Taller de Aprendizaje Automático

Taller 9: Natural Language Processing (NLP)

Instituto de Ingeniería Eléctrica
Facultad de Ingeniería

UNIVERSIDAD
DE LA REPÚBLICA
URUGUAY

Objetivos del Taller

- Aplicar modelos basados en RNN a un problema de NLP.
- Trabajar con embeddings para secuencias de texto, en particular embeddings preentrenados.
- Utilizar herramientas para la visualización de embeddings.
- (Opcional) Desarrollar una aplicación web que clasifique críticas proporcionadas por los usuarios.

El Problema

- Se cuenta con críticas de películas, y se quiere determinar si la crítica es *positiva* o *negativa*.

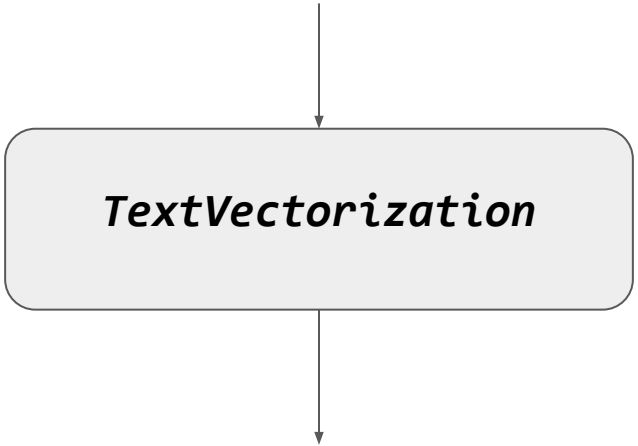
- 35000 datos de *entrenamiento* y 15000 datos de *test*.

	review	sentiment
0	One of the other reviewers has mentioned that ...	positive
1	A wonderful little production. The...	positive
2	I thought this was a wonderful way to spend ti...	positive
3	Basically there's a family where a little boy ...	negative

Figura: IMBd

Capa **TextVectorization**

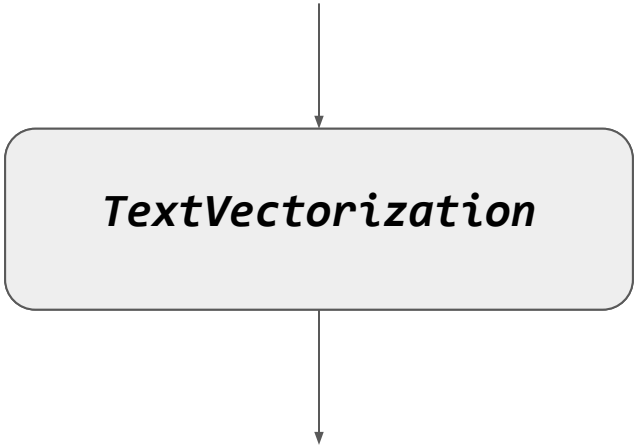
“This 1950s movie is truly boring, despite the ….”



[11, 2157, 18, 7, 343, 346, 451, 2, 111, 7, 1070]

Capa **TextVectorization**

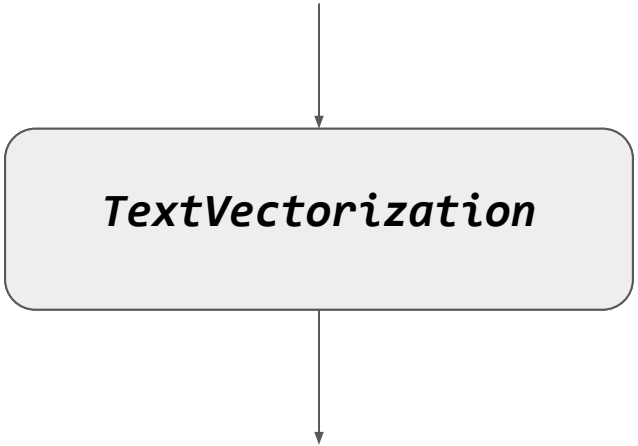
"This 1950s movie is truly boring, despite the ...."



[11, 2157, 18, 7, 343, 346, 451, 2, 111, 7, 1070]

Capa **TextVectorization**

“This 1950s movie is truly boring, despite the ….”



[11, 2157, 18, 7, 343, 346, 451, 2, 111, 7, 1070]

Capa **TextVectorization**

- Definir la capa

```
# Capa que vectoriza las palabras
text_vec_layer = tf.keras.layers.TextVectorization(max_tokens=vocab_size,
                                                   standardize='lower_and_strip_punctuation',
                                                   split='whitespace')
```

- Adaptar la capa

```
# Adaptar la capa
text_vec_layer.adapt(train_set.map(lambda reviews, labels: reviews))
```

Capa **TextVectorization**

- Obtener el diccionario

```
# Obtener el diccionario aprendido
words = text_vec_layer.get_vocabulary()
```

Palabra	Token ID
''	0
'[UNK]'	1
'the'	2
'and'	3

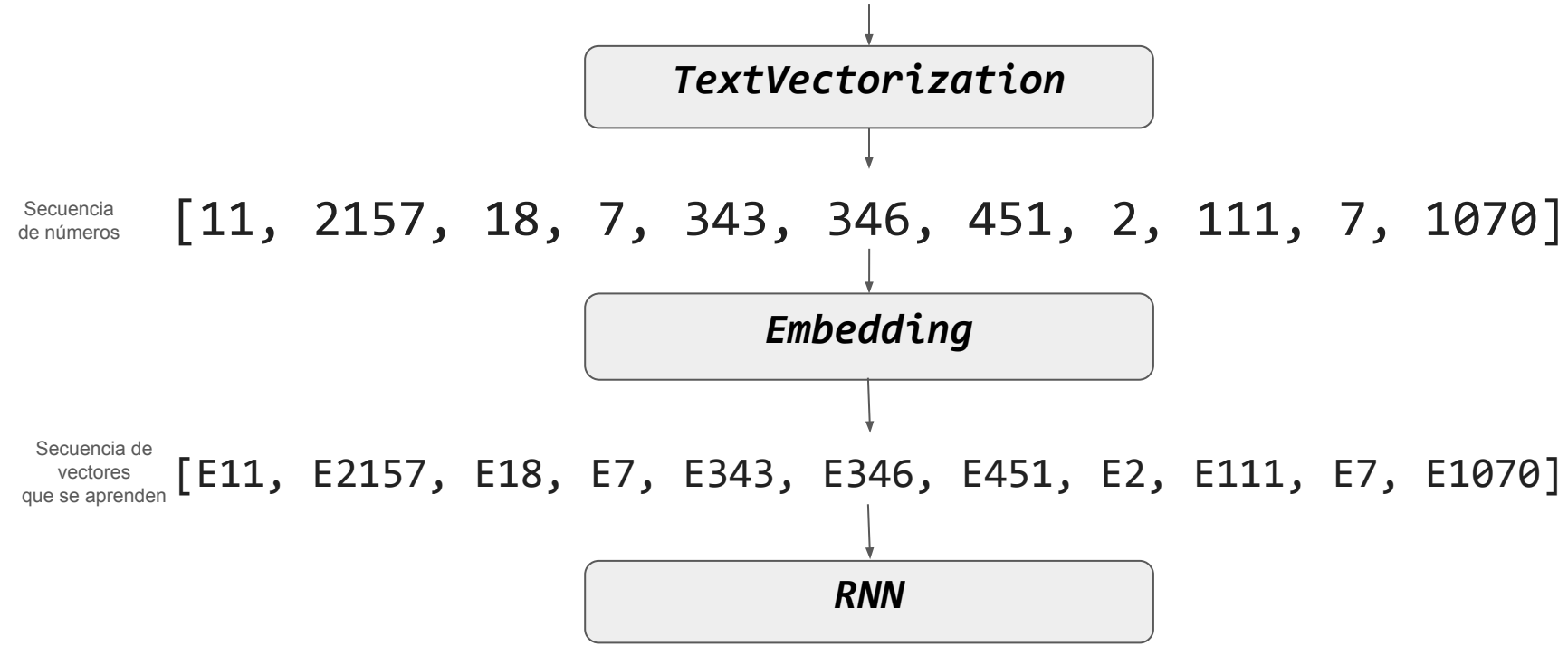
Modelo

Modelo sección *Sentiment Analysis (Capítulo 16)*

```
embed_size = 128
tf.random.set_seed(42)
model = tf.keras.Sequential([
    text_vec_layer,
    tf.keras.layers.Embedding(vocab_size, embed_size),
    tf.keras.layers.GRU(128),
    tf.keras.layers.Dense(1, activation="sigmoid")
])
model.compile(loss="binary_crossentropy", optimizer="nadam",
              metrics=["accuracy"])
history = model.fit(train_set, validation_data=valid_set, epochs=5)
```

Capa **TextVectorization + Embedding + RNN**

“This 1950s movie is truly boring, despite the ….”



Embedding

Una embedding es un vector denso entrenable.

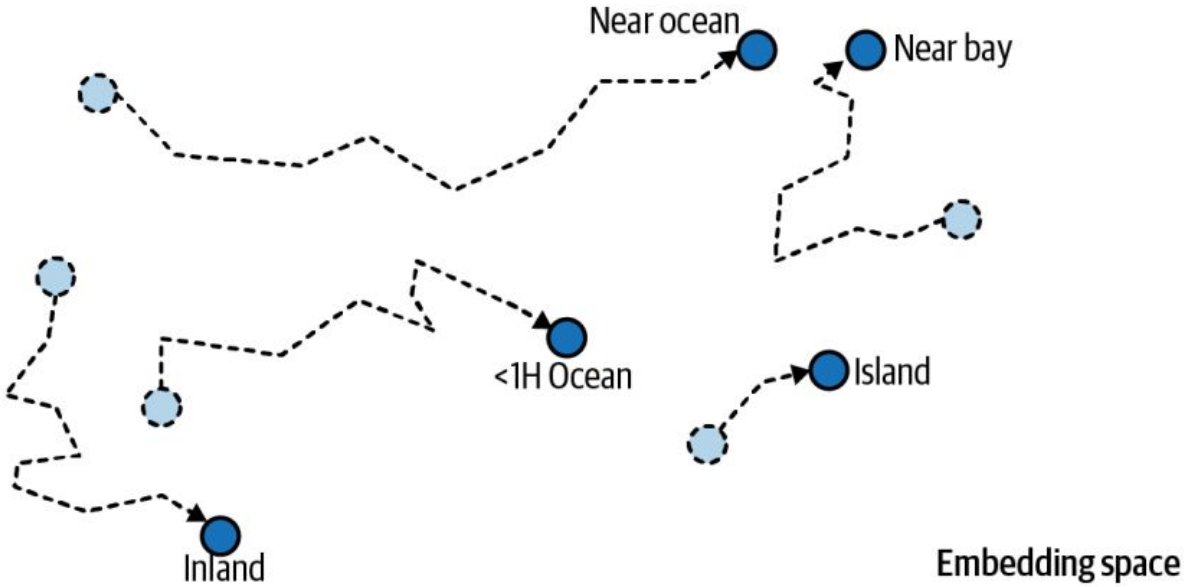
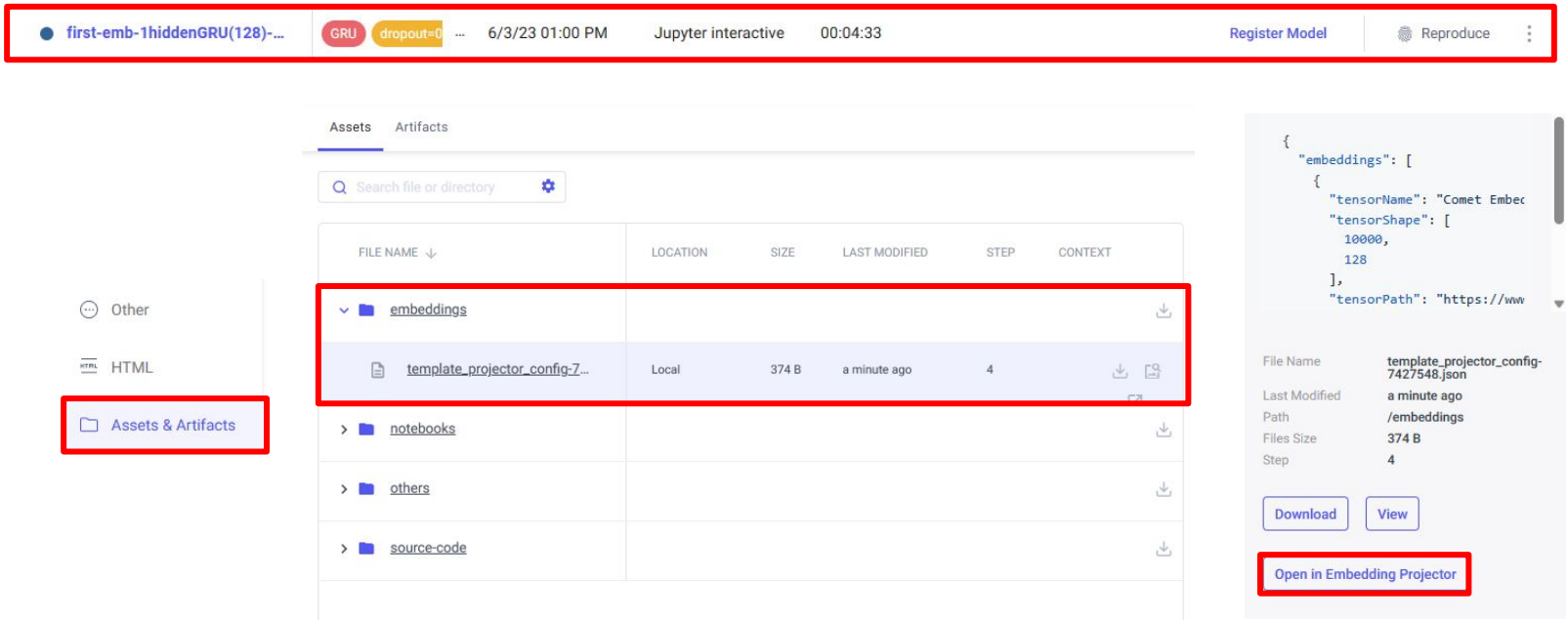


Figura: Encoding Categorical Features Using Embeddings, capítulo 13

Embedding

Visualización de Embedding en Comet:



Embedding Pre-Entrenado

GloVe: Global Vectors for Word Representation

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

3. litoria

4. leptodactylidae

5. rana

7. eleutherodactylus

Figura: Glove

Embedding Pre-Entrenado

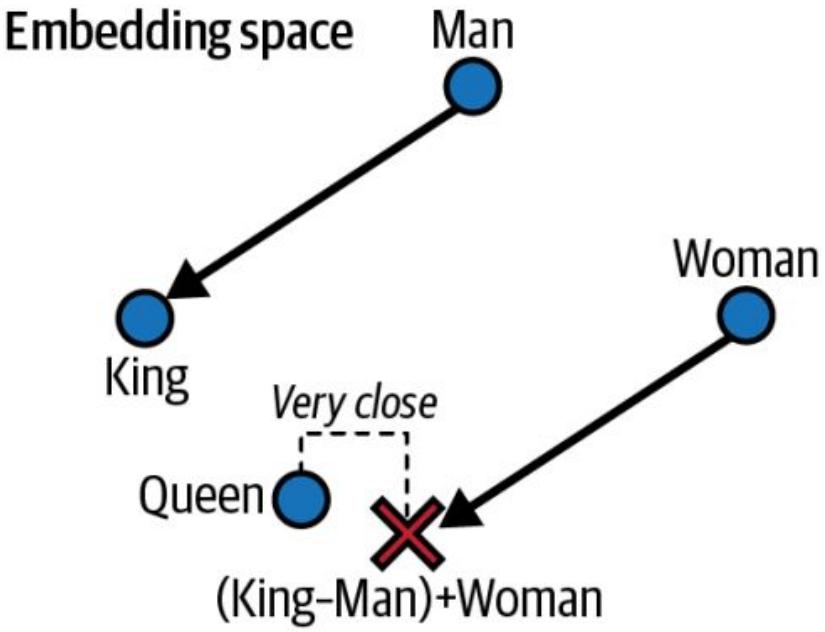
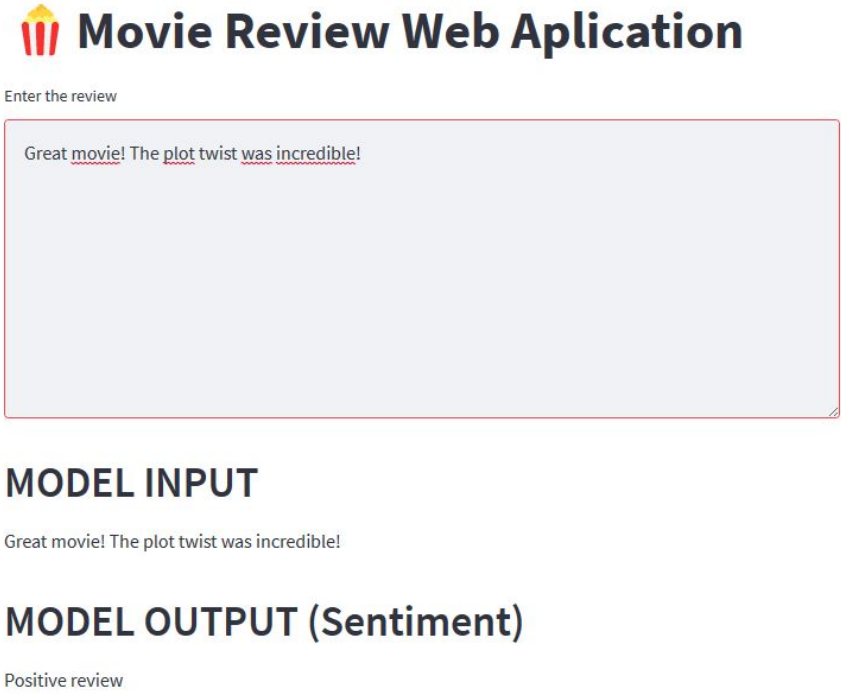


Figura: Words Embedding, capítulo 13.

Streamlit


Movie Review Web Aplication
Enter the review
Great movie! The plot twist was incredible!
MODEL INPUT
Great movie! The plot twist was incredible!
MODEL OUTPUT (Sentiment)
Positive review