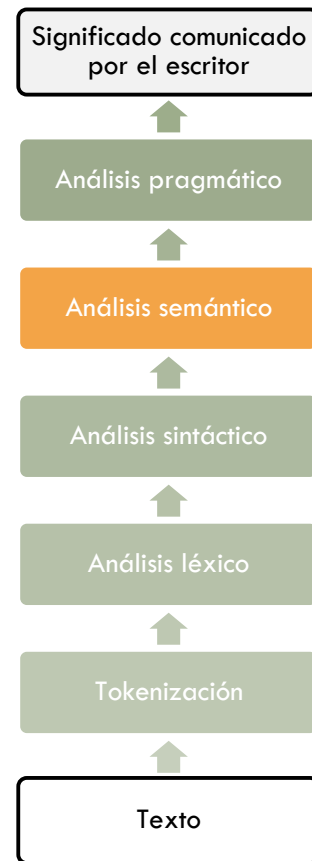


# Análisis de textos

## Análisis semántico

# Análisis semántico

- Analizar el texto para entender su **significado**
- Implica decidir qué significado expresa cada palabra, de entre todas sus posibles acepciones
  - Problema naturalmente ambiguo en el caso de palabras polisémicas
- Entradas
  - Una frase, con cada uno de sus tokens con Part of Speech y su árbol sintáctico
- Salidas
  - Para cada palabra, qué **acepción** de diccionario o a qué **sentido** o **clase de sentidos** refiere la palabra
  - Alternativamente, para cada documento, qué **temas** se tratan en el mismo o a qué otros documentos se parece desde el punto de vista del significado.



# Ejemplo de ambigüedad semántica

## ➤ **banco.**

- (Del fr. ant. *bank*, y este del germ. *\*banki*).
- **1. m.** Asiento, con respaldo o sin él, en que pueden sentarse varias personas.
- **2. m.** Madero grueso escuadrado que se coloca horizontalmente sobre cuatro pies y sirve como de mesa para muchas labores de los carpinteros, cerrajeros, herradores y otros artesanos.
- **3. m.** cama (ll del freno). U. m. en pl.
- **4. m.** En los mares, ríos y lagos navegables, bajo que se prolonga en una gran extensión.
- **5. m.** Conjunto de peces que van juntos en gran número.
- **6. m.** Establecimiento público de crédito, constituido en sociedad por acciones.
- **7. m.** Establecimiento médico donde se conservan y almacenan órganos, tejidos o líquidos fisiológicos humanos para cubrir necesidades quirúrgicas, de investigación, etc. *Banco de ojos, de sangre*
- **8. m.** Arq. sotabanco (ll piso habitable).
- **9. m.** Geol. Estrato de gran espesor.
- **10. m.** Ingen. Macizo de mineral que presenta dos caras descubiertas, una horizontal superior y otra vertical.
- **11. m.** Ven. Extensión de terreno con vegetación arbórea que sobresale en la llanura.
- **12. m. p. us.** Persona que cambia moneda.

# Aproximación clásica a la semántica

- En la historia de la lingüística existen diversas escuelas que han tratado de dar con una formalización de la semántica para permitir su estudio disciplinado
  - En general, tratan de construir un metalenguaje con el que expresar formalmente el significado
- Propuestas como
  - Utilizar **lógica de primer orden** para interpretar los textos y hacer inferencia de conocimientos
    - Sócrates es un hombre  $\rightarrow \text{Socrates IS\_A <HOMBRE>}$
    - Todos los hombres son mortales  $\rightarrow \forall x \text{ IS\_A <HOMBRE>, } x \text{ IS\_A <MORTAL>}$ 
      - Se deduce que:  $\text{Socrates IS\_A <HOMBRE>} \rightarrow \text{Socrates IS\_A <MORTAL>}$
  - Expresar los posibles significados de una palabra en **lenguaje sencillo**
    - Ej: río, en su sentido de elemento geográfico
      - Un tipo de lugar
      - Hay mucha agua en lugares como este
      - Los lugares de este tipo son alargados
      - Los lugares de este tipo son lugares grandes
      - El agua en sitios de este tipo siempre se mueve
- Ninguna de las aproximaciones puede considerarse aún un estándar firme
  - Es un campo menos explorado en NLP, comparado con la morfología y la sintáctica
- En la práctica las **aproximaciones estadísticas** pueden aportar valor

# Recursos lingüísticos de semántica: ontologías

- Uno de los formalismos más extendidos para representar conceptos es el uso de **ontologías**
  - Ontología: grafo de objetos (nodos), que pueden tener una serie de atributos, y las relaciones entre los objetos (aristas)
- Como aristas se suelen utilizar relaciones semánticas entre conceptos
  - **Sinonimia**: igual significado
    - Ejemplo: rápido – veloz
  - **Oposición**: antonimia, incompatibilidad, complementariedad, ...
    - Ej: rápido – lento
  - **Hiponimia**: cuando un significado (hipónimo) es un caso particular de otro (hiperónimo)
    - Ej: gato – animal
  - **Meronomia**: el significado al que refiere es parte del significado al que refiere otra cosa
    - Ej: mano – cuerpo
  - **Troponimia**: el significado de una acción es una forma específica de realizar otra acción
    - Ej: deambular – andar

# Wordnet

- Ontología de conceptos semánticos realizada manualmente por expertos
- Cada nodo del grafo es un **synset**: conjunto de palabras que son sinónimas en alguno de sus significados
  - El sentido del synset queda definido por la unión de palabras que lo componen
  - Las palabras polisémicas aparecen en tantos synsets como significados tengan
- Los synset se relacionan entre sí por relaciones de hiponimia, meronimia y troponimia
- <https://wordnet.princeton.edu/>

# Wordnet: synsets de “dog”

## Noun

- **S: (n) dog, domestic dog, [Canis familiaris](#)** (a member of the genus *Canis* (probably descended from the common wolf) that has been domesticated by man since prehistoric times; occurs in many breeds) *"the dog barked all night"*
- **S: (n) frump, dog** (a dull unattractive unpleasant girl or woman) *"she got a reputation as a frump"; "she's a real dog"*
- **S: (n) dog** (informal term for a man) *"you lucky dog"*
- **S: (n) cad, bounder, blackguard, dog, hound, heel** (someone who is morally reprehensible) *"you dirty dog"*
- **S: (n) frank, frankfurter, hotdog, hot dog, dog, wiener, wienerwurst, weenie** (a smooth-textured sausage of minced beef or pork usually smoked; often served on a bread roll)
- **S: (n) pawl, detent, click, dog** (a hinged catch that fits into a notch of a ratchet to move a wheel forward or prevent it from moving backward)
- **S: (n) andiron, firedog, dog, dog-iron** (metal supports for logs in a fireplace) *"the andirons were too hot to touch"*

## Verb

- **S: (v) chase, chase after, trail, tail, tag, give chase, dog, go after, track** (go after with the intent to catch) *"The policeman chased the mugger down the alley"; "the dog chased the rabbit"*

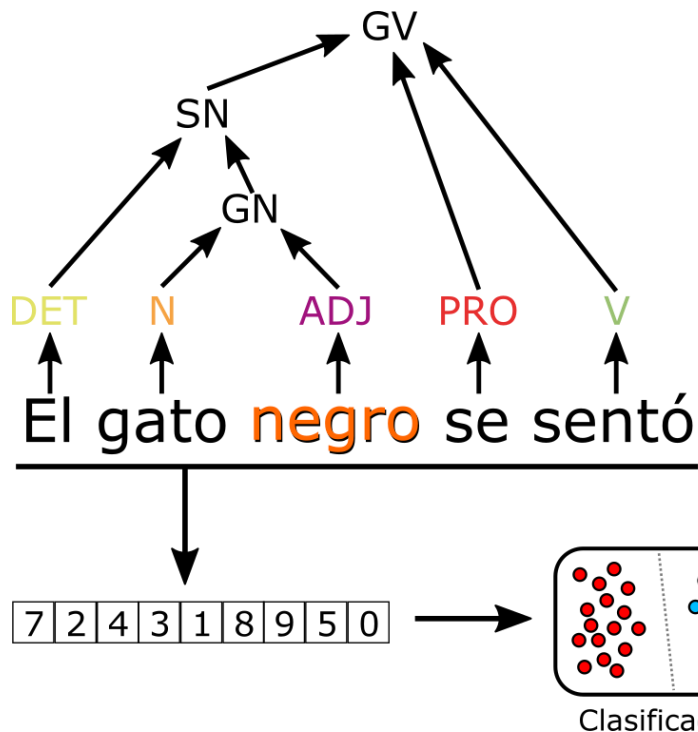
# Wordnet: hipónimos e hiperónimos de un synset “dog”

- **S: (n) dog, domestic dog, Canis familiaris** (a member of the genus Canis (probably descended from the common wolf) that has been domesticated by man since prehistoric times; occurs in many breeds) *“the dog barked all night”*
  - **direct hyponym / full hyponym**
    - **S: (n) puppy** (a young dog)
    - **S: (n) pooch, doggie, doggy, barker, bow-wow** (informal terms for dogs)
  - **S: (n) cur, mongrel, mutt** (an inferior dog or one of mixed breed)
    - **S: (n) feist, fice** (a nervous belligerent little mongrel dog)
    - **S: (n) pariah dog, pye-dog, pie-dog** (ownerless half-wild mongrel dog common around Asian villages especially India)
  - **S: (n) lapdog** (a dog small and tame enough to be held in the lap)
  - **S: (n) toy dog, toy** (any of several breeds of very small dogs kept purely as pets)
    - **S: (n) Chihuahua** (an old breed of tiny short-haired dog with protruding eyes from Mexico held to antedate Aztec civilization)
    - **S: (n) Japanese spaniel** (breed of toy dogs originating in Japan having a silky black-and-white or red-and-white coat)
    - **S: (n) Maltese dog, Maltese terrier, Maltese** (breed of toy dogs having a long straight silky white coat)
    - **S: (n) Pekinese, Pekingese, Peke** (a Chinese breed of small short-legged dogs with a long silky coat and broad flat muzzle)
    - **S: (n) Shih-Tzu** (a Chinese breed of small dog similar to a Pekingese)
    - **S: (n) toy spaniel** (a very small spaniel)
      - **S: (n) English toy spaniel** (British breed having a long silky coat and rounded head with a short upturned muzzle)
        - **S: (n) Blenheim spaniel** (red-and-white variety of English toy spaniel)
      - **S: (n) King Charles spaniel** (a toy English spaniel with a black-and-tan coat; named after Charles II who popularized it)
      - **S: (n) papillon** (small slender toy spaniel with erect ears and a black-spotted brown to white coat)
    - **S: (n) toy terrier** (a small active dog)
  - **S: (n) hunting dog** (a dog used in hunting game)
    - **S: (n) courser** (a dog trained for coursing)
    - **S: (n) Rhodesian ridgeback** (a powerful short-haired African hunting dog having a crest of reversed hair along the spine)
    - **S: (n) hound, hound dog** (any of several breeds of dog used for

- **S: (n) dog, domestic dog, Canis familiaris** (a member of the genus Canis (probably descended from the common wolf) that has been domesticated by man since prehistoric times; occurs in many breeds) *“the dog barked all night”*
  - **direct hyponym / full hyponym**
  - **part meronym**
  - **member holonym**
  - **direct hypernym / inherited hypernym / sister term**
    - **S: (n) canine, canid** (any of various fissiped mammals with nonretractile claws and typically long muzzles)
      - **S: (n) carnivore** (a terrestrial or aquatic flesh-eating mammal) *“terrestrial carnivores have four or five clawed digits on each limb”*
        - **S: (n) placental, placental mammal, eutherian, eutherian mammal** (mammals having a placenta; all mammals except monotremes and marsupials)
          - **S: (n) mammal, mammalian** (any warm-blooded vertebrate having the skin more or less covered with hair; young are born alive except for the small subclass of monotremes and nourished with milk)
            - **S: (n) vertebrate, craniate** (animals having a bony or cartilaginous skeleton with a segmented spinal column and a large brain enclosed in a skull or cranium)
              - **S: (n) chordate** (any animal of the phylum Chordata having a notochord or spinal column)
                - **S: (n) animal, animate being, beast, brute, creature, fauna** (a living organism characterized by voluntary movement)
                  - **S: (n) organism, being** (a living thing that has (or can develop) the ability to act or function independently)
                    - **S: (n) living thing, animate thing** (a living (or once living) entity)
                      - **S: (n) whole, unit** (an assemblage of parts)



# Desambiguación semántica supervisada



- Para cada palabra a analizar semánticamente, generar **características** en base a la morfología de esa palabra y sus vecinas, así como en base a la sintaxis de la frase en la que se encuentra
- Entrenar un **modelo de clasificación** utilizando un **corpus anotado semánticamente** con los sentidos de las palabras según diccionario u otra fuente de definiciones semánticas (ej. WordNet)
  - Los sentidos pueden estar anotados palabra por palabra (acepciones de diccionario) o según clases semánticas de una ontología (ej. <ANIMAL>, <CONSTRUCCIÓN>, <SENTIMIENTO>)
- Bag-of-words
- N-gramas de palabras
- N-gramas de palabras+POS
- Relación sintáctica de la palabra objetivo con sus vecinas
- ...

# Desambiguación semántica ligeramente supervisada

- Intentar aprender un modelo de desambiguación para cualquier palabra posible requiere de cantidades inmensas de texto anotado
  - WordNet tiene 117.659 synsets → ¡117.659 clases a predecir!
- Puede facilitarse el problema utilizando una ontología que contemple relaciones de **hiponimia** (como WordNet)
  - Para cada entidad de la ontología, extraer la etiqueta de una clase de alto nivel de la que sea hipónima, y que represente una **categoría semántica general**
    - Ej: gato → <ANIMAL>, taladro → <HERRAMIENTA>, triciclo → <VEHÍCULO>
  - Entrenar el clasificador para predecir estas etiquetas
  - Cuando se quiere desambiguar el significado de una palabra se usa el clasificador para predecir la etiqueta, y se escoge el significado más acorde con ella

# Named Entity Recognition

- Un caso particular de análisis semántico muy enfocado a la práctica es la **extracción de entidades**.
- Tratar de etiquetar cada palabra dentro de unas clases semánticas de utilidad práctica como: **PERSONA**, **LUGAR**, **ORGANIZACIÓN**, ...
  - Lista muy corta, entre 3 y 10 clases
- Útil para análisis de noticias, saber qué personas se relacionan con qué organizaciones, buscar menciones a los lugares donde ocurren los hechos, etc.

## Stanford Named Entity Tagger

Classifier:

Output Format:

Preserve Spacing:

Please enter your text here:

President of the United States Barack Obama announced significant changes on tax laws for tech companies. Silicon Valley giants such as Google and Microsoft have expressed their disagreement.

Enviar consulta

Clear

President of the **United States** **Barack Obama** announced significant changes on tax laws for tech companies. **Silicon Valley** giants such as **Google** and **Microsoft** have expressed their disagreement.

Potential tags:

**ORGANIZATION**

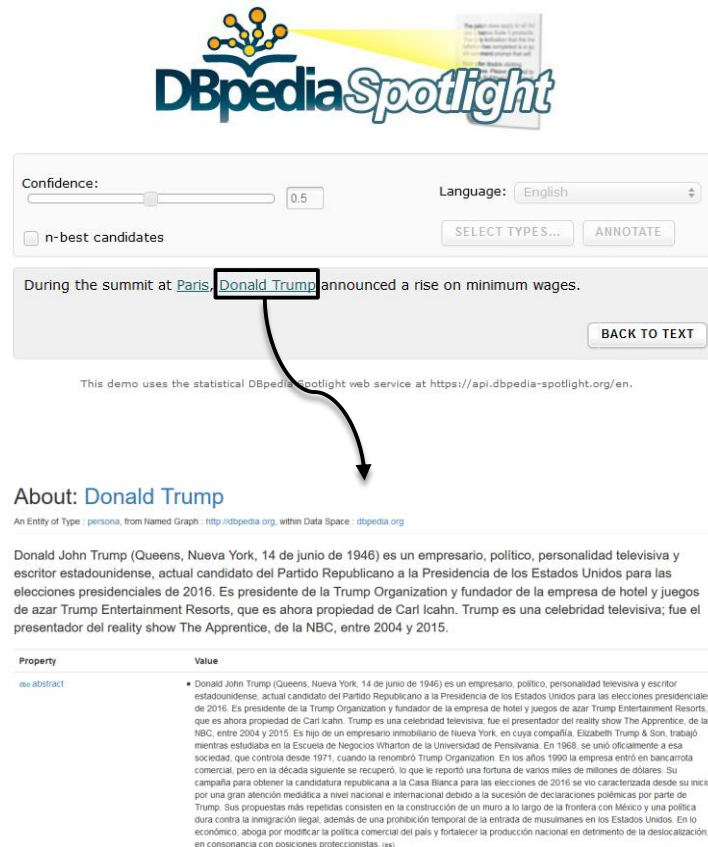
**LOCATION**

**PERSON**

Stanford Named Entity Tagger: <http://nlp.stanford.edu:8080/ner/process>

# Named Entity Linking

- Otra tarea de enriquecimiento tras el reconocimiento de entidades es el enlazado de entidades.
- Para cada entidad detectada, descubrir a qué entidad exacta se refiere dentro de una ontología como Wikipedia, DBPedia, ...
  - Si existen varias entidades que se llaman igual, se realiza una desambigüación comparando las palabras que aparecen en el contexto del texto y en el contexto de la página de Wikipedia, DBPedia, ...
- Se devuelve el texto con anotaciones en las que cada entidad encontrada lleva un enlace al correspondiente nodo de la ontología.



The screenshot shows the DBpedia Spotlight web interface. At the top, there is a logo for DBpedia Spotlight. Below the logo, there is a text input field containing the sentence: "During the summit at Paris, Donald Trump announced a rise on minimum wages." The entity "Donald Trump" is highlighted with a red box. A dropdown menu shows "Donald Trump" as the selected entity. Below the text, there is a "BACK TO TEXT" button. The interface also includes a "Confidence" slider set to 0.5, a "Language" dropdown set to "English", and buttons for "SELECT TYPES..." and "ANNOTATE". At the bottom, there is a section titled "About: Donald Trump" with a brief description and a table of properties and values.

This demo uses the statistical DBpedia Spotlight web service at <https://api.dbpedia-spotlight.org/en>.

About: [Donald Trump](#)

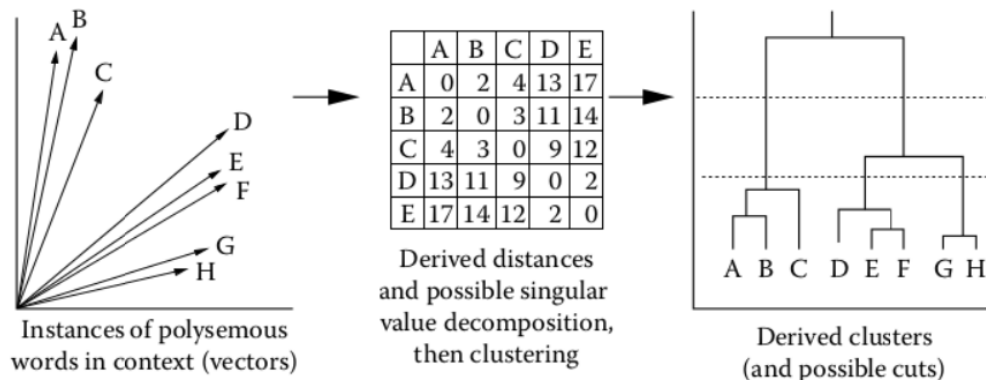
An Entity of Type: [persona](#), from Named Graph: [http://dbpedia.org](#), within Data Space: [dbpedia.org](#)

Donald John Trump (Queens, Nueva York, 14 de junio de 1946) es un empresario, político, personalidad televisiva y escritor estadounidense, actual candidato del Partido Republicano a la Presidencia de los Estados Unidos para las elecciones presidenciales de 2016. Es presidente de la Trump Organization y fundador de la empresa de hotel y juegos de azar Trump Entertainment Resorts, que es ahora propiedad de Carl Icahn. Trump es una celebridad televisiva, fue el presentador del reality show The Apprentice, de la NBC, entre 2004 y 2015. Es hijo de un empresario inmobiliario de Nueva York, en cuya compañía, Elizabeth Trump & Son, trabajó mientras estudiaba en la Escuela de Negocios Wharton de la Universidad de Pennsylvania. En 1968, se unió oficialmente a esa sociedad, que controla desde 1971, cuando la renombró Trump Organization. En los años 1990 la empresa entró en bancarrota comercial, pero en la década siguiente se recuperó, lo que le reportó una fortuna de varios miles de millones de dólares. Su campaña para obtener la candidatura republicana a la Casa Blanca para las elecciones de 2016 se vio caracterizada desde su inicio por una gran atención mediática a nivel nacional e internacional debido a la sucesión de declaraciones polémicas por parte de Trump. Sus propuestas más repetidas consisten en la construcción de un muro a lo largo de la frontera con México y una política dura contra la inmigración legal, además de una prohibición temporal de la entrada de musulmanes en los Estados Unidos. En lo económico, aboga por modificar la política comercial del país y fortalecer la producción nacional en detrimento de la deslocalización, en consonancia con posiciones proteccionistas. (en)

| Property                         | Value   |
|----------------------------------|---|
| <a href="#">dbpedia:abstract</a> | <ul style="list-style-type: none"><li>Donald John Trump (Queens, Nueva York, 14 de junio de 1946) es un empresario, político, personalidad televisiva y escritor estadounidense, actual candidato del Partido Republicano a la Presidencia de los Estados Unidos para las elecciones presidenciales de 2016. Es presidente de la Trump Organization y fundador de la empresa de hotel y juegos de azar Trump Entertainment Resorts, que es ahora propiedad de Carl Icahn. Trump es una celebridad televisiva, fue el presentador del reality show The Apprentice, de la NBC, entre 2004 y 2015. Es hijo de un empresario inmobiliario de Nueva York, en cuya compañía, Elizabeth Trump &amp; Son, trabajó mientras estudiaba en la Escuela de Negocios Wharton de la Universidad de Pennsylvania. En 1968, se unió oficialmente a esa sociedad, que controla desde 1971, cuando la renombró Trump Organization. En los años 1990 la empresa entró en bancarrota comercial, pero en la década siguiente se recuperó, lo que le reportó una fortuna de varios miles de millones de dólares. Su campaña para obtener la candidatura republicana a la Casa Blanca para las elecciones de 2016 se vio caracterizada desde su inicio por una gran atención mediática a nivel nacional e internacional debido a la sucesión de declaraciones polémicas por parte de Trump. Sus propuestas más repetidas consisten en la construcción de un muro a lo largo de la frontera con México y una política dura contra la inmigración legal, además de una prohibición temporal de la entrada de musulmanes en los Estados Unidos. En lo económico, aboga por modificar la política comercial del país y fortalecer la producción nacional en detrimento de la deslocalización, en consonancia con posiciones proteccionistas. (en)</li></ul> |

# Desambiguación semántica no supervisada

- Si no se dispone de un corpus de entrenamiento lo suficientemente grande, las estrategias anteriores no funcionan
  - Puede realizarse una **estrategia no supervisada** que trate de desvelar los posibles varios sentidos de una palabra usando un corpus sin anotaciones
  - El algoritmo no supervisado encuentra sus propios synsets o clusters de significados, que no tienen por qué corresponder con synsets o categorías semánticas reales, pero pueden ser una buena aproximación
- 
- Aproximación sencilla
    - Calcular **vectores de características** para todas las palabras del corpus (similar a la aproximación supervisada)
    - Crear **clusters** de palabras usando esas características
      - Distancias: distancia euclídea, distancia coseno, ...
      - Algoritmo de clustering: k-means, métodos de clustering jerárquicos, ...
    - En predicción, calcular vector de características de palabras nuevas y asignar al cluster más cercano
  - La elección de una función de distancia más avanzada puede producir mejores resultados



# Un método no supervisado: distancia web

- Internet es un gran corpus que puede utilizarse para tratar de aprender relaciones de similitud entre palabras de forma no supervisada

- Distancia web entre palabras: 
$$d(w_1, w_2) = \log \frac{P(w_1, w_2)}{P(w_1)P(w_2)}$$

- **Numerador**: probabilidad de que las dos palabras aparezcan en la misma página / frase
  - Número de resultados que da un buscador al buscar  $w_1$  AND  $w_2$
- **Denominador**: probabilidad de que aparezca la primera palabra en una página o frase, multiplicado por lo mismo para la segunda palabra
  - Número de resultados que da un buscador al buscar  $w_1$  + resultados al buscar  $w_2$

# Distancia web normalizada

- Puede obtenerse una mejor métrica enlazando la distancia web con la complejidad de Kolmogorov
- Se considera que  $K(w)$  puede aproximarse como

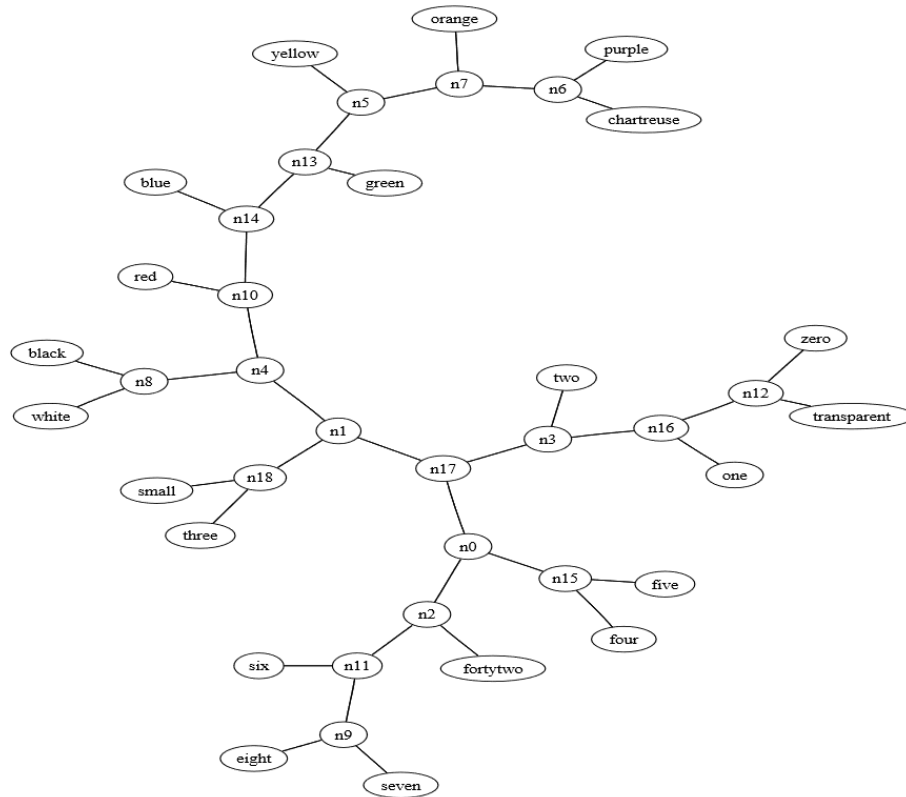
$$K(w) \simeq G(w) = \log \left( \frac{1}{g(w)} \right) = \log \left( \frac{N}{f(w)} \right)$$

con  $g(w)$  la probabilidad de que la palabra  $w$  aparezca en una página de Internet,  $N$  el total de páginas de Internet y  $f(w)$  el número de páginas en las que aparece la palabra  $w$

- Aplicando esta aproximación  $G(w)$  la distancia de Kolmogorov resulta en la distancia web normalizada (NWD, Normalized Web Distance)

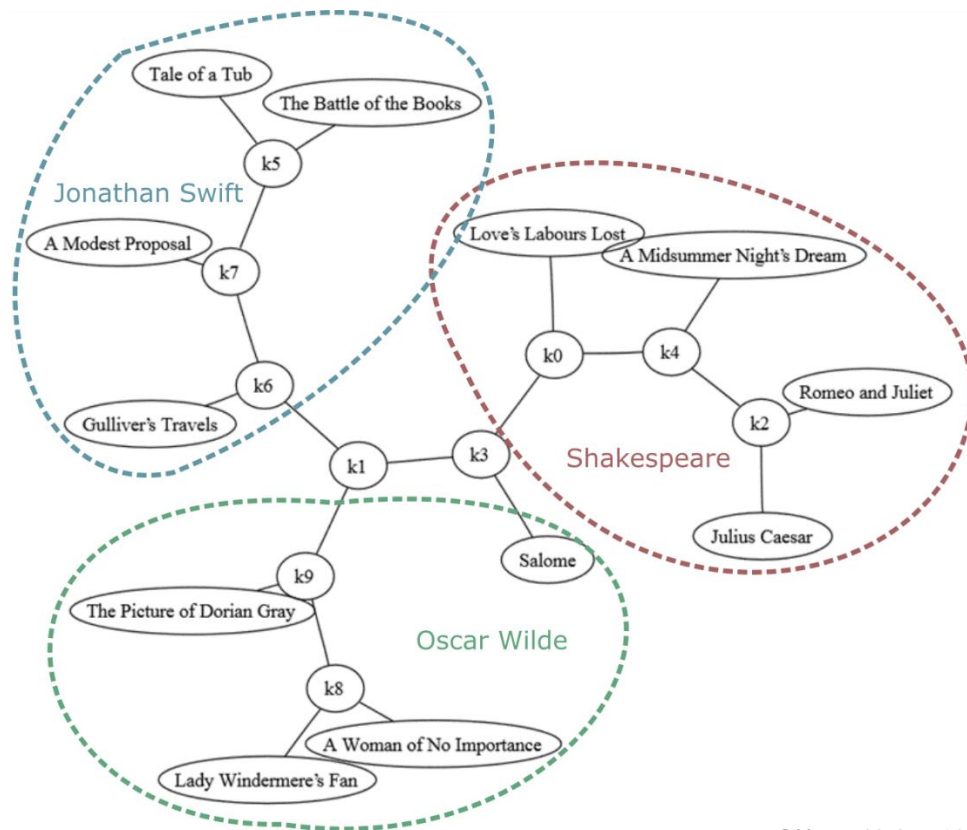
$$e_G(w_1, w_2) = \frac{\max \{ \log f(w_1), \log f(w_2) \} - \log f(w_1, w_2)}{\log N - \min \{ f(w_1), f(w_2) \}}$$

## Ejemplo NWD: clustering de colores y números





# Ejemplo NWD: clustering títulos de libros



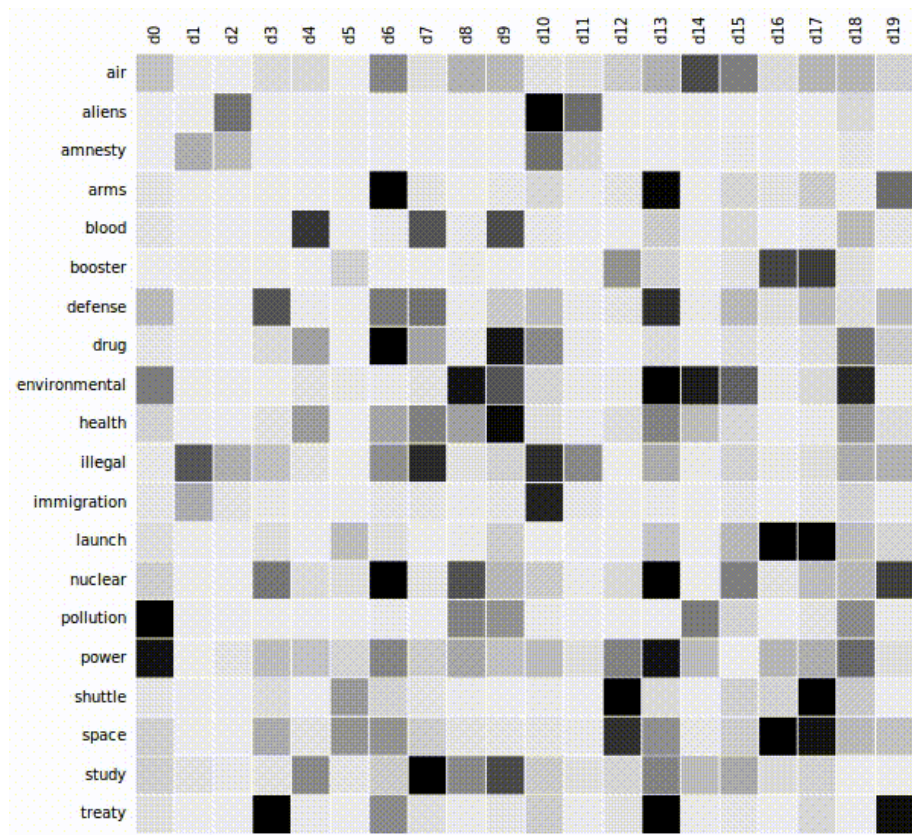
Cilibrasi, Vitányi: Normalized Web Distance and Word Similarity

# Topic Modelling

- Dado un gran corpus de documentos es posible analizar de forma no supervisada los temas (topics) que parecen tratarse en él, y asignar a cada documento una probabilidad de pertenencia a cada tema, así como las palabras representativas de cada tema.
- Puede entenderse como un clustering de documentos (no palabras) según semántica.
- Existen diversas técnicas para realizar esto
  - Latent Semantic Analysis (LSA)
  - Non-negative Matrix Factorization (NMF)
  - Latent Dirichlet Allocation (LDA)

# Matriz de documentos-palabras

- La base de todos los métodos de topic modelling es la **matriz de documentos-palabras**
  - $X_{ij}$  = frecuencia de la palabra  $i$  en el documento  $j$  del corpus
- La frecuencia puede medirse de forma binaria (aparece / no aparece), como conteo de apariciones, con TF-IDF...
- Una reordenación adecuada de esta matriz revela grupos semánticos.
  - ¿Pueden encontrarse estos grupos automáticamente?



[https://en.wikipedia.org/wiki/Topic\\_model](https://en.wikipedia.org/wiki/Topic_model)

# Singular Value Decomposition

- Para facilitar el análisis de la matriz  $X \in \mathbb{R}^{m \times n}$  de  $n$  documentos y  $m$  palabras con rango  $r$  puede emplearse la **descomposición en valores singulares** (Singular Value Decomposition, SVD):

$$X = U\Sigma V^T$$

- $U \in \mathbb{R}^{m \times r}$  matriz ortogonal ( $U^T \cdot U = I$ ) de vectores singulares
  - $\Sigma \in \mathbb{R}^{r \times r}$  matriz diagonal de valores singulares  $\Sigma_{1,1} \geq \Sigma_{2,2} \geq \dots \geq \Sigma_{r,r}$
  - $V \in \mathbb{R}^{r \times n}$  matriz ortogonal ( $V^T \cdot V = I$ ) de vectores singulares
- 
- La descomposición SVD puede interpretarse como expresar la misma matriz  $X$  en un espacio alternativo donde toma forma  $\Sigma$ 
    - Las matrices  $U$  y  $V$  realizan la transformación del espacio hacia/desde ese espacio alternativo.
  - Si existe información redundante en  $X$ :  $r < n, m$

# Latent Semantic Analysis

- Cuando se aplica SVD sobre la matriz de documentos-palabras se tiene la siguiente interpretación:
  - Las entradas de los vectores singulares  $U_{i,j}$  pueden interpretarse como la importancia que tiene cada palabra  $i$  en el tema  $j$
  - Las entradas de los vectores singulares  $V_{j,d}$  pueden interpretarse como la importancia que tiene el tema  $j$  en el documento  $d$  del corpus
- Si se sospecha que en un corpus existen solo  $k$  temas o clusters, puede truncarse la descomposición SVD como

$$X = U_{:,1:k} \Sigma_{1:k,1:k} V_{1:k,:}^T$$

- Esto es, seleccionar solo los  $k$  valores singulares de  $\Sigma$  más grandes, así como los vectores singulares correspondientes

# Ejemplo de LSA

- i. The Neatest Little Guide to Stock Market Investing
- ii. Investing For Dummies, 4th Edition
- iii. The Little Book of Common Sense Investing: The Only Way to Guarantee Your Fair Share of Stock Market Returns
- iv. The Little Book of Value Investing
- v. Value Investing: From Graham to Buffett and Beyond
- vi. Rich Dad's Guide to Investing: What the Rich Invest in, That the Poor and the Middle Class Do Not!
- vii. Investing in Real Estate, 5th Edition
- viii. Stock Investing For Dummies
- ix. Rich Dad's Advisors: The ABC's of Real Estate Investing: The Secrets of Finding Hidden Profits Most Investors Miss



| Index Words | Titles |    |    |    |    |    |    |    |    |
|-------------|--------|----|----|----|----|----|----|----|----|
|             | T1     | T2 | T3 | T4 | T5 | T6 | T7 | T8 | T9 |
| book        |        |    | 1  | 1  |    |    |    |    |    |
| dads        |        |    |    |    |    | 1  |    |    | 1  |
| dummies     |        | 1  |    |    |    |    |    | 1  |    |
| estate      |        |    |    |    |    |    | 1  |    | 1  |
| guide       | 1      |    |    |    |    | 1  |    |    |    |
| investing   | 1      | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| market      | 1      |    | 1  |    |    |    |    |    |    |
| real        |        |    |    |    |    |    | 1  |    | 1  |
| rich        |        |    |    |    |    | 2  |    |    | 1  |
| stock       | 1      |    | 1  |    |    |    |    | 1  |    |
| value       |        |    |    | 1  | 1  |    |    |    |    |

# Ejemplo de LSA

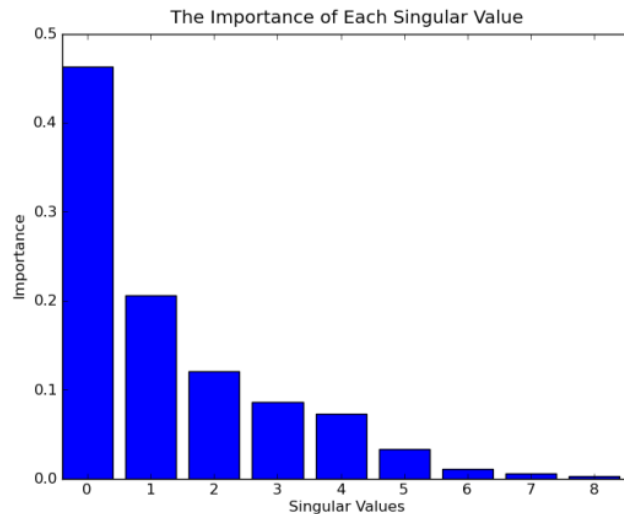
| Index Words | Titles |    |    |    |    |    |    |    |    |
|-------------|--------|----|----|----|----|----|----|----|----|
|             | T1     | T2 | T3 | T4 | T5 | T6 | T7 | T8 | T9 |
| book        |        |    | 1  | 1  |    |    |    |    |    |
| dads        |        |    |    |    |    | 1  |    |    | 1  |
| dummies     |        | 1  |    |    |    |    |    | 1  |    |
| estate      |        |    |    |    |    |    | 1  |    | 1  |
| guide       | 1      |    |    |    |    | 1  |    |    |    |
| investing   | 1      | 1  | 1  | 1  | 1  | 1  | 1  | 1  | 1  |
| market      | 1      |    | 1  |    |    |    |    |    |    |
| real        |        |    |    |    |    |    | 1  |    | 1  |
| rich        |        |    |    |    |    | 2  |    |    | 1  |
| stock       | 1      |    | 1  |    |    |    |    | 1  |    |
| value       |        |    |    | 1  | 1  |    |    |    |    |

SVD

|           |      |       |       |
|-----------|------|-------|-------|
| book      | 0.15 | -0.27 | 0.04  |
| dads      | 0.24 | 0.38  | -0.09 |
| dummies   | 0.13 | -0.17 | 0.07  |
| estate    | 0.18 | 0.19  | 0.45  |
| guide     | 0.22 | 0.09  | -0.46 |
| investing | 0.74 | -0.21 | 0.21  |
| market    | 0.18 | -0.3  | -0.28 |
| real      | 0.18 | 0.19  | 0.45  |
| rich      | 0.36 | 0.59  | -0.34 |
| stock     | 0.25 | -0.42 | -0.28 |
| value     | 0.12 | -0.14 | 0.23  |

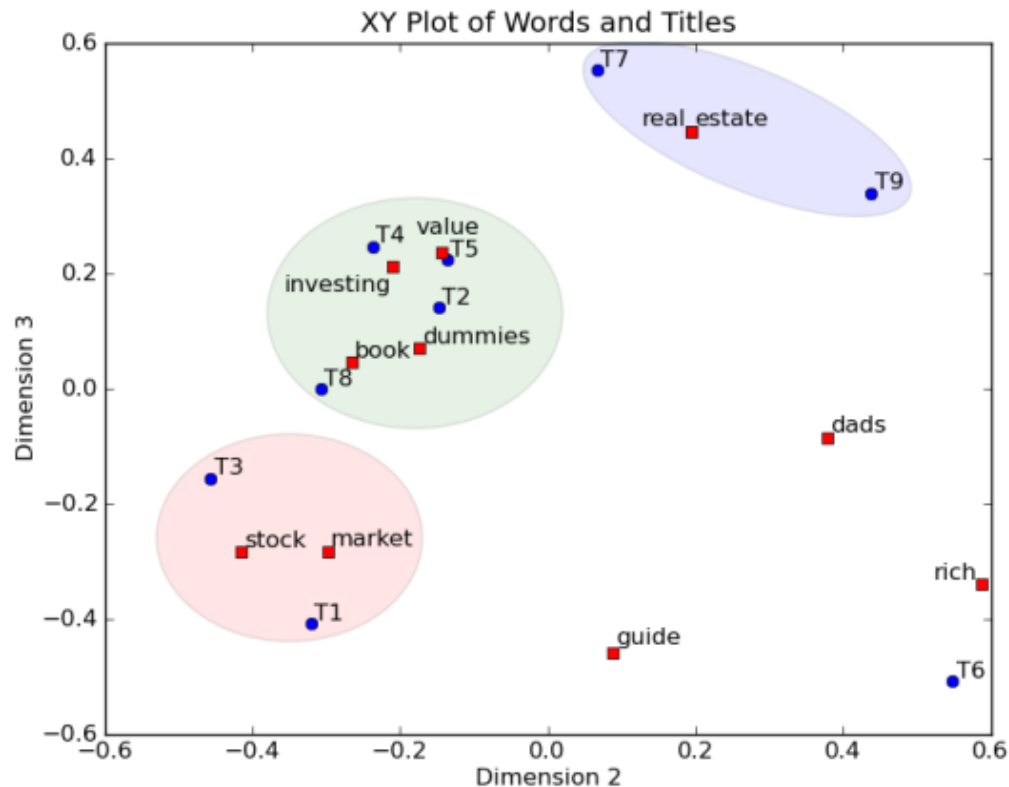
|      |      |   |
|------|------|---|
| 3.91 | 0    | 0 |
| 0    | 2.61 | 0 |
| 0    | 0    | 2 |

| T1    | T2    | T3    | T4    | T5    | T6    | T7   | T8    | T9   |
|-------|-------|-------|-------|-------|-------|------|-------|------|
| 0.35  | 0.22  | 0.34  | 0.26  | 0.22  | 0.49  | 0.28 | 0.29  | 0.44 |
| -0.32 | -0.15 | -0.46 | -0.24 | -0.14 | 0.55  | 0.07 | -0.31 | 0.44 |
| -0.41 | 0.14  | -0.16 | 0.25  | 0.22  | -0.51 | 0.55 | 0     | 0.34 |



<https://technowiki.wordpress.com/2011/08/27/latent-semantic-analysis-lsa-tutorial/>

# Ejemplo de LSA



(clusters obtenidos mediante inspección visual)

<https://technowiki.wordpress.com/2011/08/27/latent-semantic-analysis-lsa-tutorial/>



# Non-negative Matrix Factorization

$$\begin{matrix} W \\ \begin{bmatrix} & & \\ & & \\ & & \\ & & \end{bmatrix} \end{matrix} \times \begin{matrix} H \\ \begin{bmatrix} & & & & \\ & & & & \\ & & & & \\ & & & & \end{bmatrix} \end{matrix} \approx \begin{matrix} V \\ \begin{bmatrix} & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \end{bmatrix} \end{matrix}$$

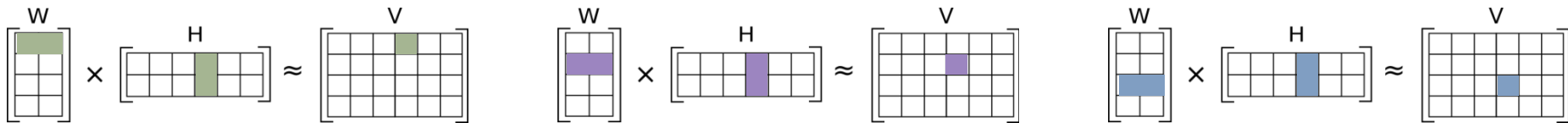
- Un método alternativo a LSA para descomponer la matriz de documentos-palabras es **Non-negative Matrix Factorization** (NMF).
  - La matriz de documentos-palabras (aquí  $V \in \mathbb{R}^{m \times n}$ ) se descompone en la multiplicación de dos matrices,  $W \in \mathbb{R}^{m \times k}$  y  $H \in \mathbb{R}^{k \times n}$ , las cuales solo pueden contener valores  $\geq 0$ .
- Para encontrar esta descomposición, se resuelve el problema de optimización  $\min_{W,H} \|V - WH\|_2$  s.t.  $W \geq 0, H \geq 0$ .
  - Esto puede hacerse alternando pasos de optimización sobre  $W$  y sobre  $H$
  - Una estrategia habitual es utilizar las ecuaciones

$$H_{[i,j]} \leftarrow H_{[i,j]} \frac{(W^T V)_{[i,j]}}{(W^T W H)_{[i,j]}}$$

$$W_{[i,j]} \leftarrow W_{[i,j]} \frac{(V H^T)_{[i,j]}}{(W H H^T)_{[i,j]}}$$

# Non-negative matrix factorization: interpretación

- $W \in \mathbb{R}^{m \times k}$  tiene una fila para cada palabra. Los valores en esas filas indican la importancia de cada palabra en cada uno de los  $k$  temas encontrados.
- $H \in \mathbb{R}^{k \times n}$  tiene una columna para cada documento. Los valores en esas columnas indican la importancia dentro de cada documento de cada uno de los  $k$  temas encontrados.
- $V = WH \rightarrow V_{:,i} = WH_{:,i} \rightarrow V_{ij} = W_{j,:}H_{:,i}$ 
  - Esto es, la frecuencia de aparición de la palabra  $j$  en el documento  $i$  ( $V_{ij}$ ) viene dada por la combinación lineal de la importancia de esa palabra  $j$  en cada tema ( $W_{j,:}$ ) y la importancia de cada tema en el documento  $i$  ( $H_{:,i}$ )



# Non-negative matrix factorization: ejemplo

| Term   | Document |   |   |   |   |  |
|--------|----------|---|---|---|---|--|
|        | 1        | 2 | 3 | 4 | 5 |  |
| "one"  | 1        |   |   |   |   |  |
| "two"  | 1        |   |   |   |   |  |
| "fish" | 2        | 2 | 2 |   |   |  |
| "red"  |          | 1 |   | 1 |   |  |
| "blue" |          | 1 |   | 1 |   |  |
| "old"  |          |   | 1 |   | 1 |  |
| "new"  |          |   | 1 |   | 1 |  |
| "some" |          |   |   | 2 | 2 |  |
| "are"  |          |   |   | 2 | 2 |  |
| "and"  |          |   |   | 1 | 1 |  |

$\approx$

| Term   | Document |   |   |   |     |  |
|--------|----------|---|---|---|-----|--|
|        | 1        | 2 | 3 | 4 | 5   |  |
| "one"  | 1        |   |   |   |     |  |
| "two"  | 1        |   |   |   |     |  |
| "fish" |          | 1 |   |   |     |  |
| "red"  |          |   | 1 |   |     |  |
| "blue" |          |   | 1 |   |     |  |
| "old"  |          |   |   | 1 |     |  |
| "new"  |          |   |   | 1 |     |  |
| "some" |          |   |   |   | 1   |  |
| "are"  |          |   |   |   | 1   |  |
| "and"  |          |   |   |   | 0.5 |  |

$\approx$

| Term                         | Document |   |   |   |   |  |
|------------------------------|----------|---|---|---|---|--|
|                              | 1        | 2 | 3 | 4 | 5 |  |
| "one" + "two"                | 1        |   |   |   |   |  |
| "fish"                       | 2        | 2 | 2 |   |   |  |
| "red" + "blue"               |          | 1 |   | 1 |   |  |
| "old + new"                  |          |   | 1 |   | 1 |  |
| "some" + "are" + 0.5 · "and" |          |   |   | 2 | 2 |  |

(Nombres de temas añadidos manualmente)  
Bergen – Text Mining and Classification

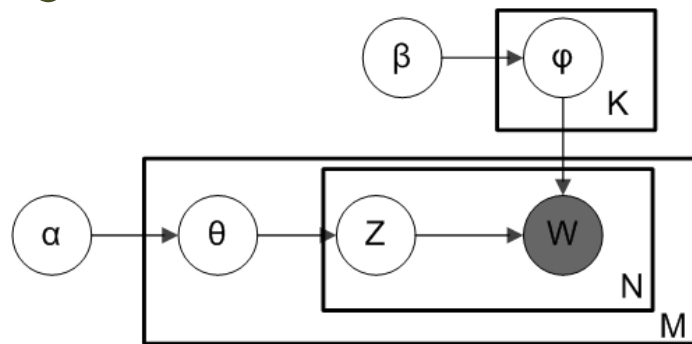
# Latent Dirichlet Allocation

- Un problema de LSA (y de algunos tipos de NMF) es que la descomposición puede ser muy densa: un documento puede estar compuesto de gran cantidad de temas, y a su vez una palabra puede ser representativa de muchos temas.
  - Esto hace difícil la interpretación de los temas encontrados
- Latent Dirichlet Allocation (LDA) es un método que trata de modelar el proceso de generación del texto mediante distribuciones de probabilidad, imponiendo restricciones de dispersión (sparsity) sobre esas distribuciones.
- Vamos a estudiar cómo LDA define la generación del texto.

# Latent Dirichlet Allocation: modelo generativo

## ➤ Comenzamos definiendo:

- $K$ , número de temas.
- $\alpha$ , priori de probabilidad de cada tema, siguiendo una distribución de Dirichlet
- $\beta$ , priori de probabilidades de cada palabra, siguiendo una distribución de Dirichlet



- Para cada tema  $k$ , generamos las listas de probabilidades de aparición de cada palabra en ese tema como  $\varphi_k \sim \text{Dir}(\beta)$
- Para cada documento  $d_i$  a generar
  - Escoger el subconjunto de temas de los que trata,  $\theta_i \sim \text{Dir}(\alpha)$ .
  - Para cada palabra  $w_{ij}$  del documento
    - Escoger el tema concreto de esta palabra:  $z_{ij} \sim \text{Multinomial}(\theta_i)$ .
    - Escoger la palabra concreta según probabilidades del tema:  $w_{ij} \sim \text{Multinomial}(\varphi_{z_{ij}})$

Blei et al – Latent Dirichlet Allocation

[https://en.wikipedia.org/wiki/Latent\\_Dirichlet\\_allocation](https://en.wikipedia.org/wiki/Latent_Dirichlet_allocation)

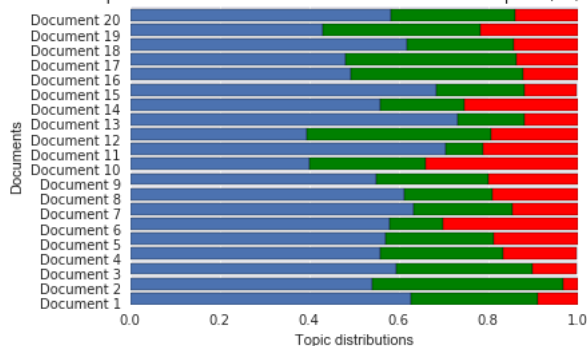
# Latent Dirichlet Allocation: propiedades Dirichlet

- La función de densidad de probabilidad de la distribución de Dirichlet es:

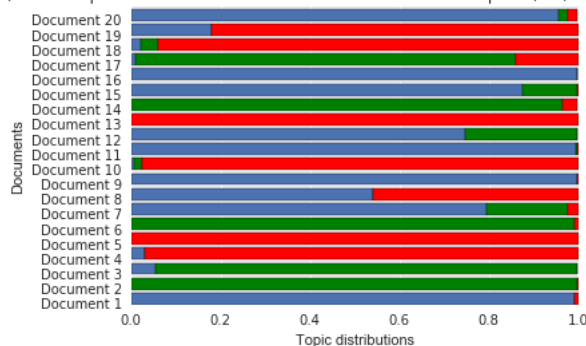
$$f(x_1, \dots, x_K; \alpha_1, \dots, \alpha_K) = \frac{1}{B(\alpha)} \prod_{i=1}^K x_i^{\alpha_i - 1}$$

- $B(\alpha)$  es la función Beta multivariante:  $B(\alpha) = \frac{\prod_{i=1}^K \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^K \alpha_i)}$ , que depende de la función Gamma,  $\Gamma(z) = \int_0^\infty x^{z-1} e^{-x} dx$ .
- El valor de los parámetros  $\alpha$  permite definir la abundancia de cada tema, y si la distribución de temas tenderá a ser uniforme o dispersa.
- Lo habitual es tomar todos los valores de  $\alpha$  como iguales, pero definir si serán valores grandes o pequeños para imponer más o menos dispersión de temas.

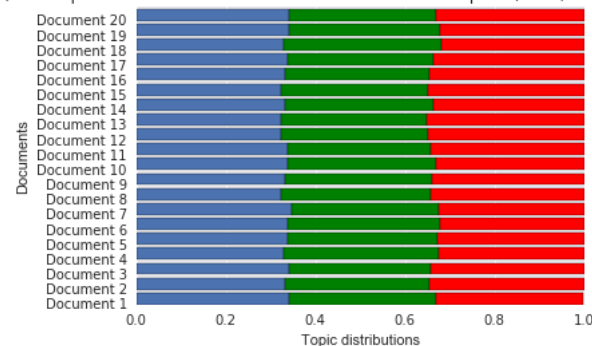
Topics distribution in each document for Dirichlet alpha=(10, 5, 3)



Topics distribution in each document for Dirichlet alpha=(0.1, 0.1, 0.1)



Topics distribution in each document for Dirichlet alpha=(1000, 1000, 1000)



# Latent Dirichlet Allocation: aprendiendo los parámetros

- La función de probabilidad conjunta de todo el modelo, fijando hiperparámetros  $K, \alpha, \beta$ , es

$$P(\mathbf{w}, \mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\varphi} | \alpha, \beta) = \prod_{k=1}^K P(\varphi_k | \beta) \prod_{i=1}^M P(\theta_i | \alpha) \prod_{j=1}^N P(z_{i,j} | \theta_i) P(w_{i,j} | \varphi_{z_{i,j}})$$

- Usando la regla de probabilidad  $P(A|B) = P(A, B)/P(B)$ , el posterior de esta función de probabilidad tomando unas palabras  $w_{i,j}$  observadas en el corpus es

$$P(\mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\varphi} | \mathbf{w}, \alpha, \beta) = \frac{P(\mathbf{w}, \mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\varphi} | \alpha, \beta)}{P(\mathbf{w} | \alpha, \beta)}$$

- Esta función de probabilidad es muy informativa porque nos dice cómo de adecuados (probables) son unos valores concretos de  $\mathbf{z}$  (tema de cada palabra),  $\boldsymbol{\theta}$  (temas de cada documento) y  $\boldsymbol{\varphi}$  (distribución de palabras por temas) para un corpus de entrenamiento formado por las palabras  $\mathbf{w}$ .

✗ Esta función es extremadamente costosa de calcular debido al término  $P(\mathbf{w} | \alpha, \beta)$ , que no acepta una expresión sencilla.

# Latent Dirichlet Allocation: Gibbs sampling

- Una técnica muy recurrida para obtener muestras de una distribución de probabilidad de muchas variables es Gibbs sampling.
- Suponer una función de probabilidad conjunta  $P(x_1, x_2, \dots, x_N)$  costosa de evaluar:
  - Comenzamos tomando valores arbitrarios para cada variable:  $x_1^0, x_2^0, \dots, x_N^0$
  - Durante un número de épocas  $t$ 
    - Para cada variable  $x_i^t$ , actualizar su valor tomando una muestra de la distribución condicionada al resto de variables:  $P(x_i^t | x_1^{t-1}, x_2^{t-1}, \dots, x_{i-1}^{t-1}, x_{i+1}^{t-1}, \dots, x_N^{t-1})$   
 $= P(x_i^t | x_{-i}^{t-1})$
- Puede demostrarse que la secuencia de valores  $x^1, x^2, x^3, \dots$  converge a una secuencia que sigue la distribución  $P(x_1, x_2, \dots, x_N)$ 
  - Markov Chain Monte Carlo



# Latent Dirichlet Allocation: aprendiendo los parámetros

- Siguiendo una estrategia de Gibbs Sampling, puede demostrarse que es posible obtener muestras de  $z$  condicionando/integrando el resto de variables como

$$P(z_{ij} = k | z_{-ij}, w) \propto \frac{n_{-ij}^{(w_{ij} \in k)} + \beta}{n_{-ij} + W\beta} \frac{n_{-ij}^{(d_i \in k)} + \alpha}{n_{-ij}^{(d_i)} + K\alpha}$$

- Donde

- $n_{-ij}^{(w_{ij} \in k)}$  veces que la palabra  $w_{ij}$  aparece asignada al tema  $k$  en otras posiciones del corpus.
- $n_{-ij}$  veces que la palabra  $w_{ij}$  aparece en otras posiciones del corpus
- $n_{-ij}^{(d_i \in k)}$  número de palabras del resto del documento  $d_i$  que están en el tema  $k$
- $n_{-ij}^{(d_i)}$  número de palabras del resto del documento  $d_i$
- $W$  número de palabras diferentes del corpus
- Intuitivamente: el primer término es la probabilidad de  $w_{ij}$  en el tema  $k$ , mientras que el segundo término es la probabilidad del tema  $k$  en el documento  $d_i$ .

## Latent Dirichlet Allocation: aprendiendo los parámetros

- Iterando suficientes veces el proceso de Gibbs Sampling anterior se llega a alcanzar una distribución estacionaria de  $z$  (temas asignados a cada palabra del corpus)
- Con las  $z$  calculadas pueden obtenerse valores para el resto de parámetros,  $\theta$  y  $\varphi$ :

$$\theta_{ik} \propto \frac{n_k^{(d_i)} + \alpha}{n^{(d_i)} + T\alpha} \simeq \text{fracción de palabras del tema } k \text{ que aparecen en el documento } d_i$$

$$\varphi_k(w) \propto \frac{n^{(w \in k)} + \beta}{n^k + W\beta} \simeq \text{fracción de veces que la palabra } w \text{ aparece en el tema } k$$

# Latent Dirichlet Allocation: ejemplo

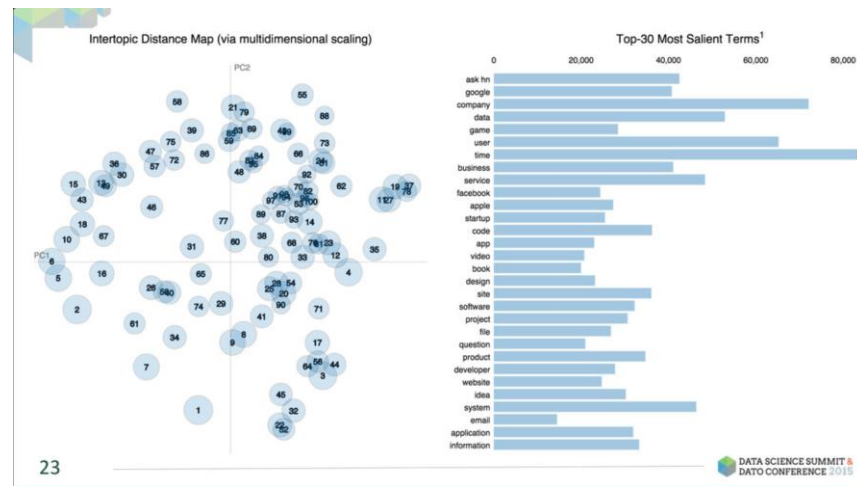
The William Randolph Hearst Foundation will give \$1.25 million to Lincoln Center, Metropolitan Opera Co., New York Philharmonic and Juilliard School. “Our board felt that we had a real opportunity to make a mark on the future of the performing arts with these grants an act every bit as important as our traditional areas of support in health, medical research, education and the social services,” Hearst Foundation President Randolph A. Hearst said Monday in announcing the grants. Lincoln Center’s share will be \$200,000 for its new building, which will house young artists and provide new public facilities. The Metropolitan Opera Co. and New York Philharmonic will receive \$400,000 each. The Juilliard School, where music and the performing arts are taught, will get \$250,000. The Hearst Foundation, a leading supporter of the Lincoln Center Consolidated Corporate Fund, will make its usual annual \$100,000 donation, too.

| “Arts”  | “Budgets”  | “Children” | “Education” |
|---------|------------|------------|-------------|
| NEW     | MILLION    | CHILDREN   | SCHOOL      |
| FILM    | TAX        | WOMEN      | STUDENTS    |
| SHOW    | PROGRAM    | PEOPLE     | SCHOOLS     |
| MUSIC   | BUDGET     | CHILD      | EDUCATION   |
| MOVIE   | BILLION    | YEARS      | TEACHERS    |
| PLAY    | FEDERAL    | FAMILIES   | HIGH        |
| MUSICAL | YEAR       | WORK       | PUBLIC      |
| BEST    | SPENDING   | PARENTS    | TEACHER     |
| ACTOR   | NEW        | SAYS       | BENNETT     |
| FIRST   | STATE      | FAMILY     | MANIGAT     |
| YORK    | PLAN       | WELFARE    | NAMPHY      |
| OPERA   | MONEY      | MEN        | STATE       |
| THEATER | PROGRAMS   | PERCENT    | PRESIDENT   |
| ACTRESS | GOVERNMENT | CARE       | ELEMENTARY  |
| LOVE    | CONGRESS   | LIFE       | HAITI       |

Figure 8: An example article from the AP corpus. Each color codes a different factor from which the word is putatively generated.

# Aplicación: extracción de temas

- **Objetivo:** dado un corpus grande de documentos, hacer un clustering no supervisado para poder organizarlos por temáticas.
- **Aproximación** para el problema
  - Métodos semánticos: LSA, NMF, LDA
- **Recursos**
  - The 20 Newsgroups data set:  
<http://qwone.com/~jason/20NewsGroups/>





Álvaro Barbero Jiménez



@albarjip



<https://github.com/albarji>



[albarji.deviantart.com](https://albarji.deviantart.com)