
THE BRIDGE
Embeddings
Introducción

Embeddings

- Representación compacta de un conjunto de datos de mayor dimensionalidad/cardinalidad pero dispersos

Embeddings

- Representación compacta de un conjunto de datos de mayor dimensionalidad/cardinalidad pero dispersos

Embeddings

- Representación compacta de un conjunto de datos de mayor dimensionalidad/cardinalidad pero dispersos

- Una *feature* categórica con 50(*) valores posibles

(*) 51 si consideramos Washington

Embeddings

- Representación compacta de un conjunto de datos de mayor dimensionalidad/cardinalidad pero dispersos

- Una feature categórica con 50 valores posibles
- Codificarla con one-hot encoding daría un vector de 50 dimensiones muy disperso (un1 y 49 ceros por capital)

Embeddings

- Representación compacta de un conjunto de datos de mayor dimensionalidad/cardinalidad pero dispersos

- Una feature categórica con 50 valores posibles
- Codificarla con one-hot encoding daría un vector de 50 dimensiones muy disperso (un1 y 49 ceros por capital)
- Aplicado a la población de USA (aprox. 341 millones) tendríamos un dataset con solo un 2% de valores (1's) y 16709 millones de ceros

Embeddings

- Representación **compacta** de un conjunto de datos **de mayor dimensionalidad/cardinalidad** pero dispersos

- Una feature categórica con 50 valores posibles
- Codificarla con one-hot encoding daría un vector de 50 dimensiones muy disperso (un 1 y 49 ceros por capital)
- Aplicado a la población de USA (aprox. 341 millones) tendríamos un dataset con solo un 2% de valores (1's) y 16709 millones de ceros

Embeddings

- Representación **compacta** de un conjunto de datos **de mayor dimensionalidad/cardinalidad** pero dispersos

- Una feature categórica con 50 valores posibles
- Codificarla con one-hot encoding daría un vector de 50 dimensiones muy disperso (un 1 y 49 ceros por capital)
- Aplicado a la población de USA (aprox. 341 millones) tendríamos un dataset con solo un 2% de valores (1's) y 16709 millones de ceros

- Por un lado, podríamos hacer un ordinal encoder -> Un vector de una única dimensión (50 valores) -> Problema con las "distancias"

Embeddings

- Representación **compacta** de un conjunto de datos **de mayor dimensionalidad/cardinalidad** pero dispersos

- Una feature categórica con 50 valores posibles
- Codificarla con one-hot encoding daría un vector de 50 dimensiones muy disperso (un 1 y 49 ceros por capital)
- Aplicado a la población de USA (aprox. 341 millones) tendríamos un dataset con solo un 2% de valores (1's) y 16709 millones de ceros

- Por un lado, podríamos hacer un ordinal encoder -> Un vector de una única dimensión (50 valores) -> Problema con las "distancias"
- La latitud y la longitud

Embeddings

- Representación **compacta** de un conjunto de datos **de mayor dimensionalidad/cardinalidad** pero dispersos

- Una feat[illegible]ategórica
- Codifi[illegible] de 50 dim[illegible]pital)
- Ap[illegible]s) tena[illegible]mos u[illegible]olo[illegible]s (1's) y 16709 [illegible]

- Por un lado, podríamos hacer un ordinal encoder -> Un vector de una única dimensión (50 valores) -> Problema con las "distancias"
- La latitud y la longitud
- Pero un embedding se aplica a cualquier "categórica", entonces… ¿para modelos de coches del mundo?

Embeddings

- Representación **compacta** de un conjunto de datos **de mayor dimensionalidad/cardinalidad** pero dispersos

- Para esos casos lo que se hace es entrenar una capa de Embeddings

Embeddings

- Representación **compacta** de un conjunto de datos **de mayor dimensionalidad/cardinalidad** pero dispersos

- Para esos casos lo que se hace es entrenar una capa de Embeddings

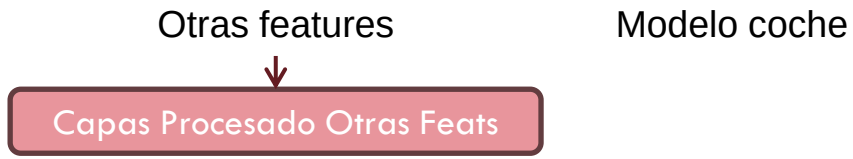
Otras features

Modelo coche

Embeddings

- Representación **compacta** de un conjunto de datos **de mayor dimensionalidad/cardinalidad** pero dispersos

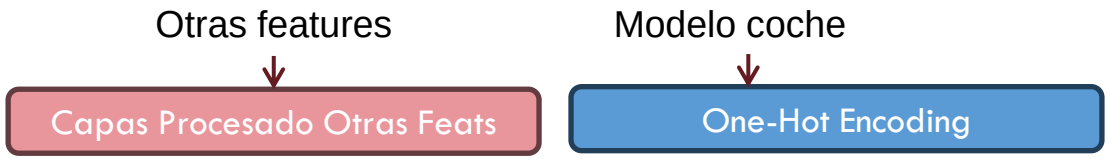
- Para esos casos lo que se hace es entrenar una capa de Embeddings



Embeddings

- Representación **compacta** de un conjunto de datos **de mayor dimensionalidad/cardinalidad** pero dispersos

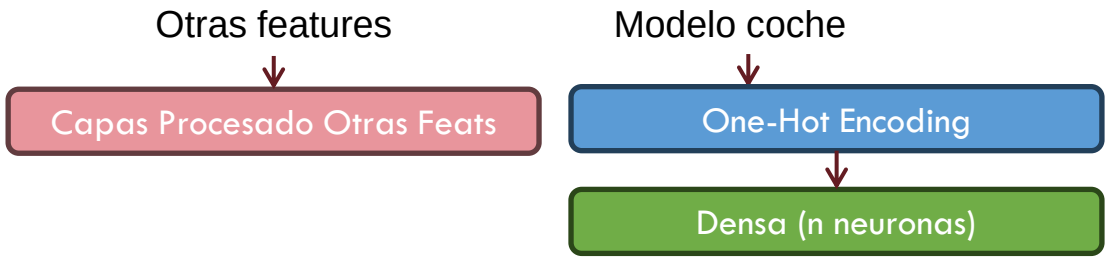
- Para esos casos lo que se hace es entrenar una capa de Embeddings



Embeddings

- Representación **compacta** de un conjunto de datos **de mayor dimensionalidad/cardinalidad** pero dispersos

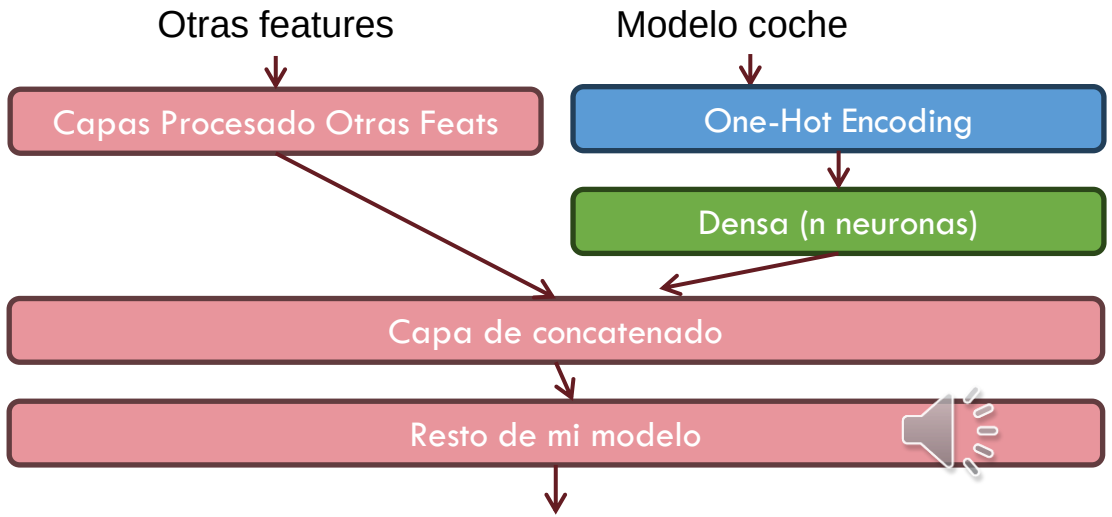
- Para esos casos lo que se hace es entrenar una capa de Embeddings



Embeddings

- Representación **compacta** de un conjunto de datos **de mayor dimensionalidad/cardinalidad** pero dispersos

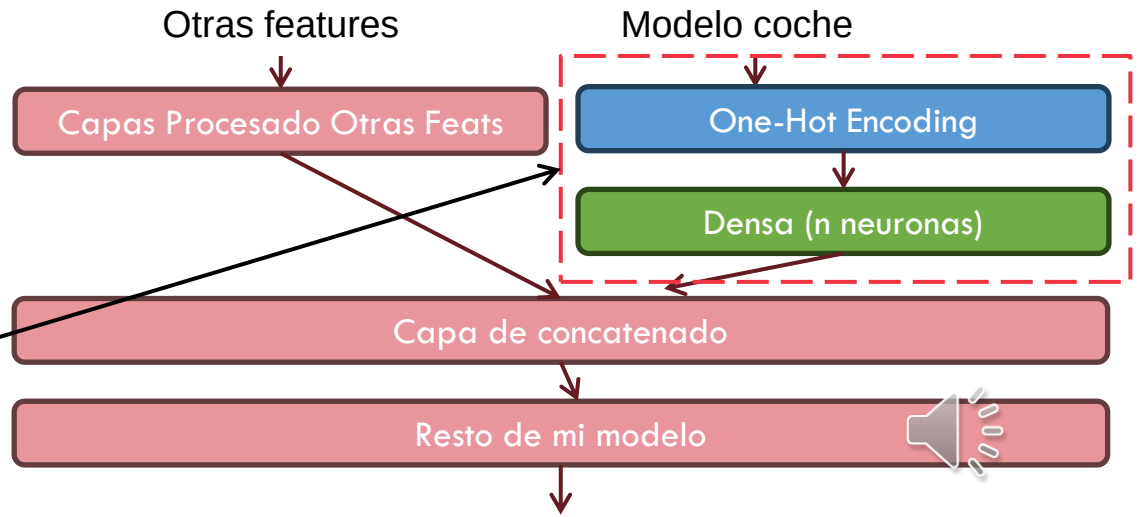
- Para esos casos lo que se hace es entrenar una capa de Embeddings



Embeddings

- Representación **compacta** de un conjunto de datos **de mayor dimensionalidad/cardinalidad** pero dispersos

- Para esos casos lo que se hace es entrenar una capa de Embeddings



Representational Learning

- Aprendizaje de embedding o codificaciones compactas de features de alta dimensionalidad

Representational Learning

- Aprendizaje de embedding o codificaciones compactas de features de alta dimensionalidad

- Permite codificar esas features categóricas “difíciles” (géneros de películas, modelos de coches, ciudades del mundo, equipos de fútbol, colores,…) (en modelos DL)

Representational Learning

- Aprendizaje de embedding o codificaciones compactas de features de alta dimensionalidad
- Permite codificar esas features categóricas "difíciles" (géneros de películas, modelos de coches, ciudades del mundo, equipos de fútbol, colores,…) (en modelos DL)
- Es especialmente interesante y útil cuando lo aplicamos a palabras (Word embeddings) y a texto en general

Representational Learning

- Aprendizaje de embedding o codificaciones compactas de features de alta dimensionalidad

- Permite codificar esas features categóricas "difíciles" (géneros de películas, modelos de coches, ciudades del mundo, equipos de fútbol, colores,…) (en modelos DL)

- Es especialmente interesante y útil cuando lo aplicamos a palabras (Word embeddings) y a texto en general (porque podemos entre otras cosas emplear Transfer Learning)


THE
BRIDGE