Laboratorio de datos: web scraping y Procesamiento de Lenguaje Natural

Clase 7. Un acercamiento a los word embeddings

factor~data
EIDAES_UNSAM

Hipótesis distribucional

- “El significado deriva del uso de las palabras en el lenguaje” (Wittgenstein)
- Podemos captar el sentido de las palabras según su “compañía”
- Palabras cercanas tienen sentidos “cercanos”
- Ítems lingüísticos con distribuciones similares tienen significados similares”
- Idea de co-ocurrencia => términos que ocurren juntos

TFM Co-ocurrencia a nivel documento

Palabras, bigramas, trigramas, lemas, solo la raíz de la palabra...

F términos

Matriz M = ***N documentos***

$$\begin{bmatrix} V_{11} & V_{12} & V_{13} & \cdots & V_{1F} \\ V_{N1} & V_{N2} & V_{N3} & \cdots & V_{NF} \end{bmatrix}$$

Frecuencia del término

Fuente: https://github.com/gefero/ws_text_mining/blob/master/slides/04_pinto.pdf

- La matriz de documentos-términos suele tener muchos ceros
- Problema: se hace difícil medir la relación entre los distintos documentos o términos

	Palabra 1	Palabra 2	Palabra 3	Palabra 4	Palabra 5	
Relato 1	0	0.12	0.01	0	0	
Relato 2	0	0	0.44	0.15	0.65	
Relato 3	0.11	0.31	0.28	0	0	(...)
Relato 4	0	0	0.05	0.21	0	
Relato 5	0	0.13	0	0.07	0	
			(...)			

La correlación lineal entre filas nos da una idea de la similitud del significado entre relatos

La correlación lineal entre columnas nos da una idea de la similitud del significado entre palabras

Pero hay un problema: la mayor parte de los valores son 0

“Sobre la mesa hay un florero con margaritas y jazmines”

“El vaso lleno de flores está apoyado sobre una mesada”

- Mismo sentido pero ninguna palabra en común
- Una solución ya la vimos: LDA, STM => detección de tópicos

- **Otra solución: word embeddings**

Word embeddings => idea general

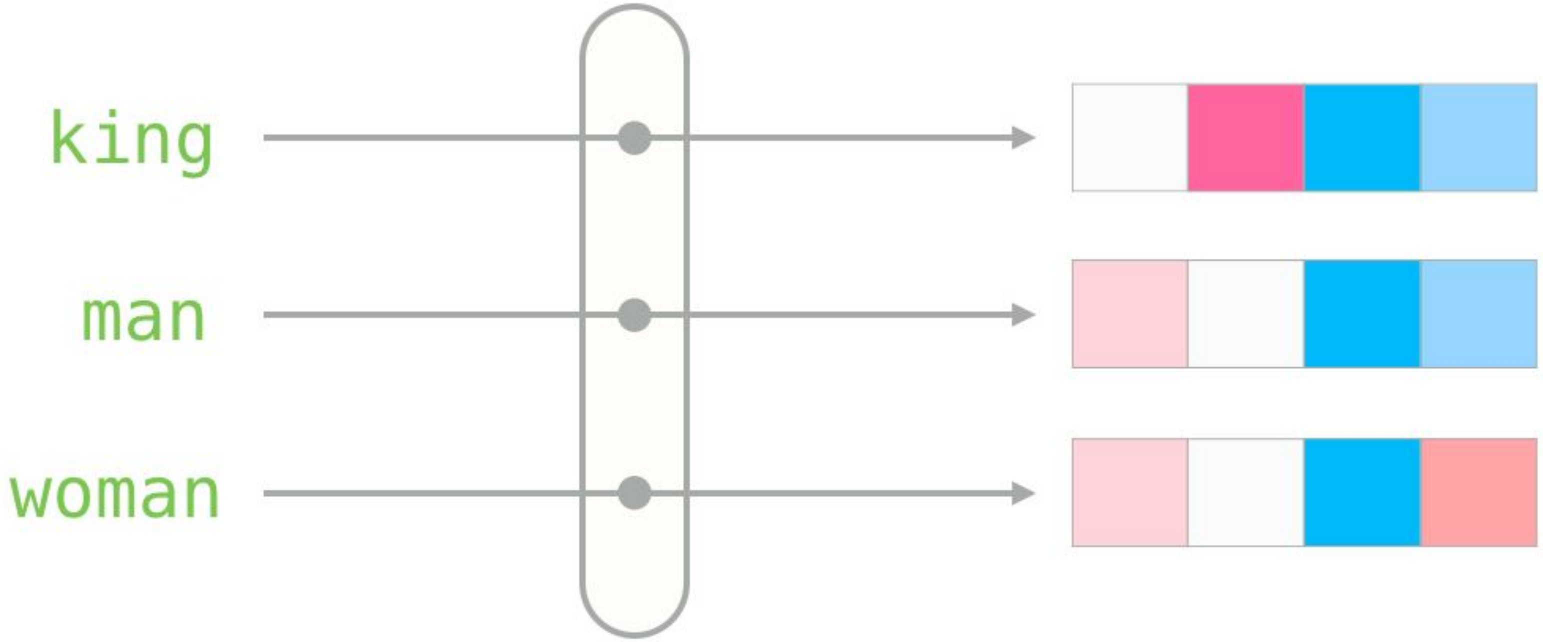
- Reducir la dimensión del vocabulario
  - ~50.000 palabras a ~100 => representación no “esparsa” sino densa

- Flexibilizar supuestos de BoW: cada columna/término/dimensión es un término y se asume independencia

- Hay interacción entre palabras => es esperable que la dimensionalidad sea menor

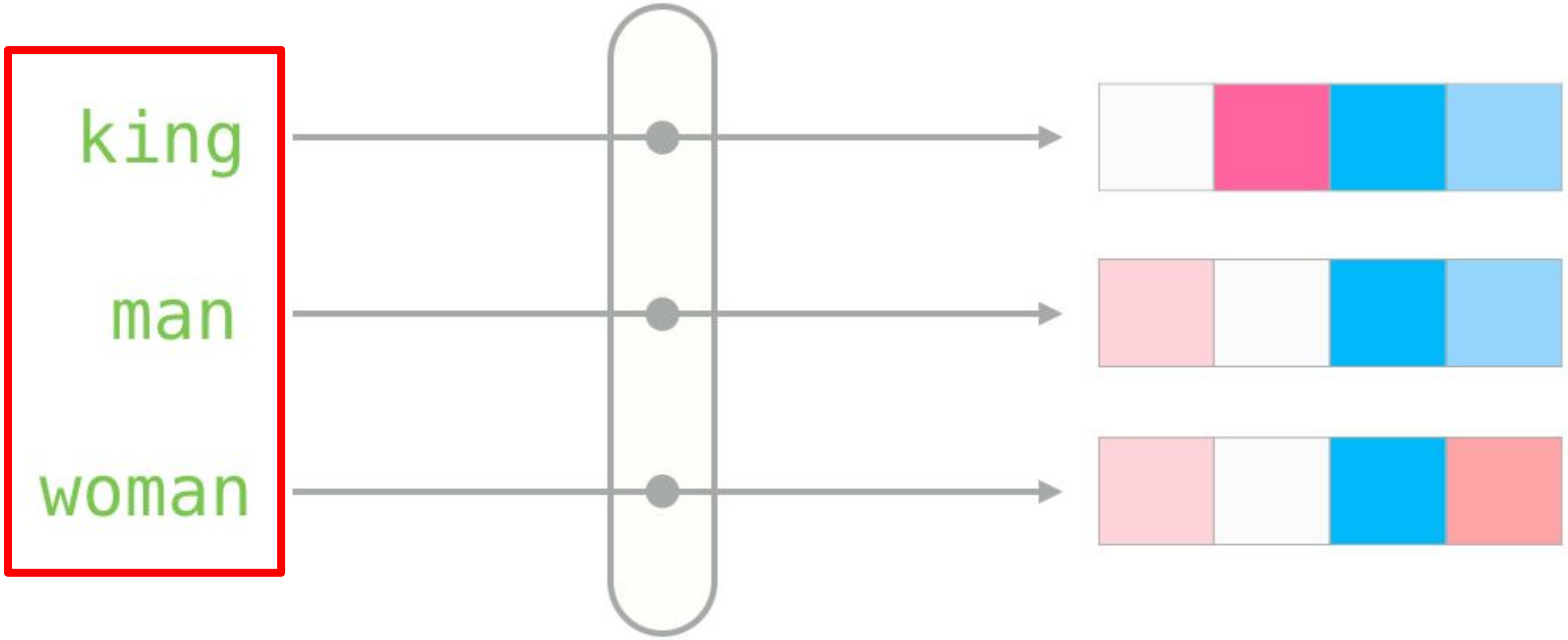
- Lograr introducir una métrica de distancia para que palabras “cerca” en el nuevo espacio estén “cerca” semánticamente estén cerca.

word2vec



Fuente: https://jalammar.github.io/illustrated-word2vec/

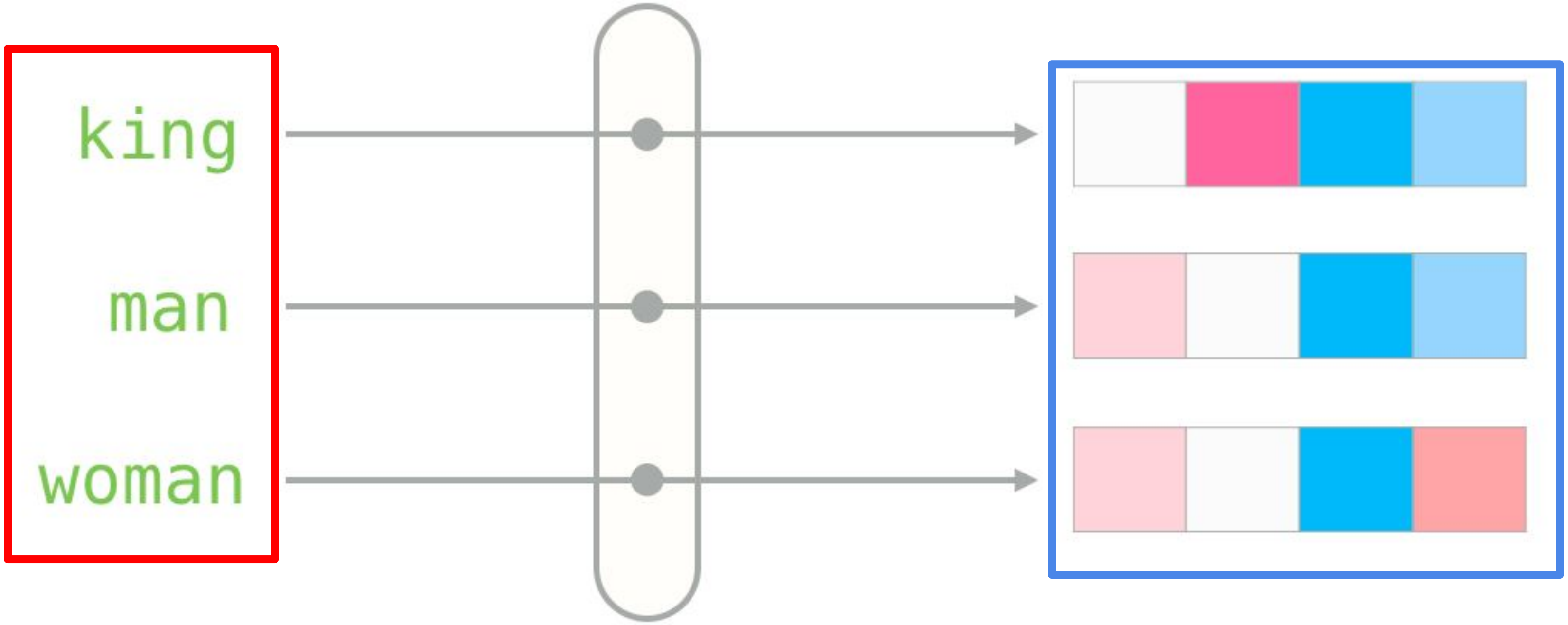
word2vec



factor~data
EIDAES_UNSAM

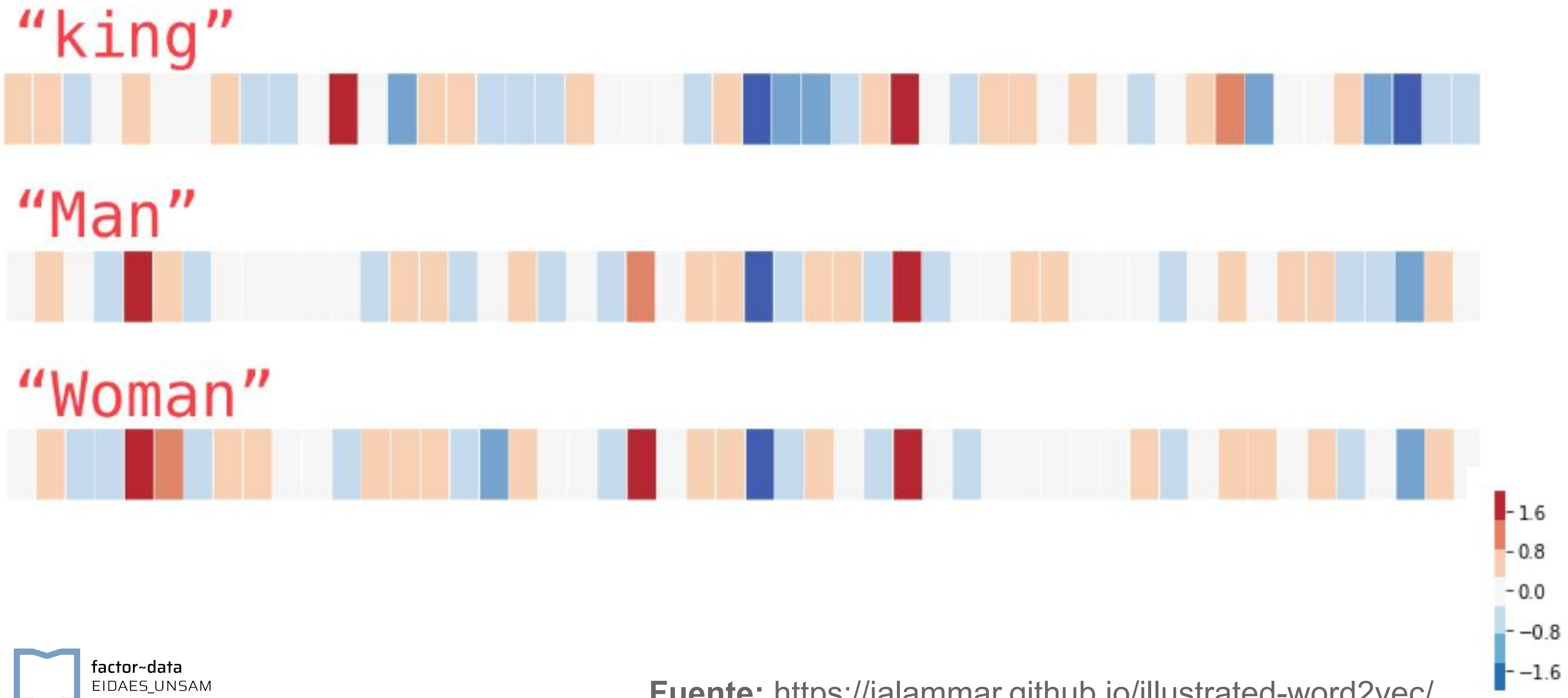
Fuente: https://jalammar.github.io/illustrated-word2vec/

word2vec



Fuente: https://jalammar.github.io/illustrated-word2vec/

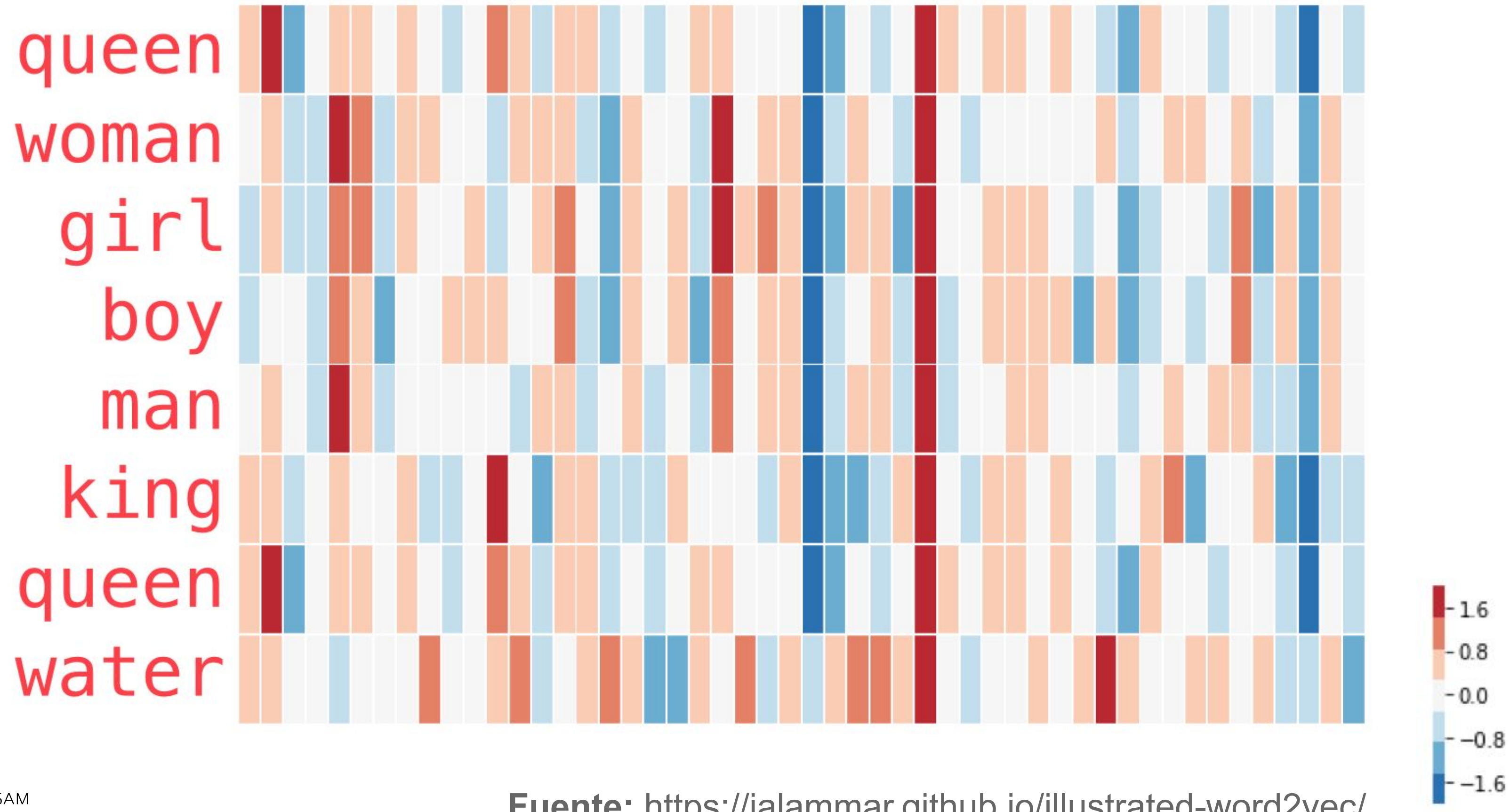
word2vec



factor~data
EIDAES_UNSAM

Fuente: https://jalammar.github.io/illustrated-word2vec/

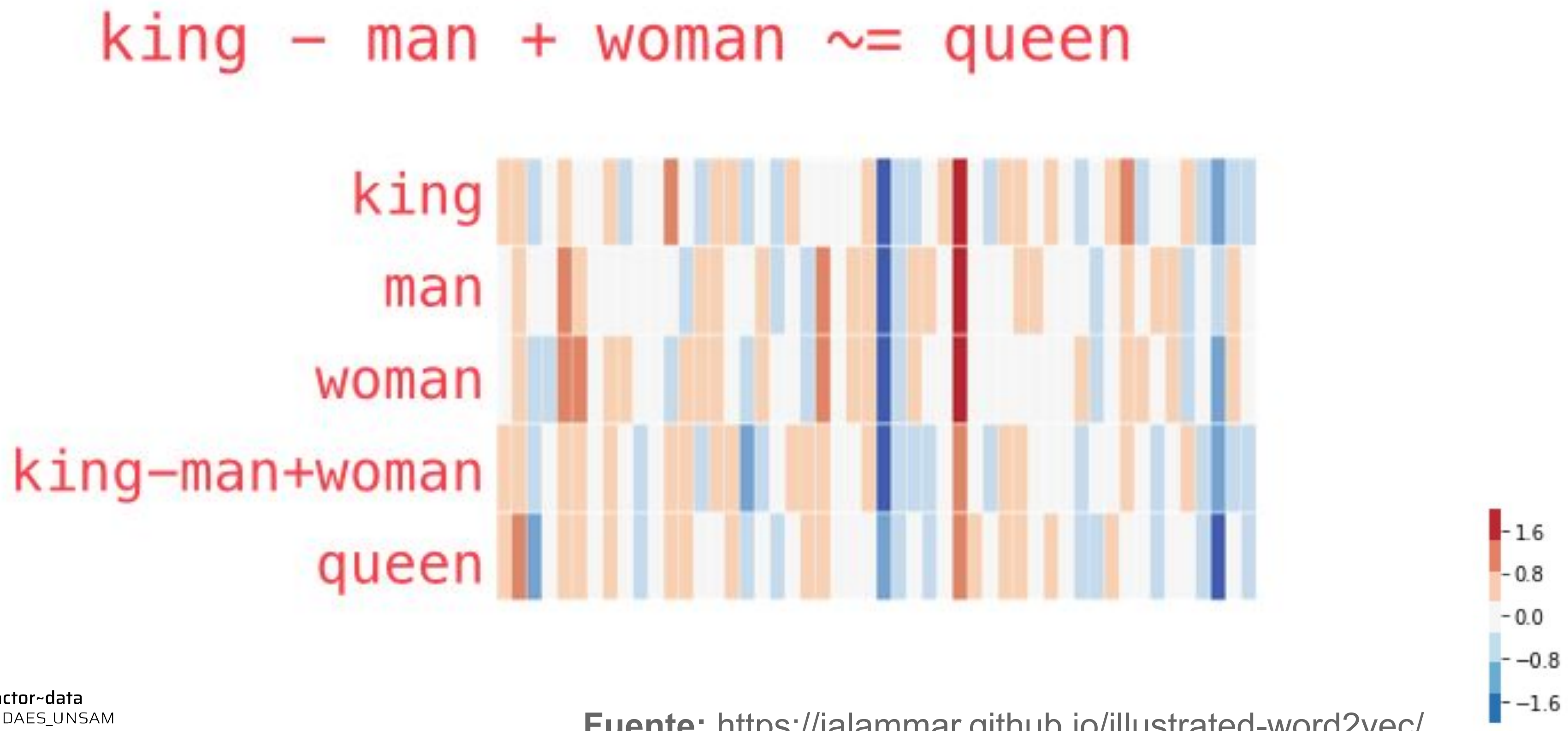
word2vec



Fuente: https://jalammar.github.io/illustrated-word2vec/

factor~data
EIDAES_UNSAM

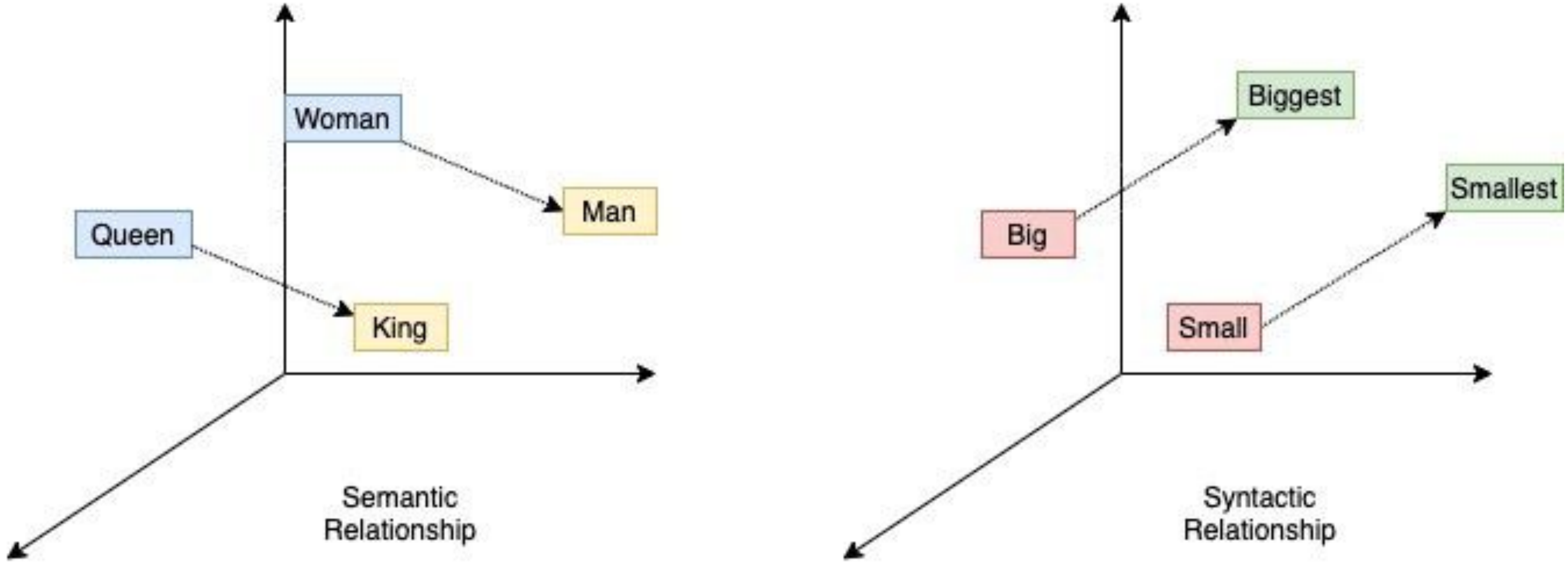
word2vec



Fuente: https://jalammar.github.io/illustrated-word2vec/

factor~data
EIDAES_UNSAM

word2vec


Woman
Man
Queen
King
Semantic Relationship
Biggest
Smallest
Big
Small
Syntactic Relationship

Evaluación de embeddings

Table 1: *Examples of five types of semantic and nine types of syntactic questions in the Semantic-Syntactic Word Relationship test set.*

Type of relationship	Word Pair 1		Word Pair 2	
Common capital city	Athens	Greece	Oslo	Norway
All capital cities	Astana	Kazakhstan	Harare	Zimbabwe
Currency	Angola	kwanza	Iran	rial
City-in-state	Chicago	Illinois	Stockton	California
Man-Woman	brother	sister	grandson	granddaughter
Adjective to adverb	apparent	apparently	rapid	rapidly
Opposite	possibly	impossibly	ethical	unethical
Comparative	great	greater	tough	tougher
Superlative	easy	easiest	lucky	luckiest
Present Participle	think	thinking	read	reading
Nationality adjective	Switzerland	Swiss	Cambodia	Cambodian
Past tense	walking	walked	swimming	swam
Plural nouns	mouse	mice	dollar	dollars
Plural verbs	work	works	speak	speaks

Fuente: Efficient Estimation of Word Representations in Vector Space (2013) https://arxiv.org/abs/1301.3781 Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean

Evaluación de embeddings

Table 4: *Comparison of publicly available word vectors on the Semantic-Syntactic Word Relationship test set, and word vectors from our models. Full vocabularies are used.*

Model	Vector Dimensionality	Training words	Accuracy [%]		
			Semantic	Syntactic	Total
Collobert-Weston NNLM	50	660M	9.3	12.3	11.0
Turian NNLM	50	37M	1.4	2.6	2.1
Turian NNLM	200	37M	1.4	2.2	1.8
Mnih NNLM	50	37M	1.8	9.1	5.8
Mnih NNLM	100	37M	3.3	13.2	8.8
Mikolov RNNLM	80	320M	4.9	18.4	12.7
Mikolov RNNLM	640	320M	8.6	36.5	24.6
Huang NNLM	50	990M	13.3	11.6	12.3
Our NNLM	20	6B	12.9	26.4	20.3
Our NNLM	50	6B	27.9	55.8	43.2
Our NNLM	100	6B	34.2	**64.5**	50.8
CBOW	300	783M	15.5	53.1	36.1
Skip-gram	300	783M	**50.0**	55.9	**53.3**

Fuente: Efficient Estimation of Word Representations in Vector Space (2013) https://arxiv.org/abs/1301.3781 Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean

Evaluación de embeddings

Table 8: *Examples of the word pair relationships, using the best word vectors from Table 4 (Skip-gram model trained on 783M words with 300 dimensionality).*

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

Fuente: Efficient Estimation of Word Representations in Vector Space (2013) https://arxiv.org/abs/1301.3781 Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean

Usos posibles

- Similitud entre palabras y documentos
- Similitud entre palabras “target” y palabras de contexto al resultado

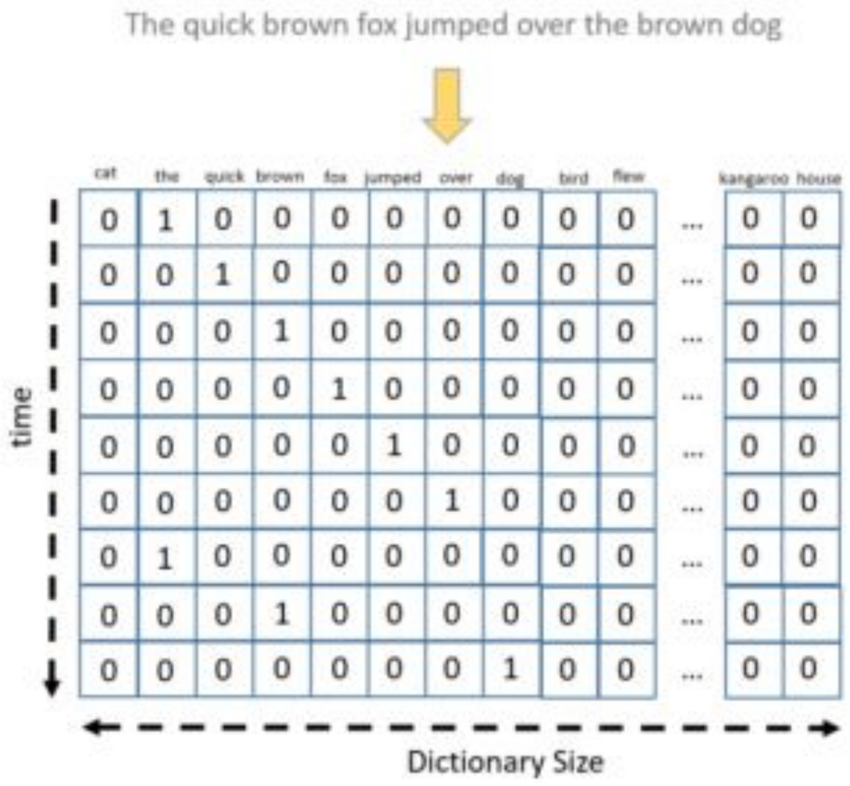
- Autocompletado
- Traducción automática
- Encontrar clusters de palabras con significados similares
- Buscar analogías entre palabras

- Modelo semántico del lenguaje para comparar con procesamiento del lenguaje hecho por humanos

¿Cómo sucede la magia?

One hot encoding

- Eje Y = tiempo
- Eje X = vocabulario
- Celdas: 1 si la palabra aparece en ese “momento”; 0 si no aparece



factor~data
EIDAES_UNSAM

Skip-gram

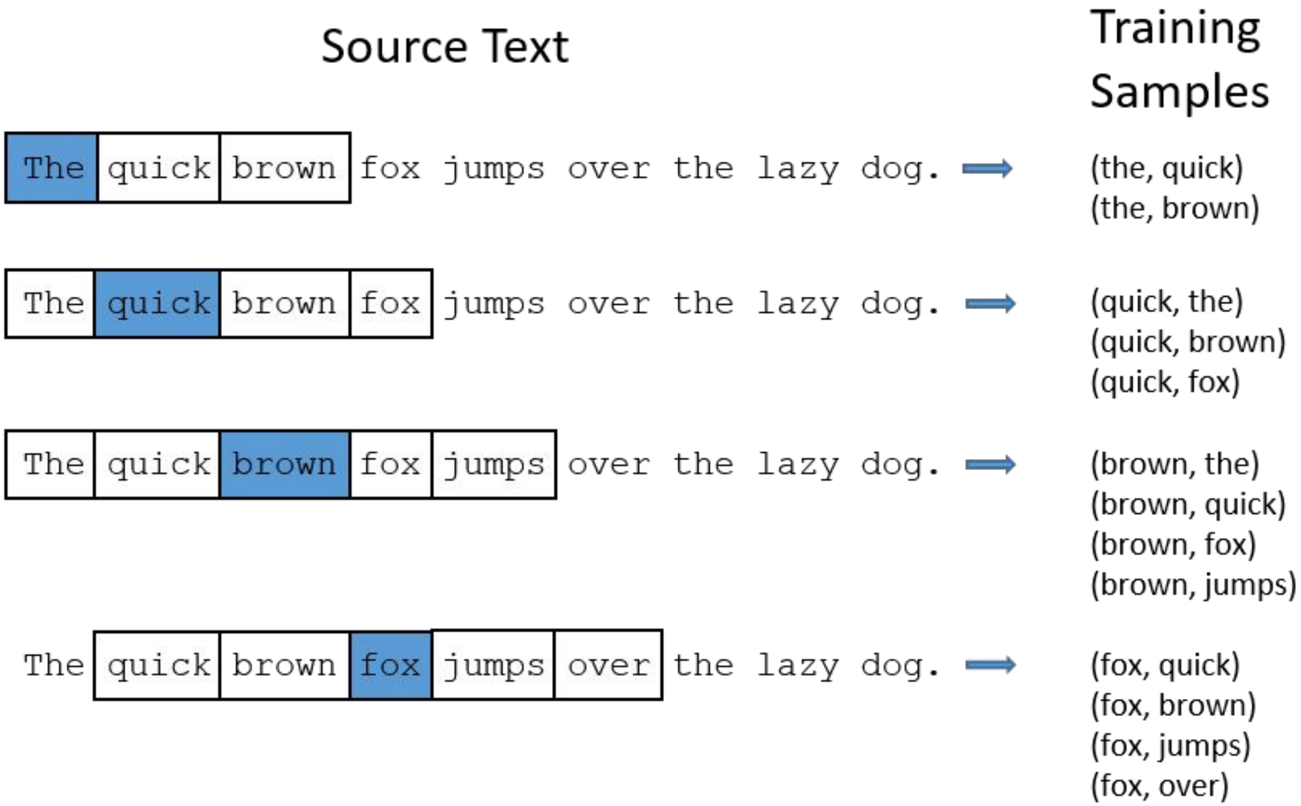
Cambia la unidad

Ahora el corpus es visto como un todo continuo…

No se ven los documentos por separado

Un parámetro importante: el tamaño de la ventana…

Otro metodo: CBOW (al revés)



Skip-gram

Contexte				Mot Cible
The	Quick	Fox	Jump	**Brown**
Quick	Brown	Jumps	Over	**Fox**
Brown	Fox	Over	The	**Jumps**

Skip-gram - Matriz de co-ocurrencias

	brown	dog	fox	jumps	lazy	over	quick	the
brown	0	0	0	0	0	0	1	1
dog	0	0	0	0	1	0	0	1
fox	1	0	0	0	0	0	1	0
jumps	1	0	1	0	0	0	0	0
lazy	0	0	0	0	0	1	0	1
over	0	0	1	1	0	0	0	0
quick	0	0	0	0	0	0	0	1
the	0	0	0	1	0	1	0	0

Skip-gram

Thou shalt not make a machine in the likeness of a human mind

thou	shalt	not	make	a	machine	in	the	…

input word	target word
not	thou
not	shalt
not	make
not	a

http://jalammar.github.io/illustrated-word2vec/ (¡recomendado!)

Skip-gram (otro ejemplo)

Thou shalt not make a machine in the likeness of a human mind

thou	shalt	not	make	a	machine	in	the	…

thou	shalt	not	make	a	machine	in	the	…

input word	target word
not	thou
not	shalt
not	make
not	a
make	shalt
make	not
make	a
make	machine

Skip-gram (otro ejemplo)

Thou shalt not make a machine in the likeness of a human mind

thou	shalt	not	make	a	machine	in	the	…

thou	shalt	not	make	a	machine	in	the	…

thou	shalt	not	make	a	machine	in	the	…

thou	shalt	not	make	a	machine	in	the	…

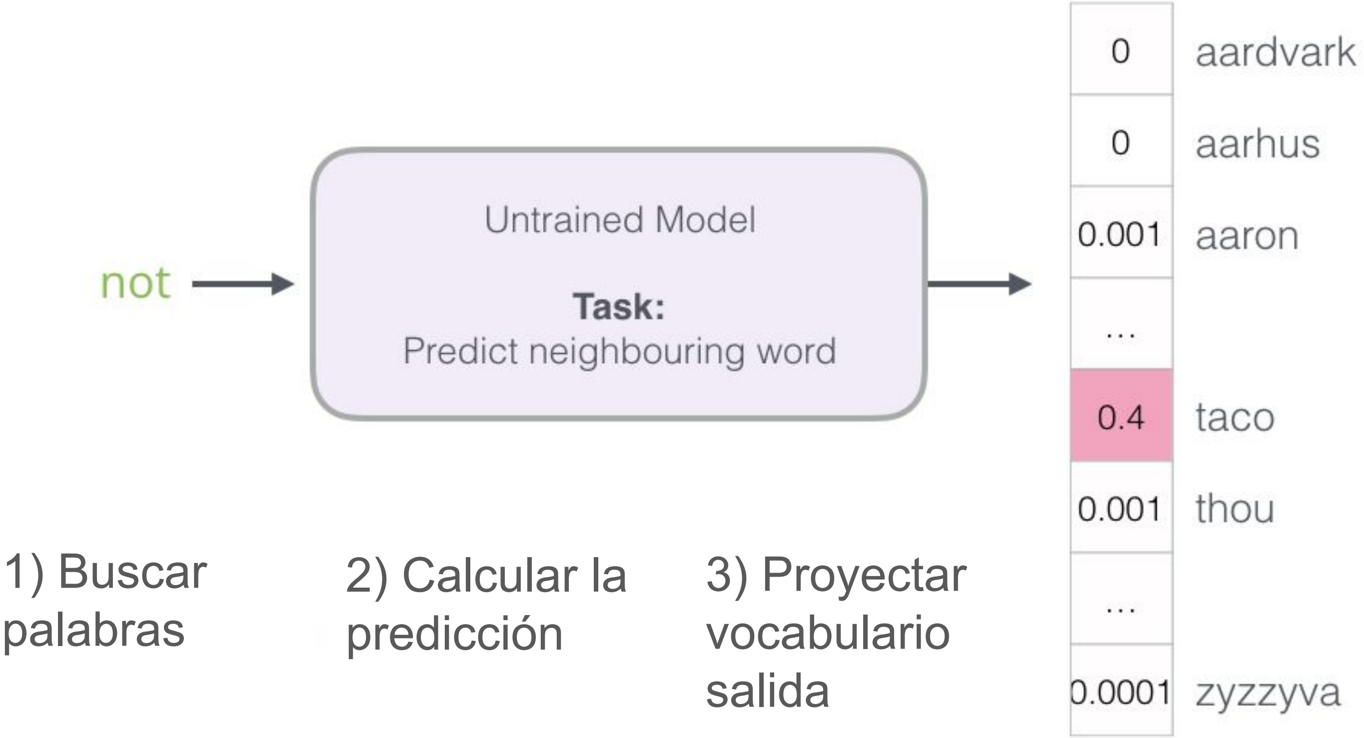
thou	shalt	not	make	a	machine	in	the	…

input word	target word
not	thou
not	shalt
not	make
not	a
make	shalt
make	not
make	a
make	machine
a	not
a	make
a	machine
a	in
machine	make
machine	a
machine	in
machine	the
in	a
in	machine
in	the
in	likeness

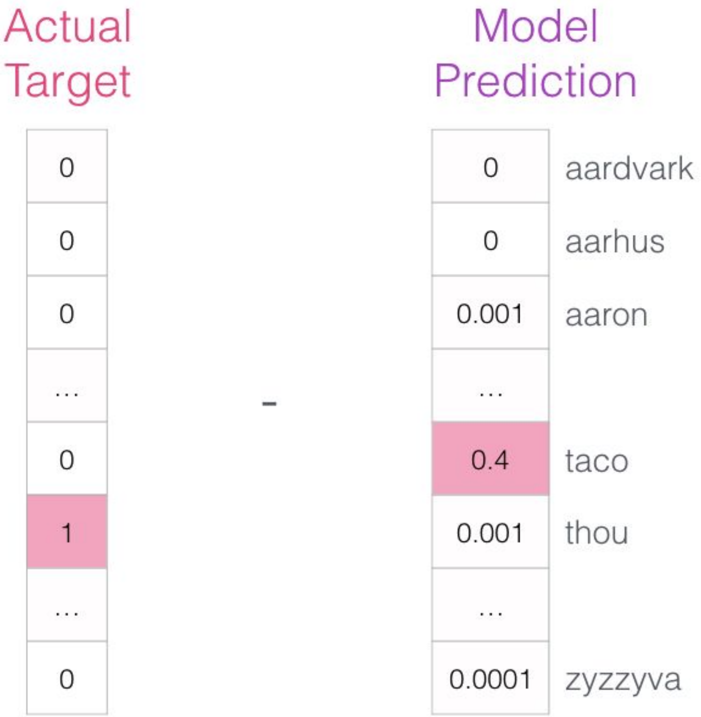
Modelando con skipgram

input word	target word
not	thou
not	shalt
not	make
not	a
make	shalt
make	not
make	a
make	machine
a	not
a	make
a	machine
a	in
machine	make
machine	a
machine	in
machine	the
in	a
in	machine
in	the
in	likeness

Modelando con skipgram



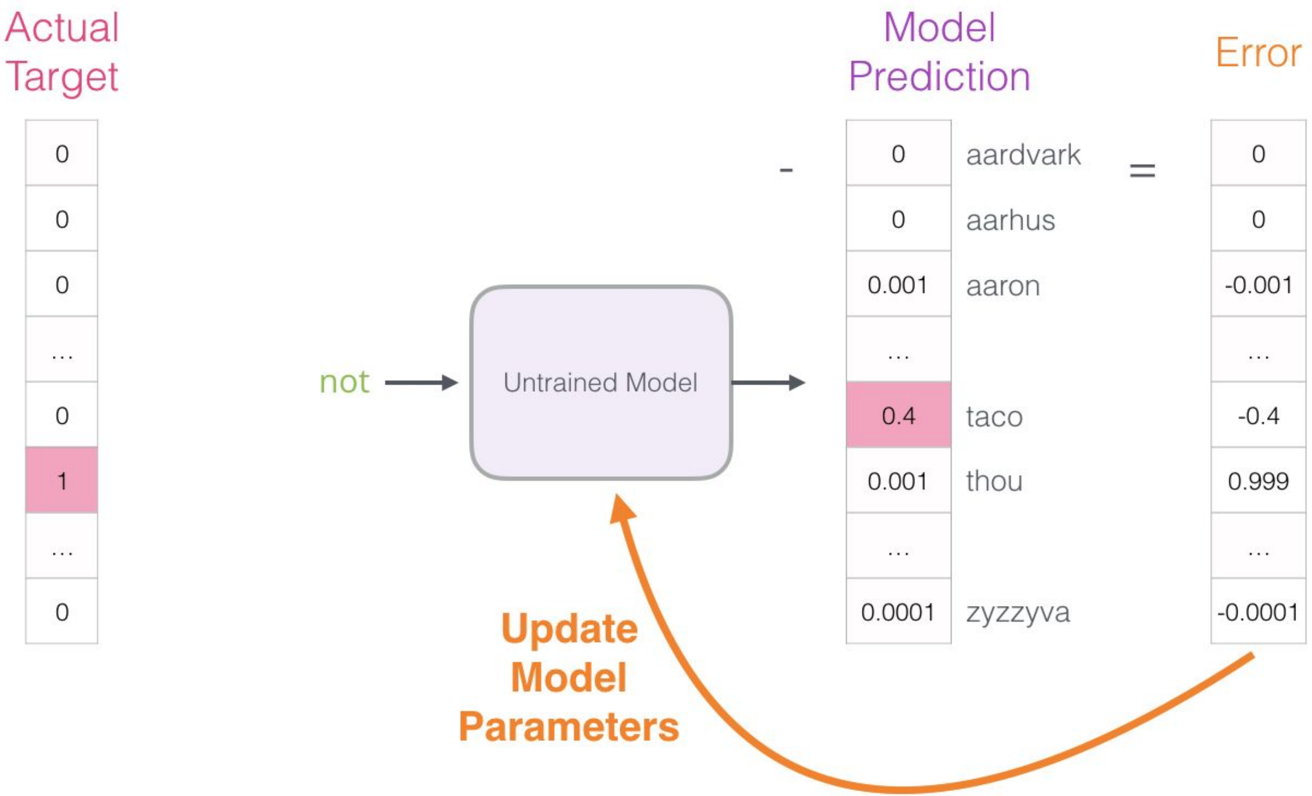
Modelando con skipgram



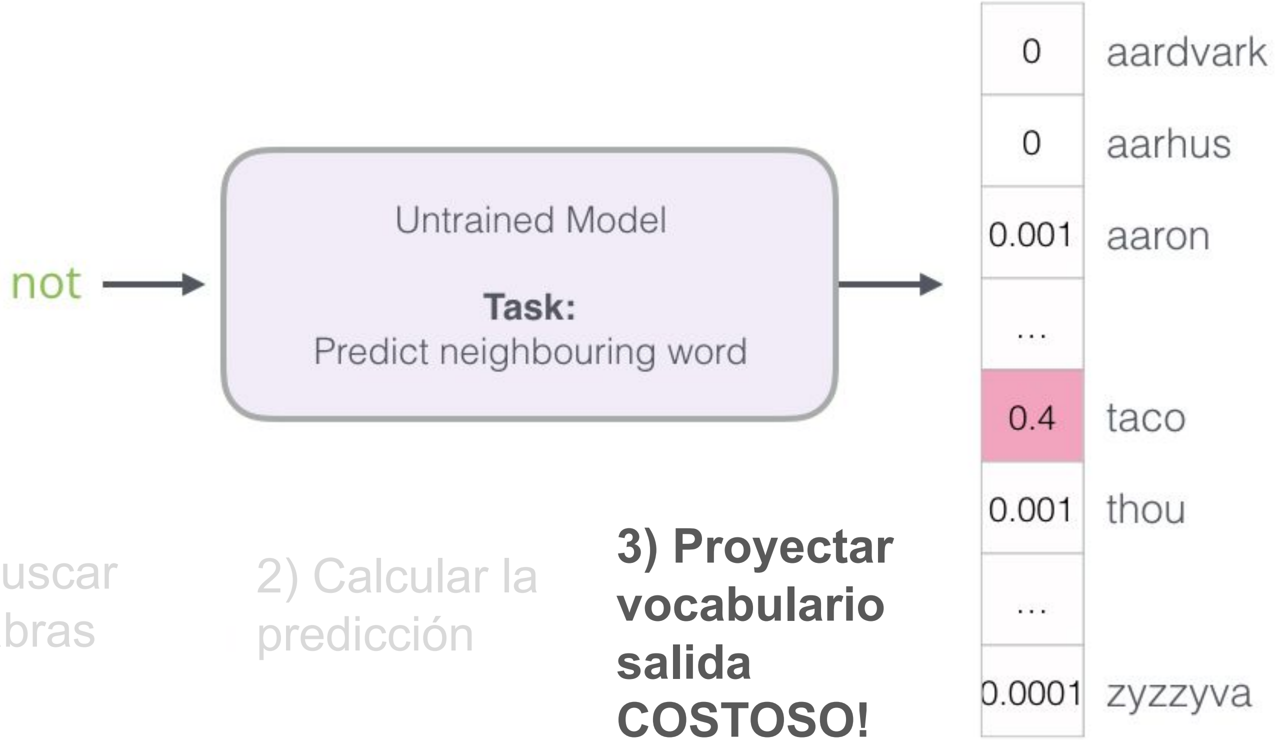
Modelando con skipgram

Actual Target		Model Prediction			Error
0		0	aardvark		0
0		0	aarhus		0
0		0.001	aaron		-0.001
…	-	…		=	…
0		0.4	taco		-0.4
1		0.001	thou		0.999
…		…			…
0		0.0001	zyzzyva		-0.0001

Modelando con skipgram



Modelando con skipgram => PROBLEMA



1) Buscar palabras

2) Calcular la predicción

3) Proyectar vocabulario salida COSTOSO!

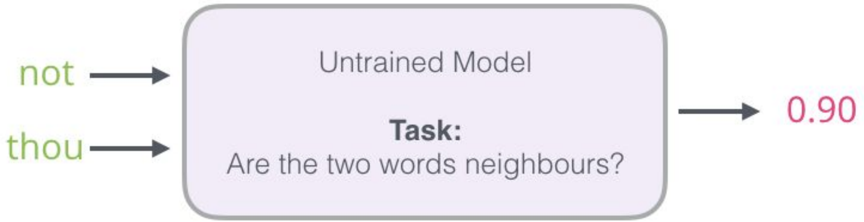
Modelando con skipgram => PROBLEMA

Change Task from

Modelando con skipgram => PROBLEMA

Change Task from

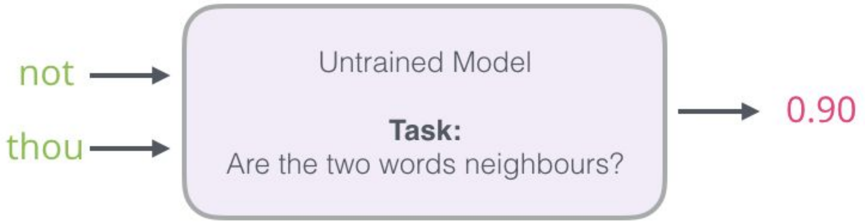
To:



Modelando con skipgram => PROBLEMA

Change Task from

To:



input word	target word
not	thou
not	shalt
not	make
not	a
make	shalt
make	not
make	a
make	machine

input word	output word	target
not	thou	1
not	shalt	1
not	make	1
not	a	1
make	shalt	1
make	not	1
make	a	1
make	machine	1

Problema!
Todos ejemplos positivos…

OVERFITTING

Negative sampling

input word	output word	target
not	thou	1
not		0
not		0
not	shalt	1
not	make	1

Negative examples

factor~data
EIDAES_UNSAM

Negative sampling

input word	output word	target
not	thou	1
not	aaron	0
not	taco	0
not	shalt	1
not	make	1

Pick randomly from vocabulary (random sampling)

Word	Count	Probability
aardvark		
aarhus		
aaron		
taco		
thou		
zyzzyva		

La fórmula mágina de w2vec

Skipgram

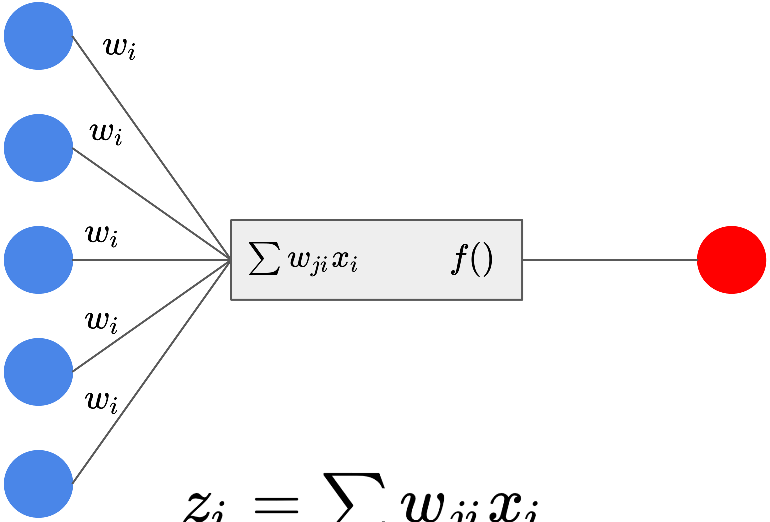
shalt	not	make	a	machine

input	output
make	shalt
make	not
make	a
make	machine

Negative Sampling

input word	output word	target
make	shalt	1
make	aaron	0
make	taco	0

Regresión logística en forma de red neuronal

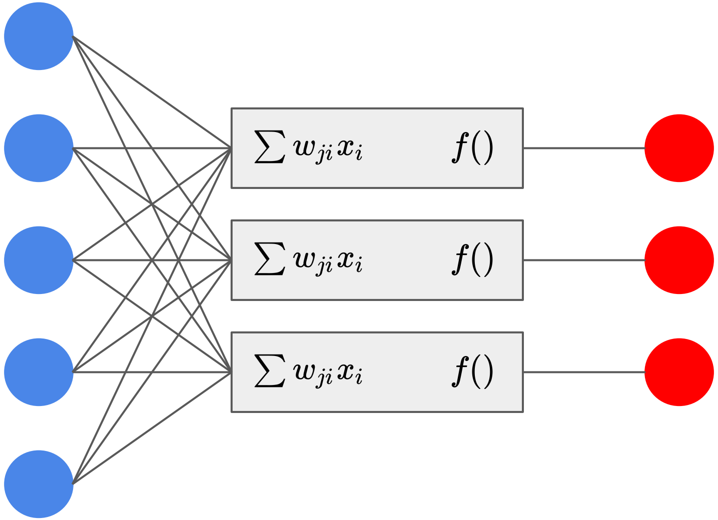


$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_{í}}}$$

$$z_i = \sum w_{ji} x_i$$

$$x_i$$

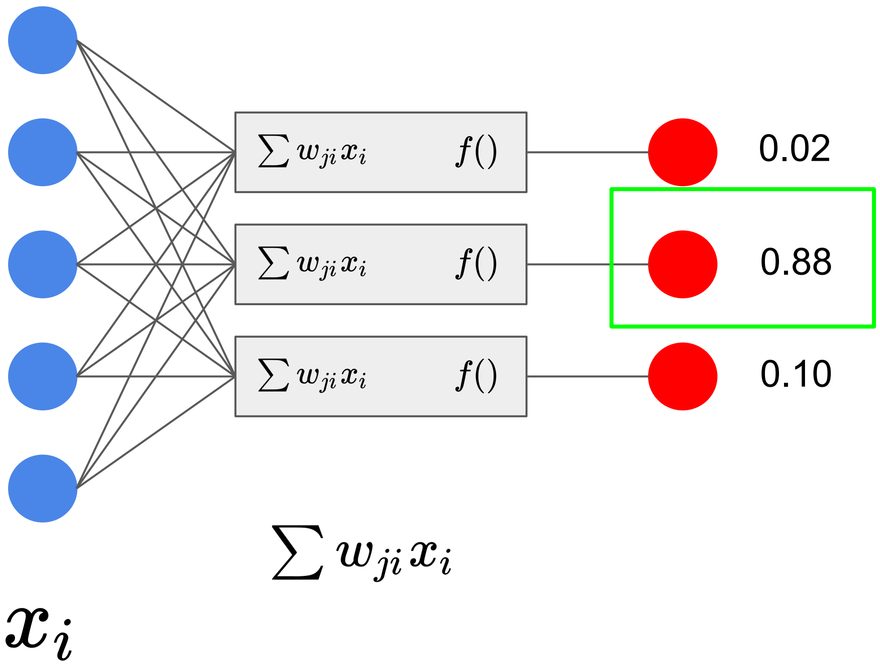
Redes neuronales (intuición)



$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_{\acute{\imath}}}}$$

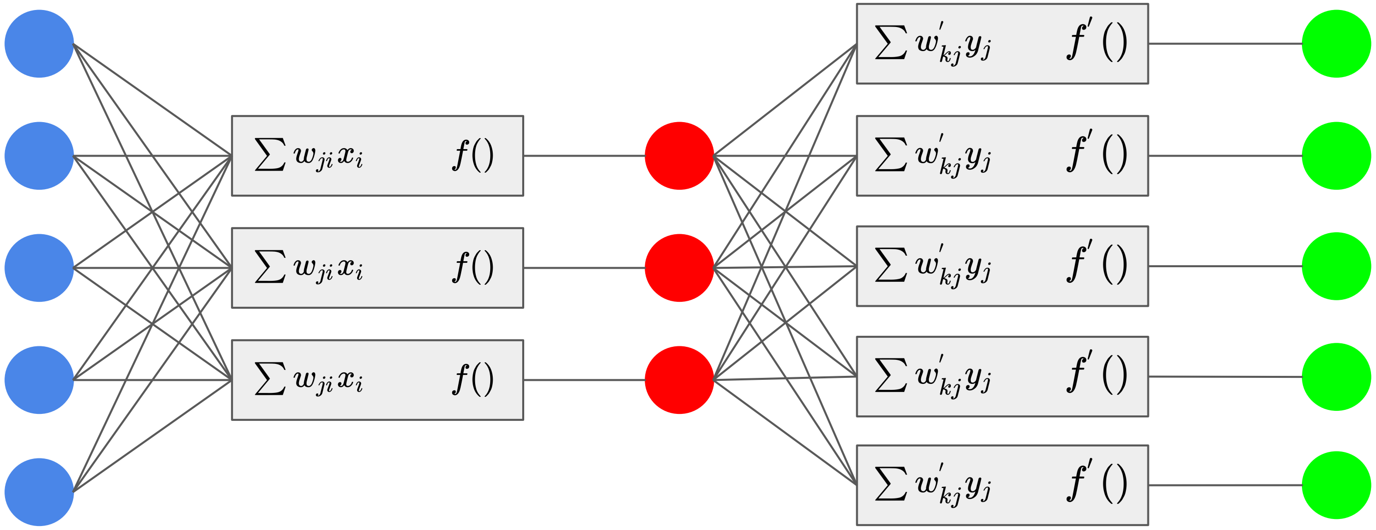
$$z_i = \sum w_{ji} x_i$$

Redes neuronales (intuición)



$$y_i = f(\sum w_{ji}x_i)$$

Ahora sí… word2vec



$\sum w_{ji} x_i$

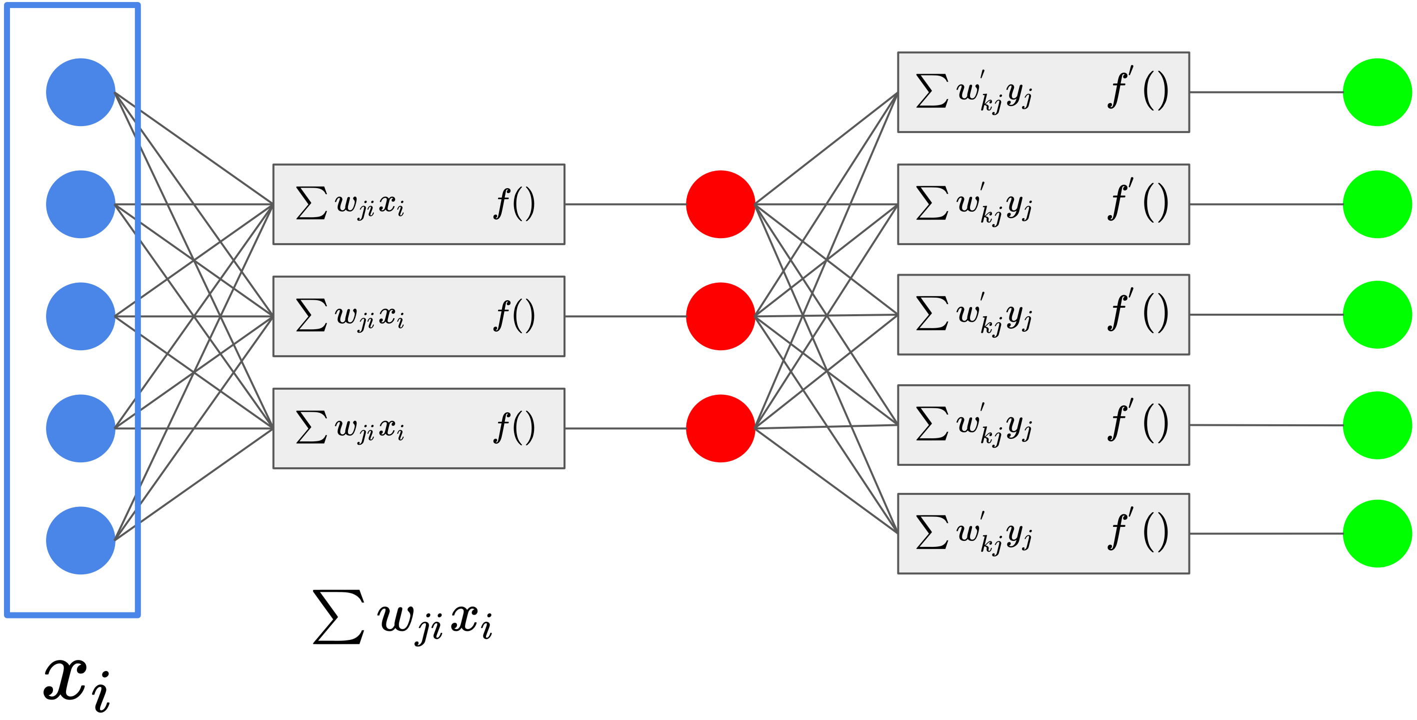
x_i

$$y_i = f(\sum w_{ji} x_i) \qquad z_k = f'(\sum w'_{kj} y_j)$$

Ahora sí… word2vec

Una “unidad” por palabra en el vocabulario => One hot encoded

1 x 5

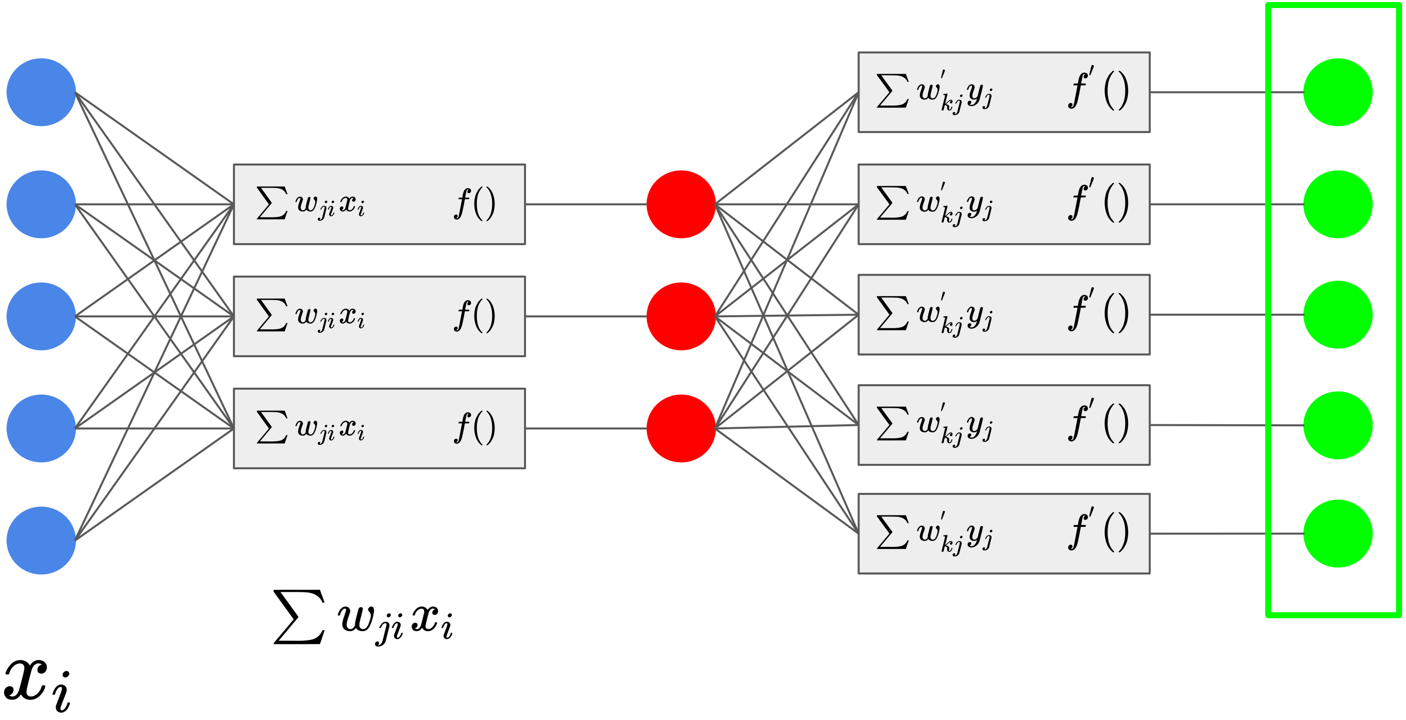


$$y_i = f(\sum w_{ji} x_i) \qquad z_k = f'(\sum w'_{kj} y_j)$$

Ahora sí… word2vec

Una "unidad" por palabra en el vocabulario => One hot encoded 1 x 5

Una "unidad" por palabra en el vocabulario => One hot encoded



$$y_i = f(\sum w_{ji} x_i)$$

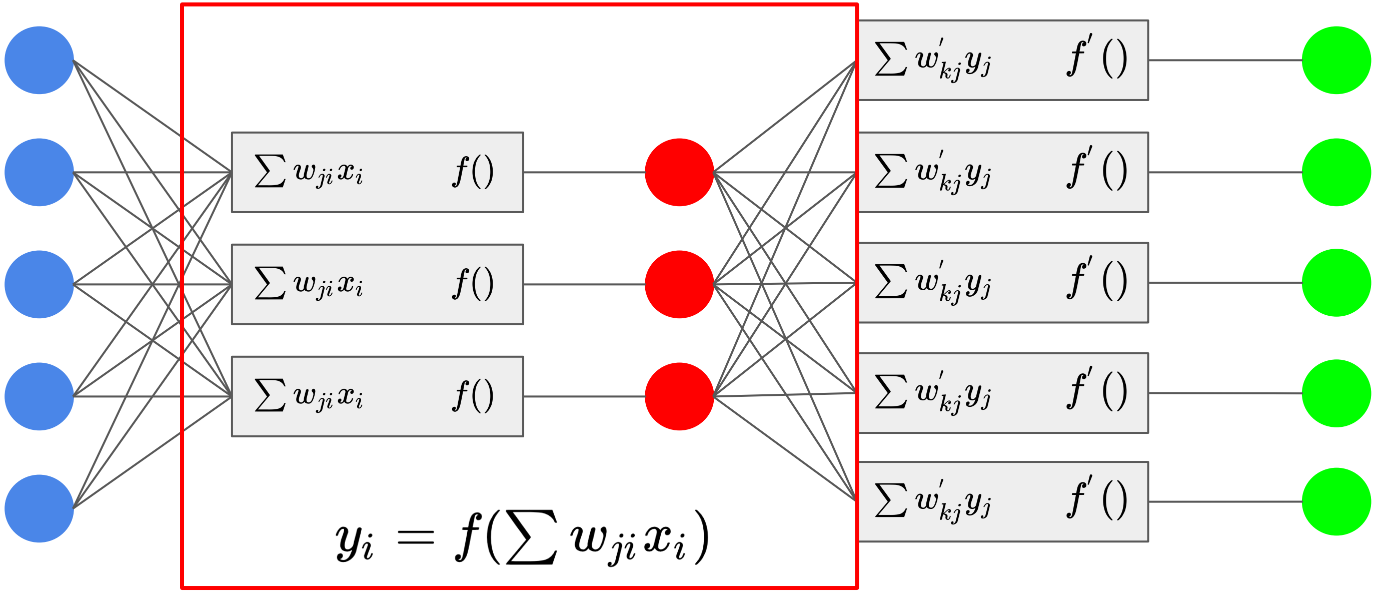
$$z_k = f'(\sum w'_{kj} y_j)$$

factor~data
EIDAES_UNSAM

Ahora sí… word2vec

Este es el **embedding**. Es la representación de baja dimensionalidad de una palabra
1 x 3

Una “unidad” por palabra en el vocabulario => One hot encoded



x_i

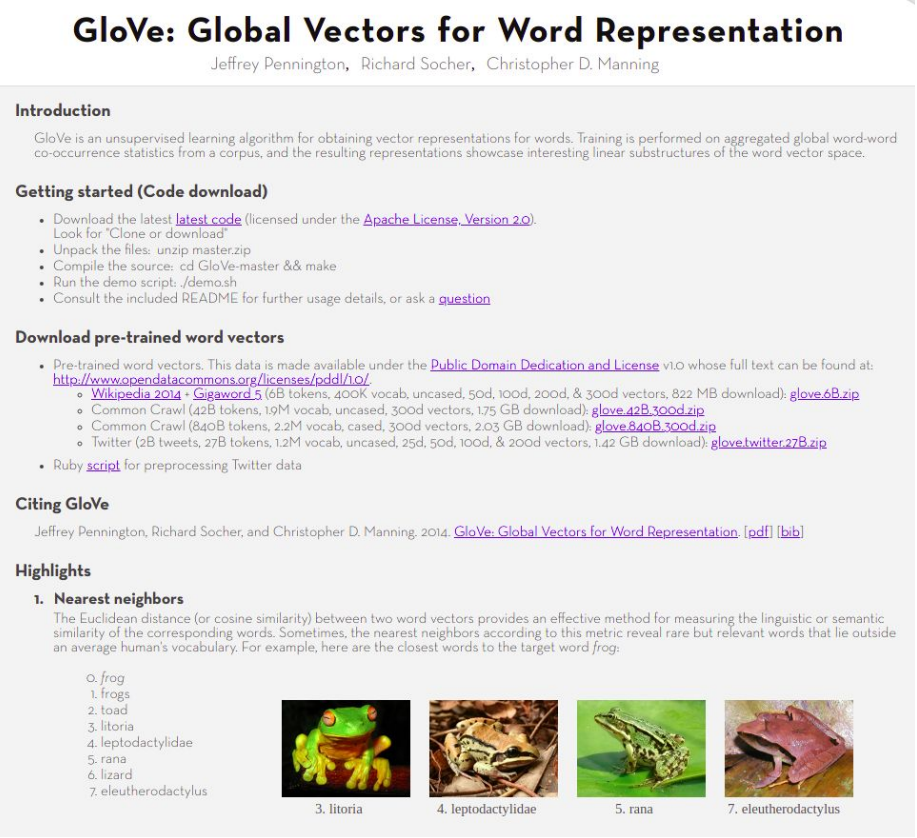
$$z_k = f'(\sum w'_{kj} y_j)$$

factor~data
EIDAES_UNSAM

Otros métodos para construir embeddings

- word2vec fue pionero (2013) pero hoy hay métodos mejores

- GloVe: trabaja directamente sobre la matriz de co-ocurrencias



GloVe: Global Vectors for Word Representation

Jeffrey Pennington, Richard Socher, Christopher D. Manning

Introduction

GloVe is an unsupervised learning algorithm for obtaining vector representations for words. Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations showcase interesting linear substructures of the word vector space.

Getting started (Code download)

- Download the latest latest code (licensed under the Apache License, Version 2.0). Look for "Clone or download"
- Unpack the files: unzip master.zip
- Compile the source: cd GloVe-master && make
- Run the demo script: ./demo.sh
- Consult the included README for further usage details, or ask a question

Download pre-trained word vectors

- Pre-trained word vectors. This data is made available under the Public Domain Dedication and License v1.0 whose full text can be found at: http://www.opendatacommons.org/licenses/pddl/1.0/.
  - Wikipedia 2014 + Gigaword 5 (6B tokens, 400K vocab, uncased, 50d, 100d, 200d, & 300d vectors, 822 MB download): glove.6B.zip
  - Common Crawl (42B tokens, 1.9M vocab, uncased, 300d vectors, 1.75 GB download): glove.42B.300d.zip
  - Common Crawl (840B tokens, 2.2M vocab, cased, 300d vectors, 2.03 GB download): glove.840B.300d.zip
  - Twitter (2B tweets, 27B tokens, 1.2M vocab, uncased, 25d, 50d, 100d, & 200d vectors, 1.42 GB download): glove.twitter.27B.zip
- Ruby script for preprocessing Twitter data

Citing GloVe

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. [pdf] [bib]

Highlights

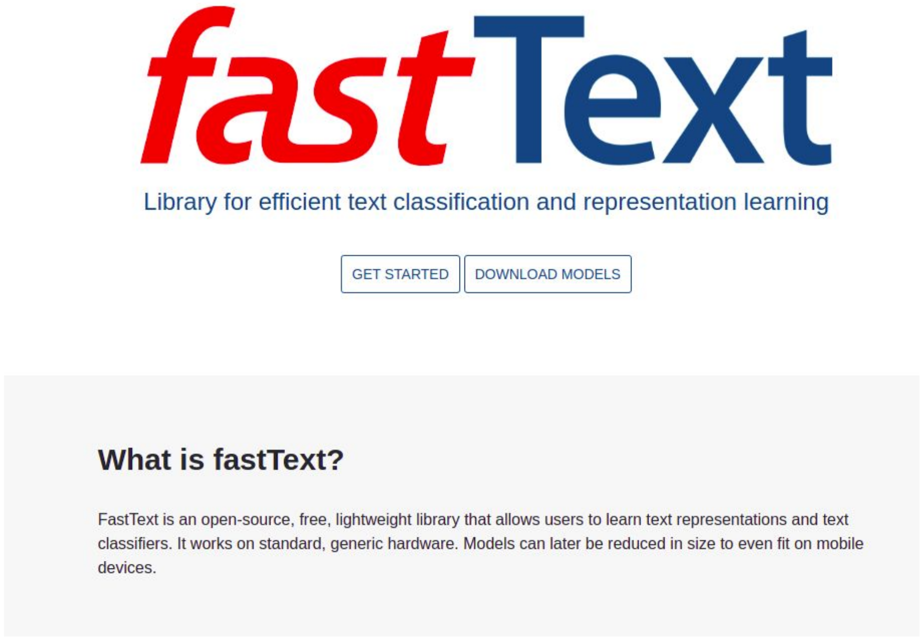
1. **Nearest neighbors**
   The Euclidean distance (or cosine similarity) between two word vectors provides an effective method for measuring the linguistic or semantic similarity of the corresponding words. Sometimes, the nearest neighbors according to this metric reveal rare but relevant words that lie outside an average human's vocabulary. For example, here are the closest words to the target word *frog*:

   0. *frog*
   1. frogs
   2. toad
   3. litoria
   4. leptodactylidae
   5. rana
   6. lizard
   7. eleutherodactylus

3. litoria 4. leptodactylidae 5. rana 7. eleutherodactylus

Otros métodos para construir embeddings

- word2vec fue pionero (2013) pero hoy hay métodos mejore
- GloVe: trabaja directamente sobre la matriz de co-ocurrencias
- FastText: permite un abordaje supervisado y usa algo que se llama “sub n-gramas” => robusto y rápido



Aplicaciones en Ciencias Sociales

npj Schizophrenia

www.nature.com/npjschz

ARTICLE OPEN

Automated analysis of free speech predicts psychosis onset in high-risk youths

Gillinder Bedi[1,2,9], Facundo Carrillo[3,9], Guillermo A Cecchi[4], Diego Fernández Slezak[3], Mariano Sigman[5], Natália B Mota[6], Sidarta Ribeiro[6], Daniel C Javitt[1,7], Mauro Copelli[8] and Cheryl M Corcoran[1,7]

BACKGROUND/OBJECTIVES: Psychiatry lacks the objective clinical tests routinely used in other specializations. Novel computerized methods to characterize complex behaviors such as speech could be used to identify and predict psychiatric illness in individuals.

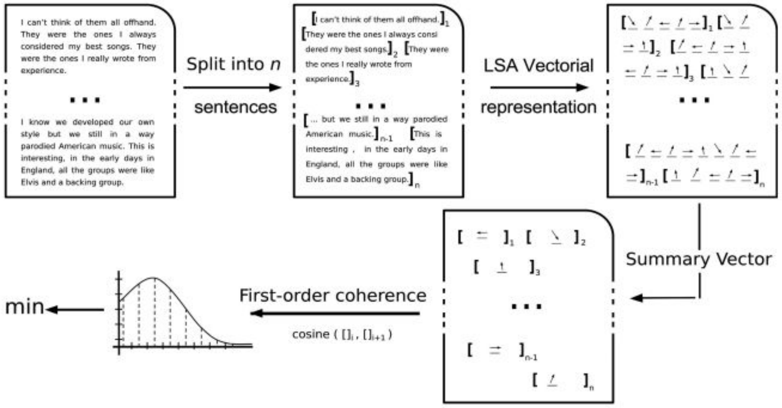
AIMS: In this proof-of-principle study, our aim was to test automated speech analyses combined with Machine Learning to predict later psychosis onset in youths at clinical high-risk (CHR) for psychosis.

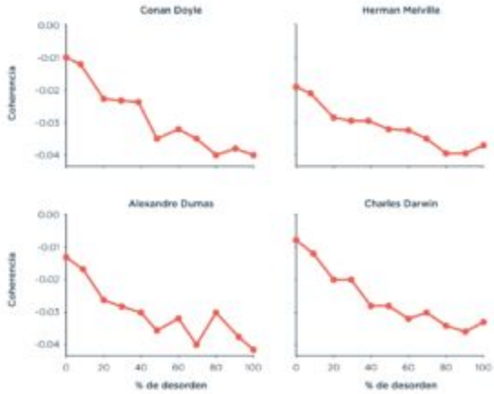
METHODS: Thirty-four CHR youths (11 females) had baseline interviews and were assessed quarterly for up to 2.5 years; five transitioned to psychosis. Using automated analysis, transcripts of interviews were evaluated for semantic and syntactic features predicting later psychosis onset. Speech features were fed into a convex hull classification algorithm with leave-one-subject-out cross-validation to assess their predictive value for psychosis outcome. The canonical correlation between the speech features and prodromal symptom ratings was computed.

RESULTS: Derived speech features included a Latent Semantic Analysis measure of semantic coherence and two syntactic markers of speech complexity: maximum phrase length and use of determiners (e.g., *which*). These speech features predicted later psychosis development with 100% accuracy, outperforming classification from clinical interviews. Speech features were significantly correlated with prodromal symptoms.

CONCLUSIONS: Findings support the utility of automated speech analysis to measure subtle, clinically relevant mental state changes in emergent psychosis. Recent developments in computer science, including natural language processing, could provide the foundation for future development of objective clinical tests for psychiatry.

npj Schizophrenia (2015) **1**, Article number: 15030; doi:10.1038/npjschz.2015.30; published online 26 August 2015





factor~data
EIDAES_UNSAM

Aplicaciones en Ciencias Sociales - Estereotipos

Semantics derived automatically from language corpora contain human-like biases

Aylin Caliskan,[1*] Joanna J. Bryson,[1,2*] Arvind Narayanan[1*]

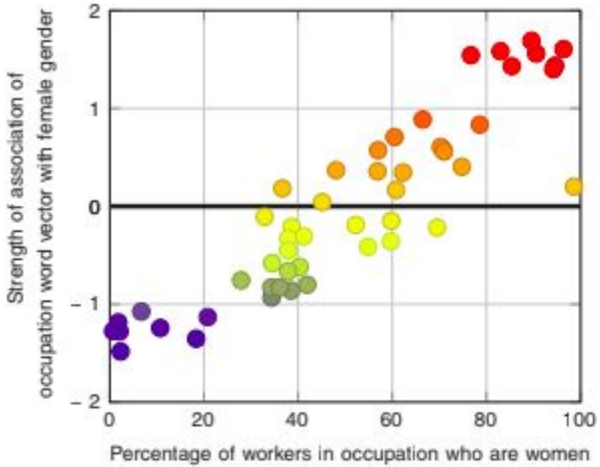


Fig. 1. Occupation-gender association. Pearson's correlation coefficient $\rho = 0.90$ with $P < 10^{-18}$.

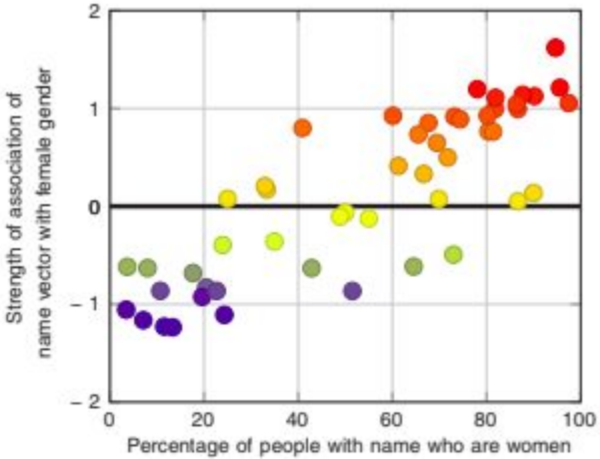


Fig. 2. Name-gender association. Pearson's correlation coefficient $\rho = 0.84$ with $P < 10^{-13}$.

Aplicaciones en Ciencias Sociales - Estereotipos

American Sociological Review
2019, Vol. 84(5) 905–949

DOI: 10.1177/0003122419877135
journals.sagepub.com/home/asr
SAGE

The Geometry of Culture: Analyzing the Meanings of Class through Word Embeddings

Austin C. Kozlowski,[a] Matt Taddy,[b]
and James A. Evans[a,c]

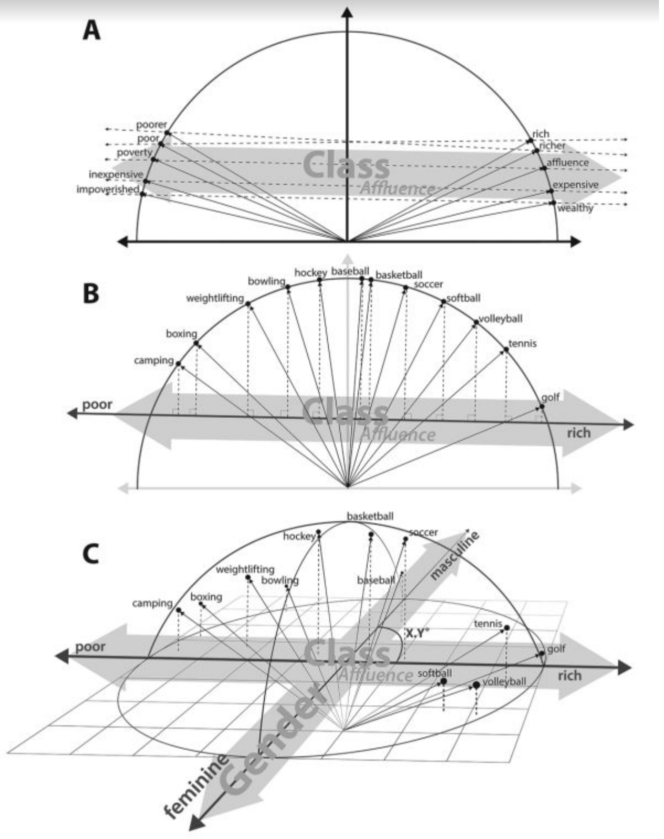


Figure 2. Conceptual Diagram of (A) the Construction of a Cultural Dimension; (B) the Projection of Words onto That Dimension; and (C) the Simultaneous Projection of Words onto Multiple Dimensions

factor~data
EIDAES_UNSAM