

Wordnet y Deep Learning: Una posible unión

Trabajo de fin de grado

Autor: Raquel Leandra Pérez Arnal

Directores: Dario Garcia Gasulla y Claudio Ulises Cortés García

Universidad Politécnica de Cataluña
Facultad de Matemáticas y Estadística

raquelpa93@gmail.com

22/01/18

Tabla de contenidos

- 1 Conocimientos Previos
- 2 Trabajo Relacionado
- 3 Enfoque
- 4 Análisis
- 5 Conclusiones

1 Conocimientos Previos

2 Trabajo Relacionado

3 Enfoque

4 Análisis

5 Conclusiones

Una Neurona

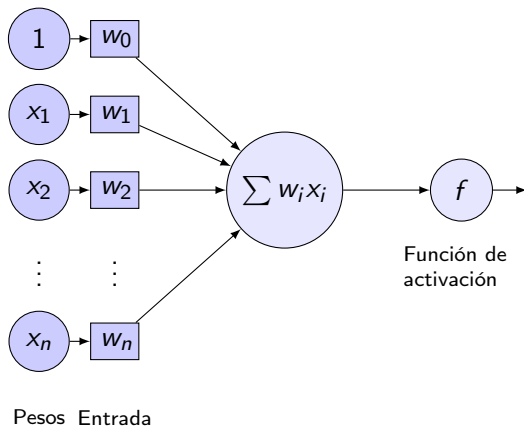


Figura 1: Ejemplo de una neurona

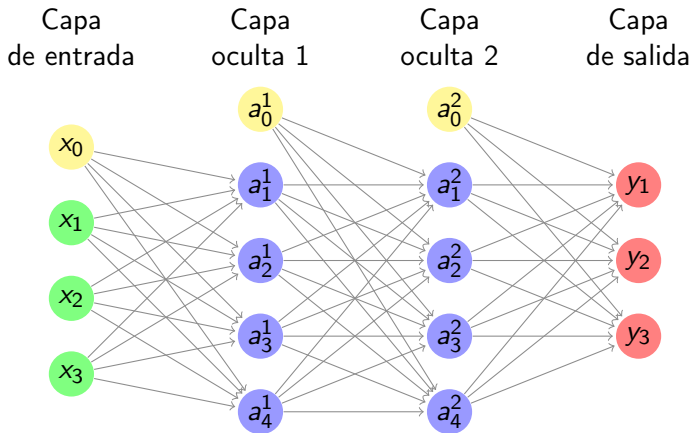


Figura 2: Ejemplo de red neuronal compuesta por capas completas

Redes Convolucionales

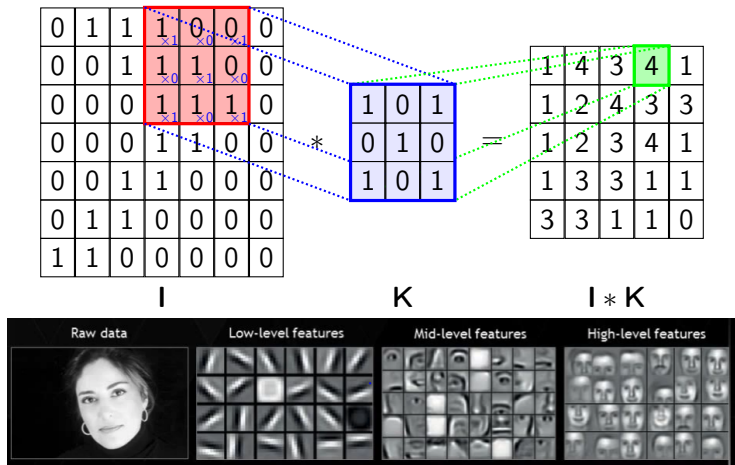


Figura 3: Ejemplo de convolución y de feature

Definición

Transfer learning es el campo de estudio que reutiliza el lenguaje de representación de un problema (que llamaremos problema origen o *Source*) para resolver otro (que llamaremos objetivo o *Target*).

- Fine Tuning
- Feature Extraction

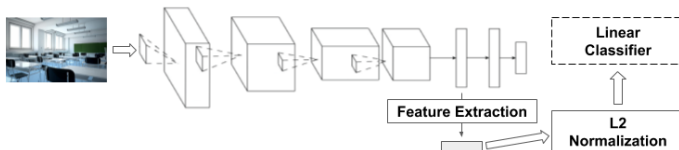


Figura 4: Estructura básica que se suele utilizar en *feature extraction*

Importancia del *Transfer Learning*

- Permite aplicar métodos de *Deep Learning* a conjuntos de datos de cualquier tamaño.
- No requiere tiempo y poder computacional.
- No requiere ajustar los hiper-parámetros.

1 Conocimientos Previos

2 Trabajo Relacionado

3 Enfoque

4 Análisis

5 Conclusiones

Full-Network Embedding

Partes del algoritmo:

- Forward Pass
- Spatial Pooling
- Feature Standarization
- Feature Discretization

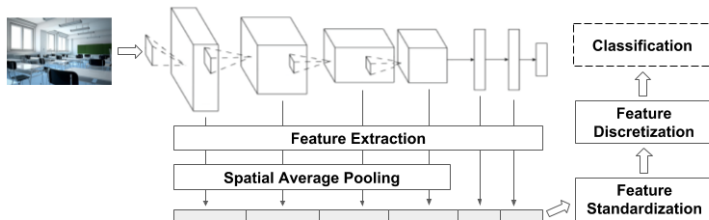


Figura 5: Estructura del *full-network embedding*

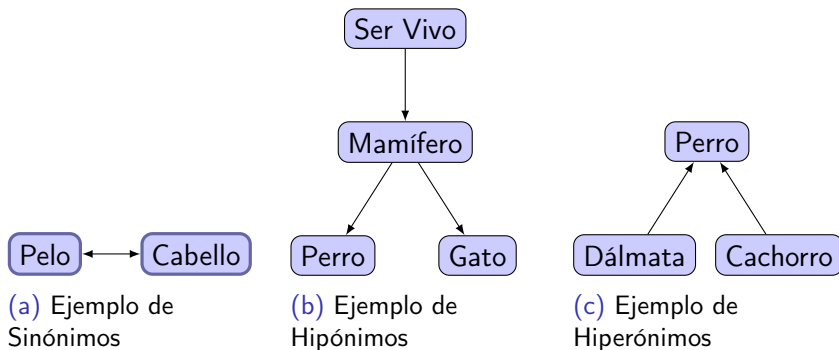
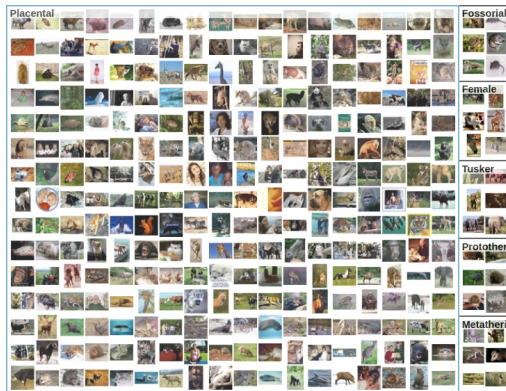


Figura 6: Ejemplo de las relaciones sintácticas de Wordnet



En nuestro caso hemos utilizado el subconjunto del reto de *Imagenet2012*:

	Cantidad	Clases
Entrenamiento	1.2 M	1000
Validación	50,000	1000

1 Conocimientos Previos

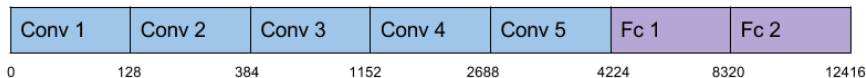
2 Trabajo Relacionado

3 Enfoque

4 Análisis

5 Conclusiones

Datos iniciales



	<i>features</i>							
ser vivo	1	1	-1	1	0	0	-1	} muestras
	0	0	-1	-1	-1	0	0	
	0	0	0	1	-1	-1	0	
mamífero	1	1	-1	1	-1	0	0	
	0	0	-1	1	0	0	0	
	0	-1	1	0	0	0	1	
ser vivo	1	1	-1	-1	-1	0	0	
	-1	-1	0	0	0	0	0	

Figura 7: Muestra de una sección del embedding (de tamaño total 50,000 x 12,416).

Objetivos

- Analizar el *embedding* dado y el comportamiento de las *features* en las distintas capas.
- Analizar si hay alguna relación entre el *embedding* y los *synsets* seleccionados.

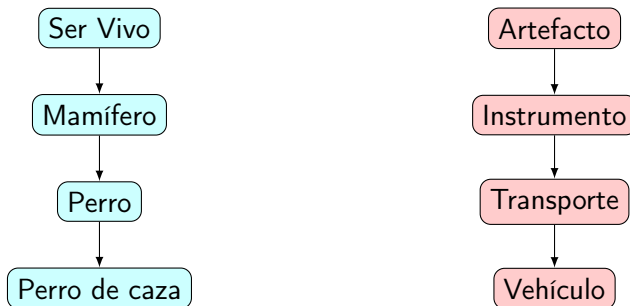


Figura 8: Conjuntos de synsets que estudiaremos

Estadísticas del embedding

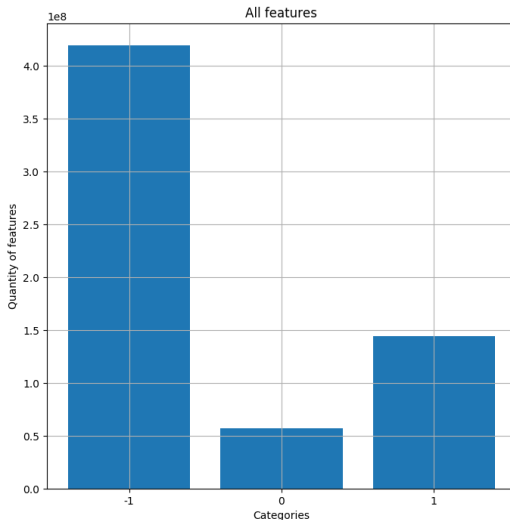
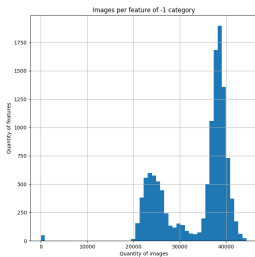
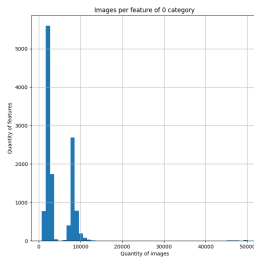


Figura 9: Cantidad de *features* de cada categoría

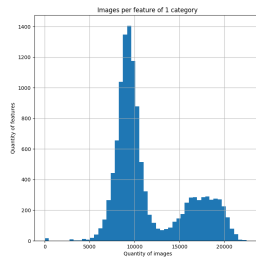
Estadísticas del embedding



(a) Categoría -1



(b) Categoría 0



(c) Categoría 1

Figura 10: Distribución del número de *features* con los distintos valores categóricos, para las 50,000 imágenes

Distribución de los synsets en el embedding

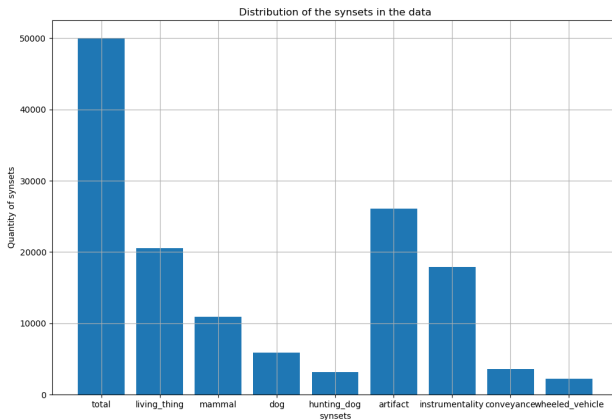


Figura 11: Cantidad de imágenes de cada *synset* respecto al embedding total

- 1 Las características se distribuyen de diferente manera en las capas convolucionales y los completos.
- 2 La cantidad de *features* representativas aumenta con la profundidad.
- 3 Cuanto más concreto es un *synset*, más *features* representativas.
- 4 Se puede ver una relación entre los embeddings de synsets hipónimos.

1 Conocimientos Previos

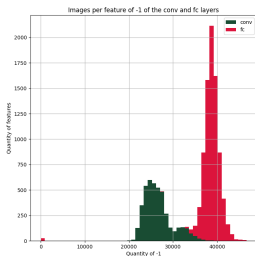
2 Trabajo Relacionado

3 Enfoque

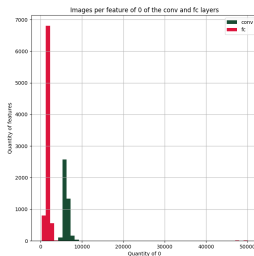
4 **Análisis**

5 Conclusiones

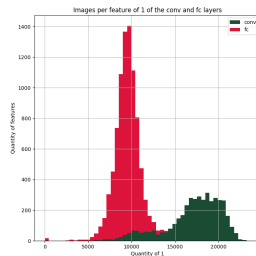
Distribución por tipo de capa



(a) Categoría -1



(b) Categoría 0



(c) Categoría 1

Figura 12: Distribución del número de *features* con los distintos valores categóricos distinguiendo las capas convolucionales de las *fully-connected*

Comportamiento respecto a la profundidad

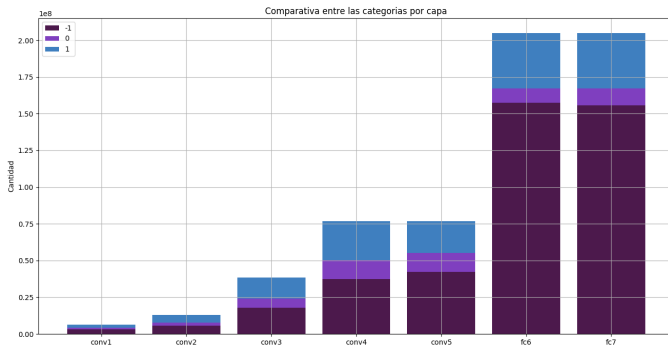


Figura 13: Cantidad de *features* de cada categoría por capa

Sub-matriz

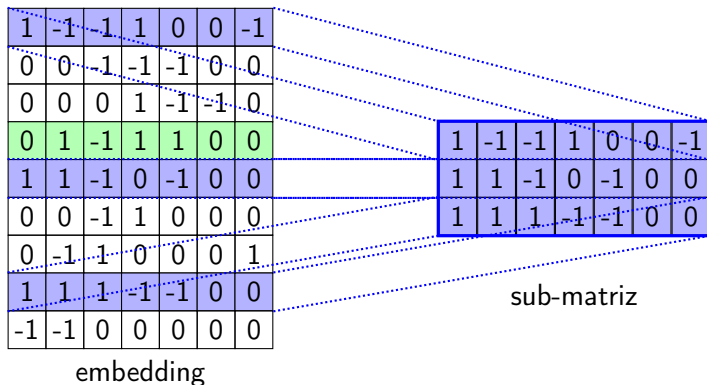
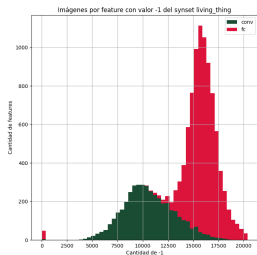
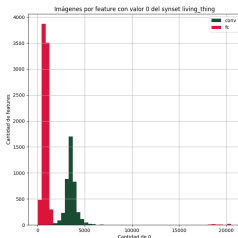


Figura 14: Ejemplo de una sub-matriz de un *synset*.

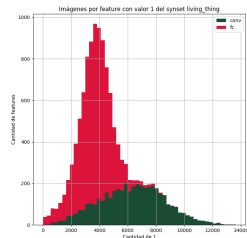
Distribución por tipo de capa entre los *synsets*



(a) Categoría -1



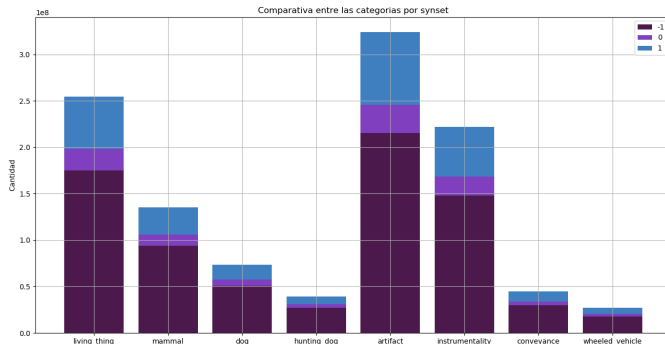
(b) Categoría 0



(c) Categoría 1

Figura 15: Distribución del número de *features* con los distintos valores categóricos distinguiendo las capas convolucionales de las *fully-connected* del *synset* seres vivos

Distribución de las *features* entre los *synsets*



	Ser Vivo	Mamífero	Perro	Perro de Caza	Artefacto	Instrumento	Transporte	Vehículo
-1	0.69	0.69	0.70	0.70	0.66	0.67	0.66	0.65
0	0.09	0.09	0.09	0.09	0.09	0.09	0.09	0.09
1	0.22	0.22	0.21	0.21	0.24	0.24	0.25	0.26

Figura 16: Cantidad y proporción de *features* de cada categoría por synset

Representante

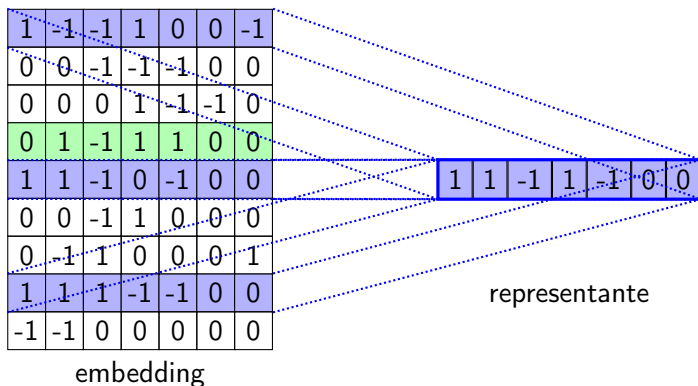


Figura 17: Ejemplo de un representante de *synset*.

Distribución de las *features* entre los representantes

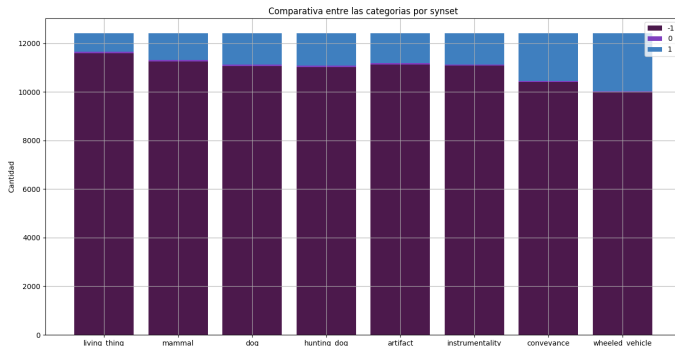


Figura 18: Cantidad de *features* de cada tipo de los representantes de los distintos *synsets*

Distribución de las *features* entre los representantes por capa

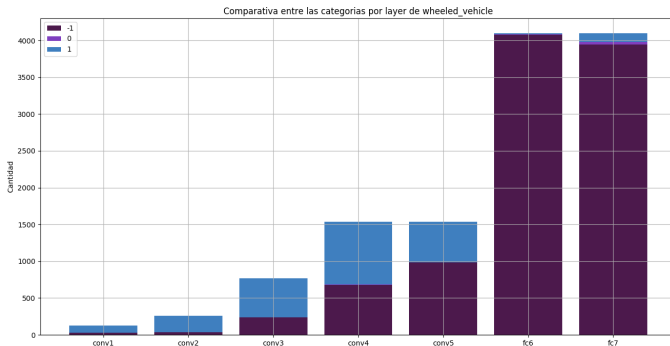


Figura 19: Cantidad de *features* de cada tipo del representante del *synset* Vehículo por capa.

Matrices de cambio

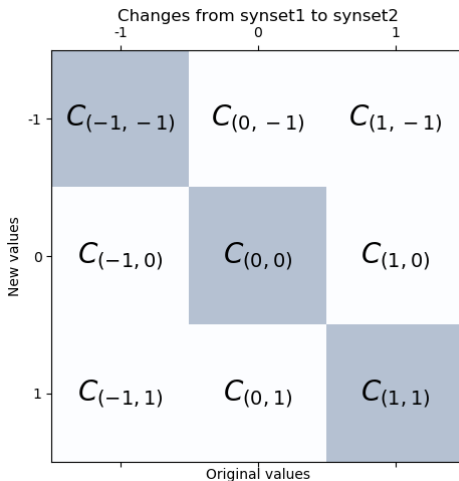


Figura 20: Matriz de cambios general

$$T = C_{(1,-1)}(s_1, s_2) + C_{(1,0)}(s_1, s_2) + C_{(1,1)}(s_1, s_2) + C_{(1,1)}(s_1, s_2) + C_{(0,1)}(s_1, s_2) + C_{(-1,1)}(s_1, s_2)$$

$$d(s_1, s_2) = 1 - \frac{C_{(1,1)}(s_1, s_2)}{T}$$

	Instrumento	Vehículo	Perro de Caza
Ser Vivo	0.9965	0.9333	0.5520
Perro	0.9671	0.8753	0.1614
Transporte	0.2201	0.2192	0.9124



(a) Spaniel



(b) Greyhound



(c) Water Spaniel

Figura 21: Ejemplos de razas

	Spaniel	Grayhound	Water Spaniel
Spaniel	0	0.7371	0.6442
Grayhound	0.7371	0	0.8330
Water Spaniel	0.6442	0.8330	0

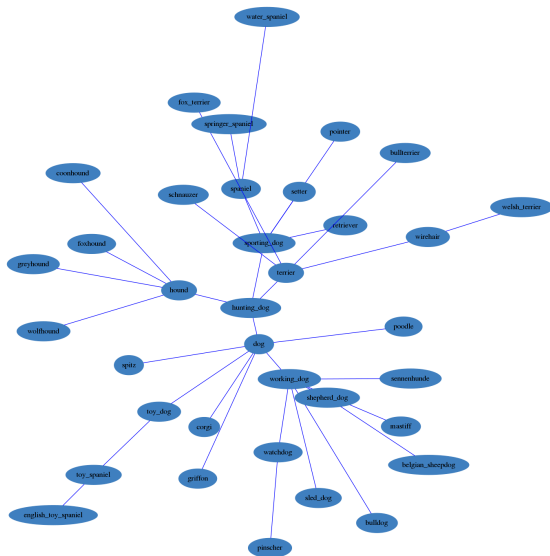


Figura 22: Ejemplo de Árbol del *synset* perro con distancias normalizadas según la métrica comentada.

1 Conocimientos Previos

2 Trabajo Relacionado

3 Enfoque

4 Análisis

5 Conclusiones

- Las capas *fully-connected* tienen una distribución diferente de las convolucionales.
- Todas las *features* contienen información que caracteriza el espacio de representación.
- Las proporciones de las *features* se mantienen respecto a las profundidad de las capas.
- La proporción de las *features* de las tres categorías se mantiene respecto a los diferentes synsets.
- El *embedding* detecta similitud a nivel de *synset*.
- Utilizando la pseudo-métrica definida podemos medir esta distancia y representarla gráficamente.

Gracias por vuestra atención.



Podéis encontrar el código utilizado en el trabajo en:
github.com/RaquelLeandra/TFG-WordnetDeepLearning