Estudio sobre Sistemas de Recomendación y Predicción basados en el procesamiento del lenguaje natural

Hugo Ferrando Seage

7 de diciembre de 2017

Universidad Europea de Madrid
Escuela de Arquitectura, Ingeniería y Diseño

Introducción

Introducción

Los recomendadores son una parte esencial de cualquier servicio de Video on Demand y de otros sectores.

- Netflix
- Movistar+
- Amazon
- Hulu
- HBO
- IMDb
- FilmAffinity
- Jinni

Introducción

Existen tres grandes tipos de sistemas de recomendación:

- Filtrado colaborativo
- Filtrado por contenido
- Sistemas híbridos

Filtrado Colaborativo

Consiste en emparejar usuarios que tengan gustos similares y recomendar en base a esos datos.

Los usuarios deben puntuar los contenidos, o se pueden usar otras métricas.

Se puede visualizar usando una matriz donde las filas representan usuarios y la columnas representan productos.

Filtrado por Contenido

Consiste en la creación de un modelo que determina la similitud entre productos en base a algún criterio.

Ese criterio puede ser cualquier elemento del producto. Para películas puede ser el género. Para restaurantes el tipo de cocina. Etc.

Filtrado Híbrido

Usan una combinación de ambas técnicas para complementar las recomendaciones.

Objetivos

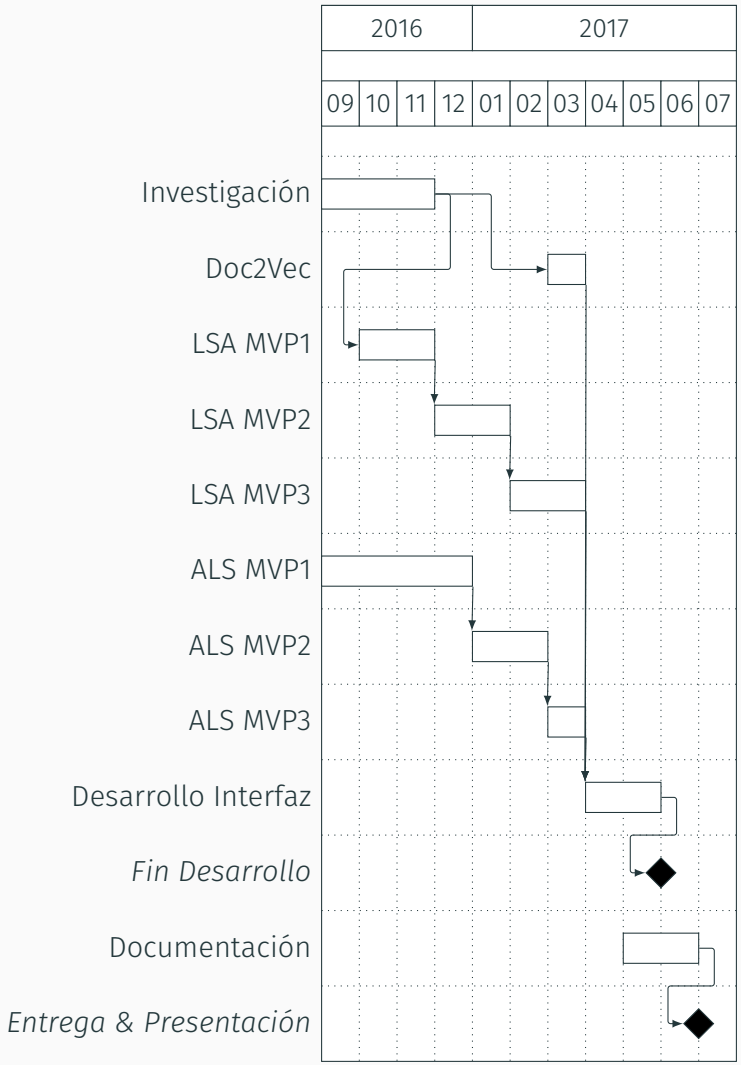
Objetivos

- Construir un recomendador de películas
- Crear el modelo en base a tres algoritmos
  - LSA
  - Doc2Vec
  - E-Modelo
- Optimizar modelos
- Crear una interfaz desde donde poder probarlos

Metodología

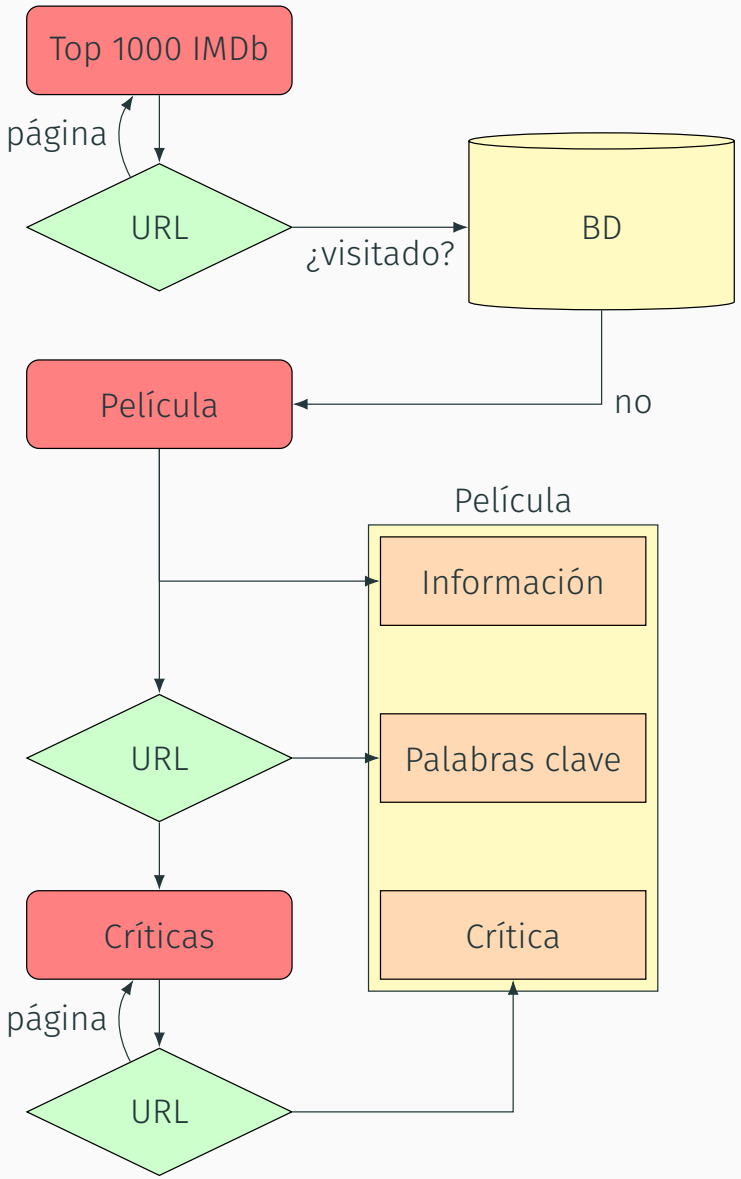
Metodología

La metodología usada ha sido ágil, basada en MVPs.



Descarga de datos

Descarga de datos



Limpieza de textos

Limpieza de textos

```
Zeus is a Greek God.
```

NNP	VBZ	DT	NN	NN
Zeus	is	a	Greek	God.

Limpieza de textos

```
Zeus is a country deity. (Hiperónimos)

is a country deity. (Nombres)

country deity (Stopwords)

counti deiti (Stemmer)
```

LSA

LSA

Latent Semantic Analysis trata de extraer conceptos de cada texto y analizar la relación entre documentos.

TF-IDF

$$
tfidf = \begin{array}{r} \\ \text{The Matrix} \\ \text{Alien} \\ \text{Serenity} \\ \text{Casablanca} \\ \text{Amelie} \end{array}
\begin{array}{c}
\begin{array}{ccccccc} \text{says} & \text{just} & \text{room} & \text{dead} & \text{asks} & \text{ship} & \text{mother} \end{array} \\
\begin{pmatrix}
0{,}39 & 0{,}16 & 0{,}19 & 0{,}01 & 0{,}25 & 0{,}79 & 0{,}27 \\
0{,}12 & 0{,}12 & 0{,}06 & 0{,}46 & 0{,}21 & 0{,}07 & 0{,}83 \\
0{,}46 & 0{,}55 & 0{,}15 & 0{,}55 & 0{,}22 & 0{,}27 & 0{,}11 \\
0{,}00 & 0{,}60 & 0{,}51 & 0{,}00 & 0{,}00 & 0{,}60 & 0{,}00 \\
0{,}41 & 0{,}00 & 0{,}35 & 0{,}83 & 0{,}00 & 0{,}00 & 0{,}00
\end{pmatrix}
\end{array}
$$

SVD

$$V^T = \begin{array}{r} \\ \text{Sci-Fi topic} \\ \\ \text{Romance topic} \end{array} \begin{array}{c} \begin{array}{ccccc} \text{Matrix} & \text{Alien} & \text{Serenity} & \text{Casablanca} & \text{Amelie} \end{array} \\ \begin{pmatrix} 0{,}56 & 0{,}59 & 0{,}56 & 0{,}09 & 0{,}09 \\ 0{,}12 & -0{,}02 & 0{,}12 & -0{,}69 & -0{,}69 \end{pmatrix} \end{array}$$

Similitud Coseno

$$\cos\left(\begin{pmatrix} 0{,}56 \\ \\ 0{,}12 \end{pmatrix}, \begin{pmatrix} 0{,}59 \\ \\ -0{,}02 \end{pmatrix}\right) = 0{,}97 \tag{1}$$

Figura 1: Alta similitud entre Matrix y Alien

$$\cos\left(\begin{pmatrix} 0{,}56 \\ \\ 0{,}12 \end{pmatrix}, \begin{pmatrix} 0{,}09 \\ \\ -0{,}69 \end{pmatrix}\right) = -0{,}08 \tag{2}$$

Figura 2: Baja similitud entre Matrix y Amelie

Doc2Vec

Word2Vec

Word2Vec es un algoritmo creado por Google en 2013. Es conceptualmente similar a LSA, pero teniendo en cuenta cada palabra dentro de su contexto.

Es decir, calcula la probabilidad de que una palabra esté en la vecindad de otra palabra en el vocabulario.

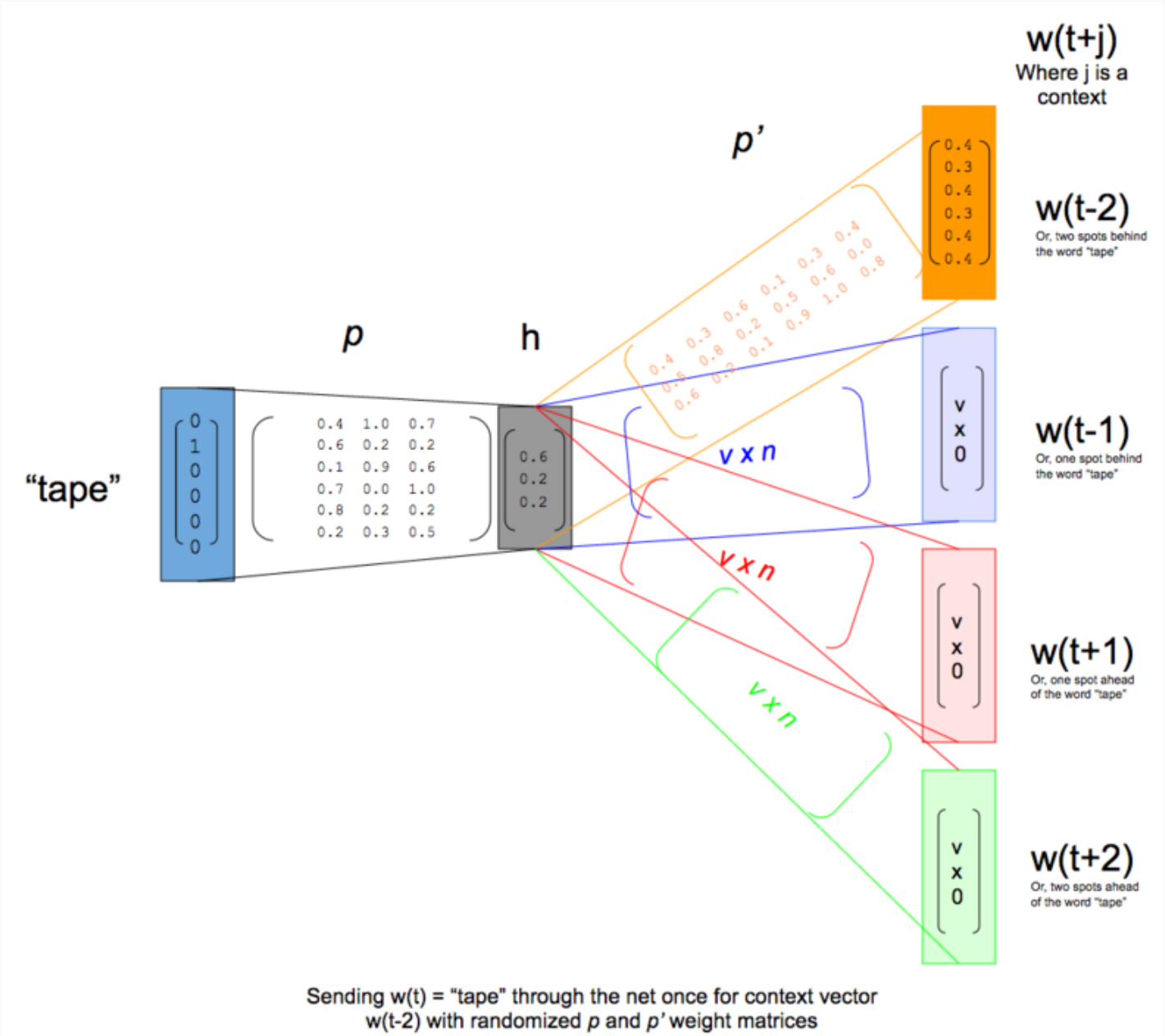
Word2Vec

Source Text	Training Samples
The quick brown fox jumps over the lazy dog	(‘the’, ‘quick’) (‘the’, ‘brown’)
The quick brown fox jumps over the lazy dog	(‘quick’, ‘the’) (‘quick’, ‘brown’) (‘quick’, ‘fox’)
The quick brown fox jumps over the lazy dog	(‘brown’, ‘the’) (‘brown’, ‘quick’) (‘brown’, ‘fox’) (‘brown’, ‘jumps’)
The quick brown fox jumps over the lazy dog	(‘fox’, ‘quick’) (‘fox’, ‘brown’) (‘fox’, ‘jumps’) (‘fox’, ‘over’)

Word2Vec

Palabra	Posición por orden alfabético	Vector
fox	2/3	[0, 1, 0]
dog	1/3	[1, 0, 0]
zebra	3/3	[0, 0, 1]

Word2Vec



Sending w(t) = "tape" through the net once for context vector w(t-2) with randomized *p* and *p'* weight matrices

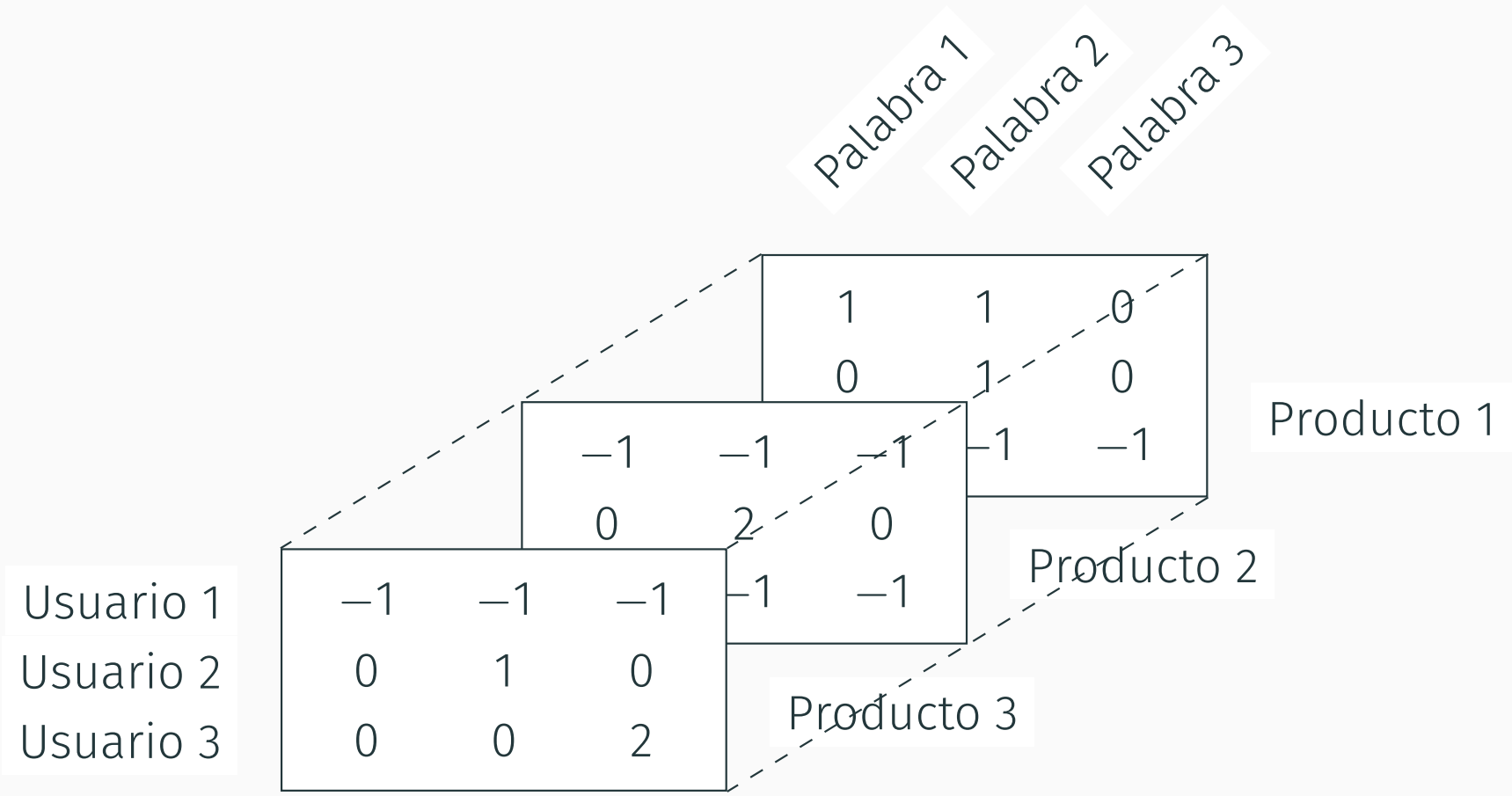
E-Modelo

E-Modelo

Modelo de predicción de frecuencia de uso de palabras híbrido.

Combina el filtrado colaborativo con los features extraidos de un filtrado por contenido.

E-Modelo


Palabra 1
Palabra 2
Palabra 3
Producto 1
Producto 2
Producto 3
Usuario 1
Usuario 2
Usuario 3
1 1 0
0 1 0
−1 −1
−1 −1 −1
0 2 0
−1 −1
−1 −1 −1
0 1 0
0 0 2

E-Modelo

	Producto 1			Producto 2			Producto 3		
	Palabra 1	Palabra 2	Palabra 3	Palabra 1	Palabra 2	Palabra 3	Palabra 1	Palabra 2	Palabra 3
Usuario 1	1	1	0	−1	−1	−1	−1	−1	−1
Usuario 2	0	1	0	0	2	0	0	1	0
Usuario 3	−1	−1	−1	−1	−1	−1	0	0	2

40 % de precisión

Optimización

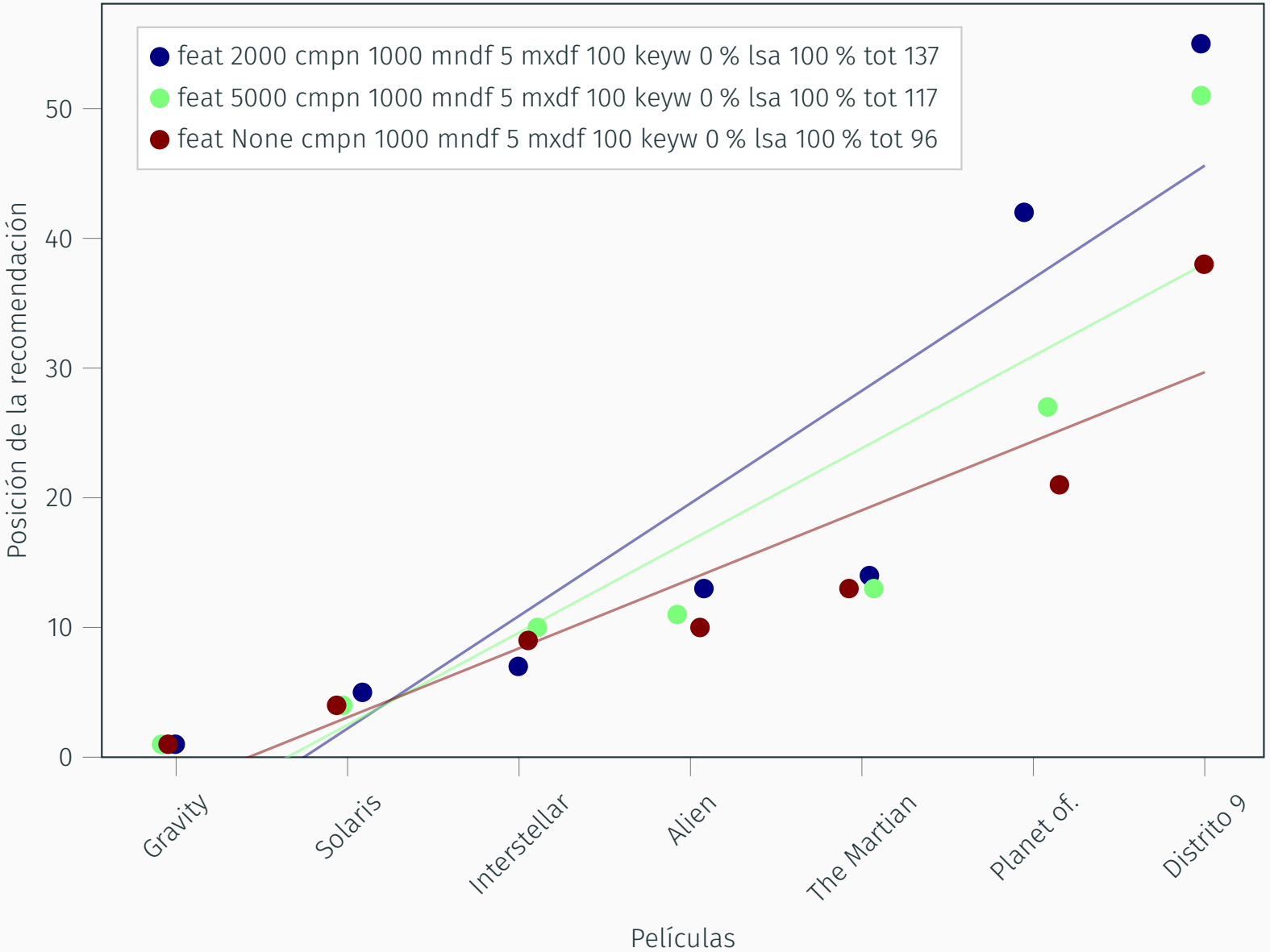
Parámetros LSA

- Número de ‘features’ TF-IDF
- Número de componentes LSA
- Frecuencia Mínima de Documentos
- Frecuencia Máxima de Documentos

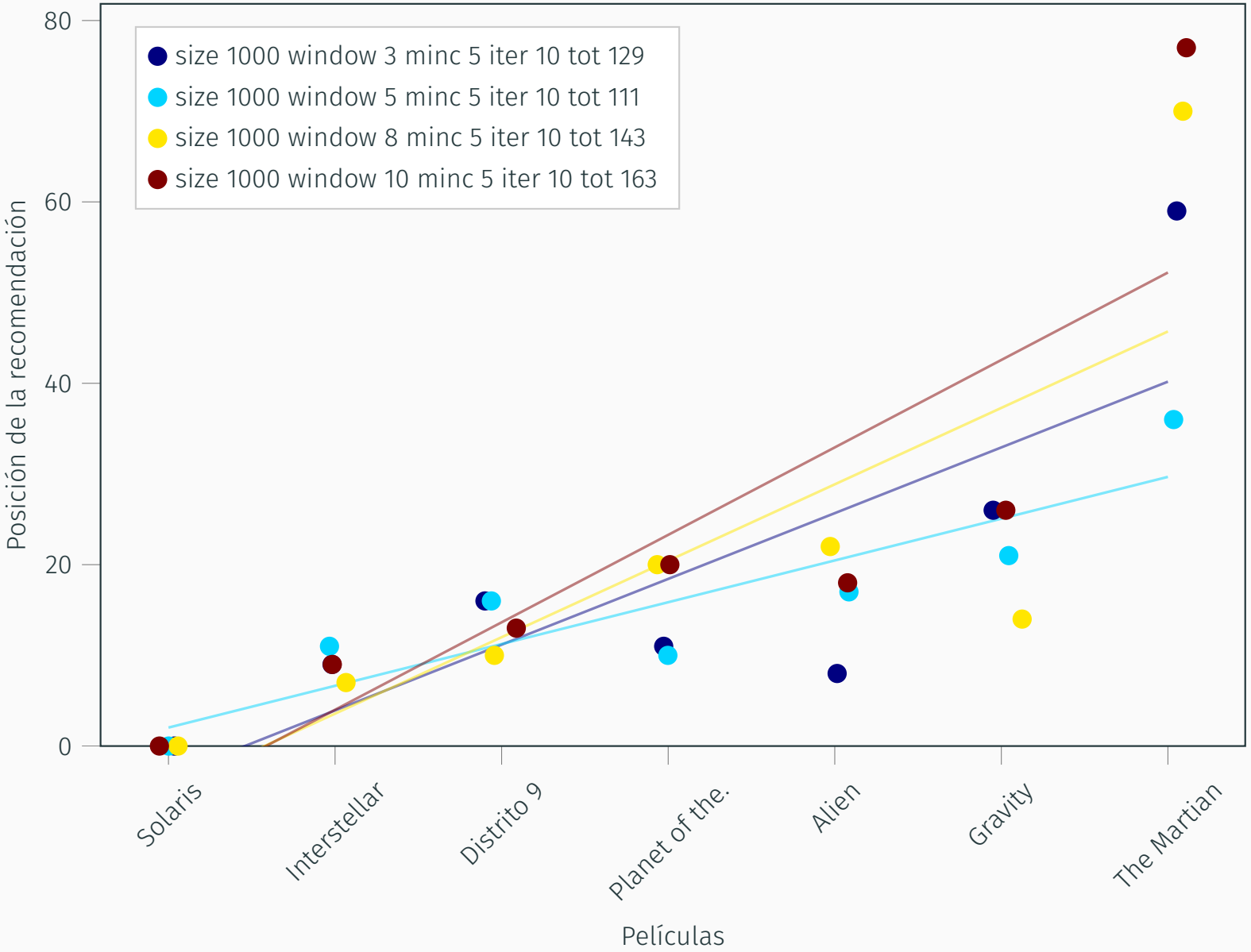
Parámetros Doc2Vec

- Size
- Window
- Minimum Word Count
- Iteraciones

Optimización LSA 2001: A Space Odyssey



Optimización Doc2Vec 2001: A Space Odyssey



Demo

Demo

https://moviepepper.hugofs.com