embargo, la mayoría de ellas tienen muchas palabras similares (gato – perro). La noción de similitud entre palabras es muy útil en diversas tareas semánticas, por ejemplo, decidir si dos oraciones significan cosas parecidas.

- Podemos obtener las similitudes entre palabras de listas predefinidas (por ejemplo, SimLex-999).

vanish	disappear	9.8
behave	obey	7.3
belief	impression	5.95
muscle	bone	3.65
modest	flexible	0.98
hole	agreement	0.3

- Podemos aprender las similitudes a partir de co-ocurrencias.

Modelos de semántica vectorial

- Módelos basados en conteos: El significado de una palabra está dado en términos de ocurrencias en documentos.
  - Bag of Words (BOW)
  - Term Frequency - Inverse Document Frequency (TF-IDF)
- Módelos basados en redes neuronales:
  - Clásicos: `Word2Vec`, `GloVe`, ...
  - LLMs: `GPT`, `LLaMA`, `Jamba`, `Gemma`, ...

Modelos de semántica vectorial

- Módelos basados en conteos:
  - + Modelo sencillo y simple
  - + Interpretabilidad
  - - Vectores ralos (*sparse*)
  - - Alta dimensionalidad (del orden de miles o más)
- Módelos basados en redes neuronales:
  - + Vectores densos
  - + Menor dimensionalidad (del orden de cientos)
  - - Algunos pueden ser computacionalmente caros de obtener

¿Cómo medimos la similitud/distancia entre vectores?

Consideremos dos embeddings $u, v \in \mathbb{R}^D$.

- La **similitud coseno** es un valor entre -1 y 1 y está dada por

$$\text{sim}(u, v) = \frac{u \cdot v}{\|u\|\|v\|} = \cos(u0v)$$

  donde $u \cdot v$ es el producto punto y $\|u\|$ es la norma del vector.

- La distancia angular es un valor entre 0 y π dado por

$$d_\theta(u, v) = \arccos(\text{sim}(u, v))$$

En general, no usamos la distancia Euclidiana.

Ejemplo

La matriz term-document

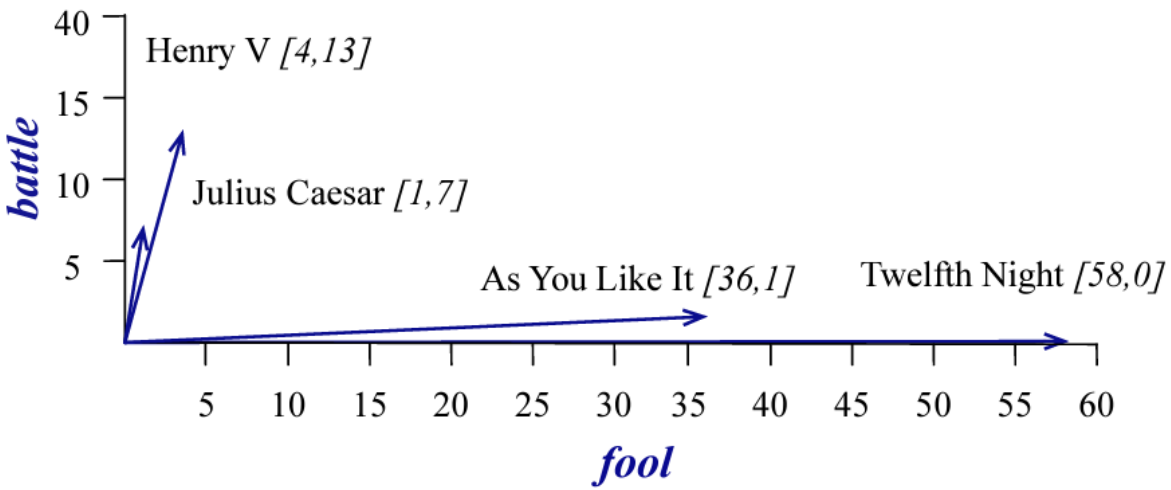
En una matriz término-documento, cada fila representa una palabra del vocabulario y cada columna representa un documento de alguna colección de documentos.

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	0	7	13
good	114	80	62	89
fool	36	58	1	4
wit	20	15	2	3

Modelo BOW

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	0	7	13
good	114	80	62	89
fool	36	58	1	4
wit	20	15	2	3

El modelo BOW (bag of words) asigna a cada documento el vector correspondiente a la columna. El vector de cada palabra es su fila.



Preguntas

- ¿Qué obtenemos si sumamos las filas?
- ¿Qué obtenemos si sumamos las columnas?
- ¿Qué tamaño tiene la matriz anterior?
- ¿Qué palabras tenderán a dominar la matriz?

Las matrices término-documento se definieron originalmente como un medio de encontrar documentos similares para la tarea de recuperación de información documental. Dos documentos que son similares tenderán a tener palabras similares, y si dos documentos tienen palabras similares sus vectores columna tenderán a ser similares.

¿Por qué la similitud coseno (o métrica angular) captura mejor la similitud?

TF-IDF

1. The first is the **term frequency** (Luhn, 1957): the frequency of the word in the document. Normally we want to downweight the raw frequency a bit, since a word appearing 100 times in a document doesn't make that word 100 times more likely to be relevant to the meaning of the document. So we generally use the $\log_{10}$ of the frequency, resulting in the following definition for the term frequency weight:

$$\mathrm{tf}_{t,d} = \begin{cases} 1 + \log_{10} \mathrm{count}(t,d) & \text{if } \mathrm{count}(t,d) > 0 \\ 0 & \text{otherwise} \end{cases}$$

Thus terms which occur 10 times in a document would have a tf=2, 100 times in a document tf=3, 1000 times tf=4, and so on.

TF-IDF

2. The second factor is used to give a higher weight to words that occur only in a few documents. Terms that are limited to a few documents are useful for discriminating those documents from the rest of the collection; terms that occur frequently across the entire collection aren't as helpful. The **document frequency** df_t of a term t is simply the number of documents it occurs in. By contrast, the **collection frequency** of a term is the total number of times the word appears in the whole collection in any document. Consider in the collection Shakespeare's 37 plays the two words *Romeo* and *action*. The words have identical collection frequencies of 113 (they both occur 113 times in all the plays) but very different document frequencies, since Romeo only occurs in a single play. If our goal is find documents about the romantic tribulations of Romeo, the word *Romeo* should be highly weighted:

$$\text{idf}_t = \log_{10}\left(\frac{N}{\text{df}_t}\right)$$