

NLP

Sequence to Sequence (seq2seq)

Dr. Rodrigo Cardenas Szigety
rodrigo.cardenas.sz@gmail.com

Programa de la materia



Clase 1: Introducción a NLP, Vectorización de documentos.

Clase 2: Preprocesamiento de texto, librerías de NLP, bots de información.

Clase 3: Word Embeddings, CBOW y SkipGRAM, entrenamiento de embeddings.

Clase 4: Redes recurrentes (RNN), problemas de secuencia y estimación de próxima palabra.

Clase 5: Redes LSTM, análisis de sentimientos.

Clase 6: Modelos Seq2Seq, traductores y bots conversacionales.

Clase 7: Celdas con Attention. Transformers, BERT & ELMo, fine tuning.

Clase 8: Cierre del curso, NLP hoy y futuro, deploy.

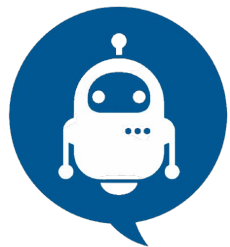
*Unidades con desafíos a presentar al finalizar el curso.

*Último desafío y cierre del contenido práctico del curso.

Soluciones Seq2Seq



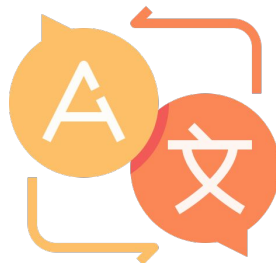
Trabaja principalmente con el concepto many-to-many en formato "codificador" a "decodificador", en donde la secuencia de entrada se codifica a una representación intermedia y se decodifica al espacio de salida.



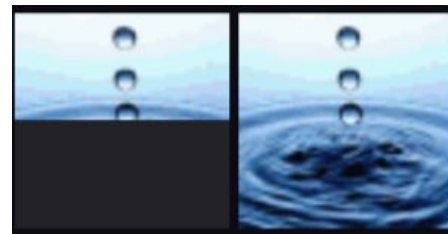
Bots Conversacionales
(modelos de lenguaje)



Generar música



Traducción de
idiomas



Completar una
imagen

Chat bot conversacional

[LINK](#)



"Utilizan IA entrenados en un dominio cerrado o abierto para generar una respuesta basada en el set de entrenamiento".

Requiere mucha más información pero tiene más poder de interpretabilidad y de responder a preguntas nunca antes realizadas.

La respuesta es totalmente generada, por lo que se tiene menos control del resultado y es más probable obtener un error de sentencia.

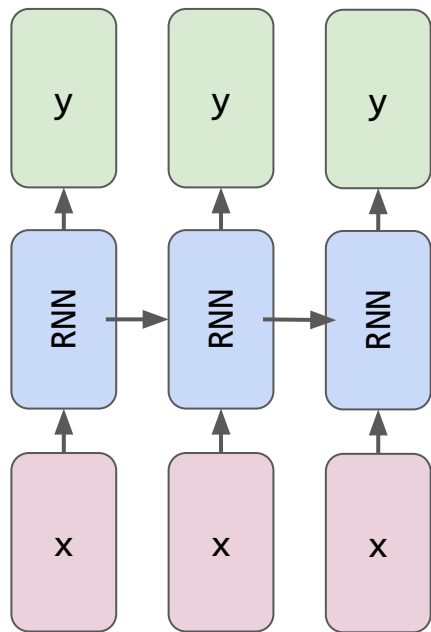
Conversations	Open Domain	Impossible	General AI [Hardest]
	Closed Domain	Rules-Based [Easiest]	Smart Machine [Hard]
		Retrieval-Based	Generative-Based
		Responses	

many-to-many

[LINK](#)



"Dada una entrada de tamaño fijo el sistema arroja una sentencia o oración a partir de ella de longitud fija"



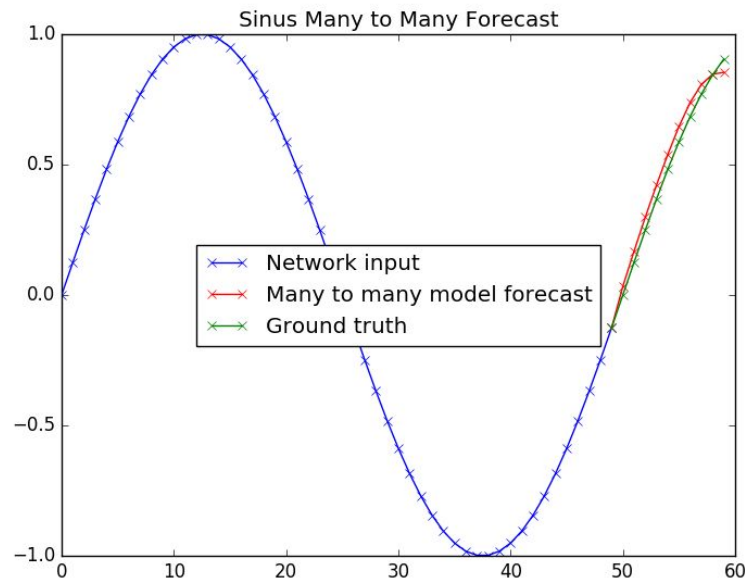
many-to-many



Este tipo de estructuras no son muy utilizadas porque solo son útiles para secuencias de entrada y salida fija (no necesariamente iguales).



Es mucho más simple este tipo de estructuras que las que veremos para NLP con encoder-decoder





Link al Colab



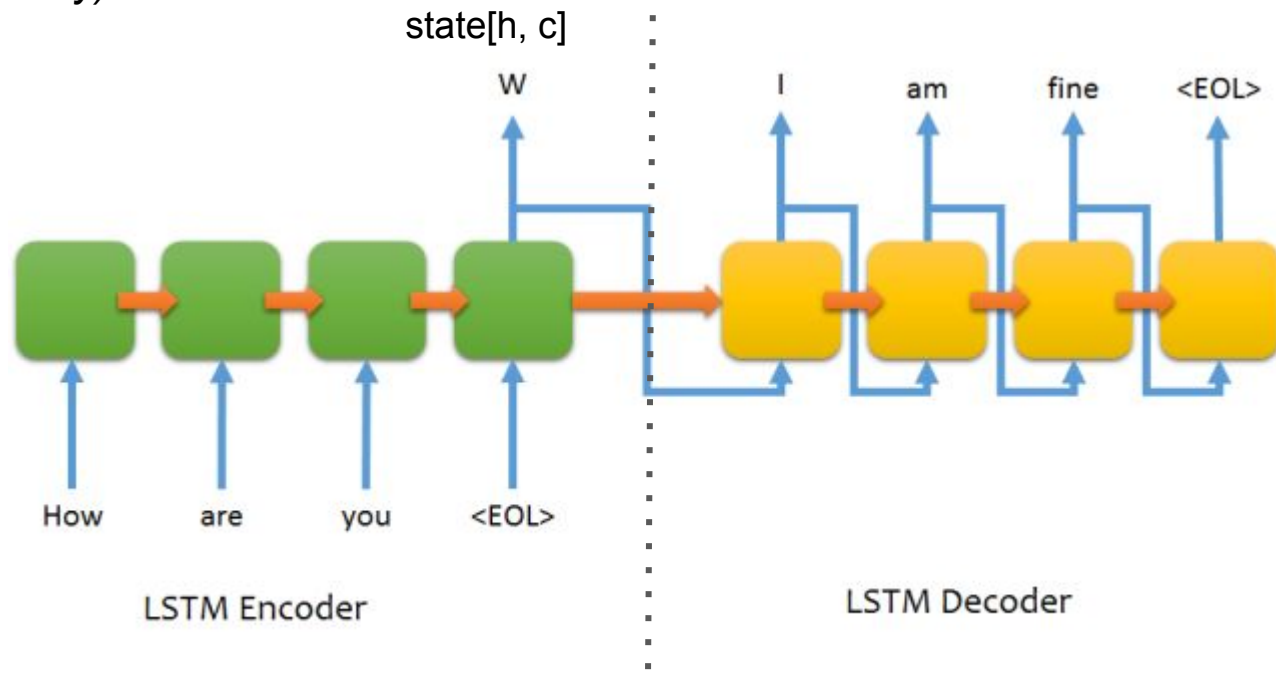
LINK

Seq2Seq con encoder-decoder



"Modelo basado en dos partes, la primera genera un "espacio latente" o "contexto" que alimenta a la segunda parte, la cual realiza una inferencia realimentada de la última salida." (simil one-to-many)

La primera inferencia depende del encoder y su estado final reemplaza en el decoder el estado inicial h_{t0} . En el decoder el modelo es auto-regresivo.

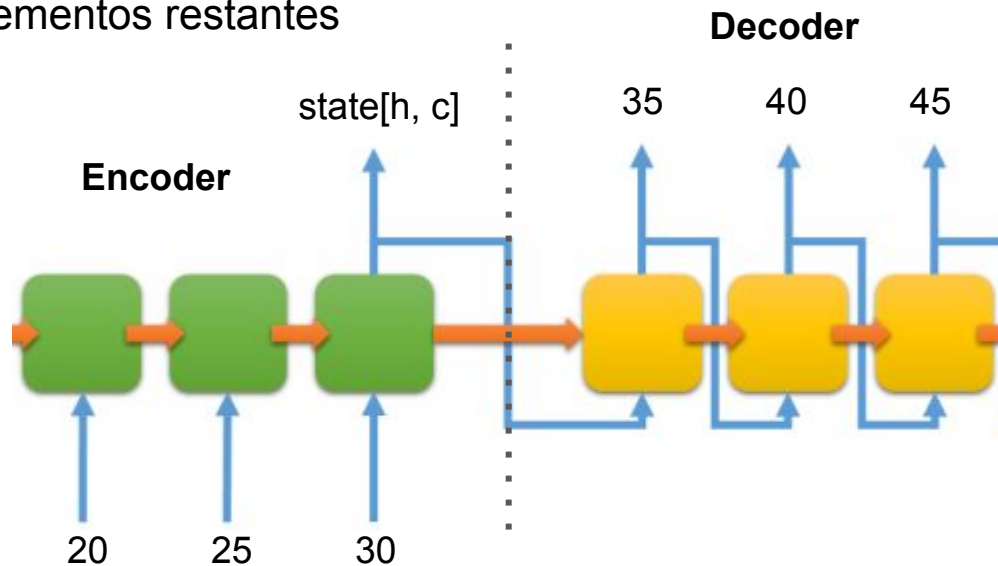


Encoder-Decoder en secuencia numérica



Al igual que en el ejemplo de many to many, el encoder procesará toda la secuencia de entrada produciendo un estado oculto que se pasará como primer estado al decoder.

El decoder utiliza ese estado oculto y su propia realimentación de salida para producir los elementos restantes



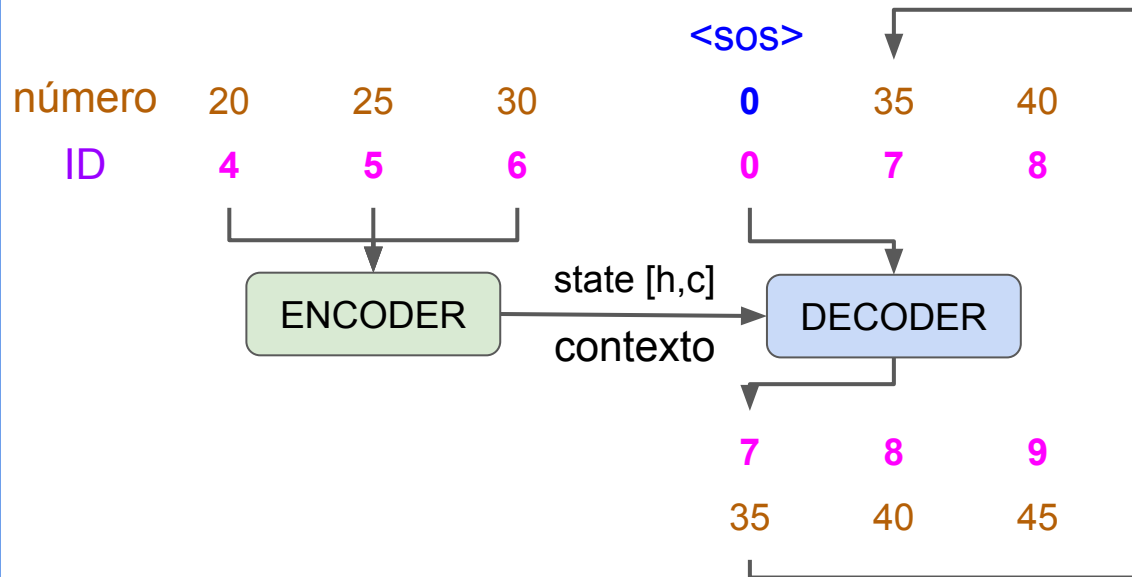
***Nota**

Tener en cuenta que el encoder-decoder recibirá un vector onehotEncoding que represente (embedding) a los números, ya que el espacio de posibilidades debe ser acotado (discreto)

Tokens especiales <SOS> & <EOS>



Tokens que se reservan para indicarle al modelo el comienzo (Start Of Sequence) o el fin (End Of Sequence) de la secuencia.



Las **palabras/números** se transforman en tokens (**ids**) con el Tokenizador o LabelEncoder

En este ejemplo de secuencia numérica usaremos el número “0” como **<sos>**.

El LabelEncoder le dará un **ID** a ese token especial (en este caso también 0)

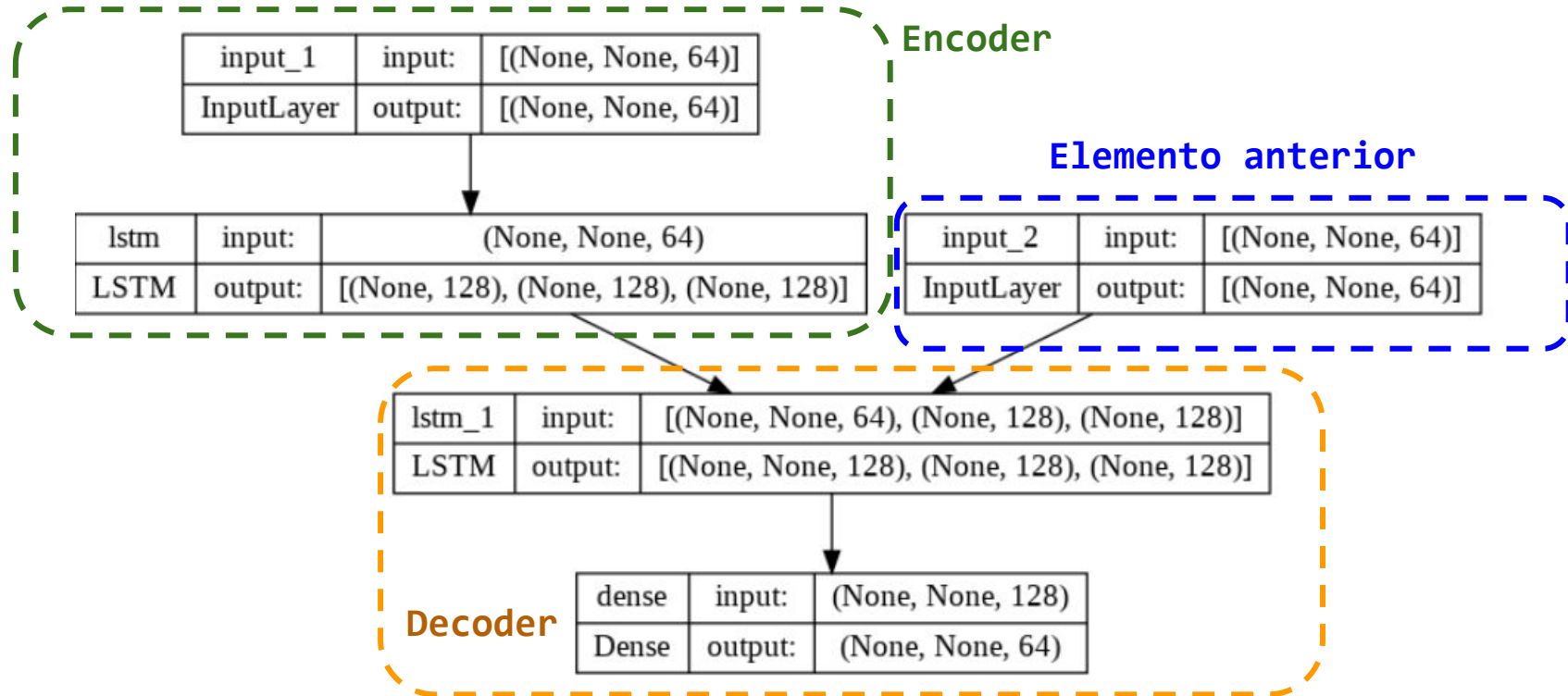
Es importante que el token esté representado por una palabra/número que no exista en el vocabulario (para no confundirlo)

LSTM encoder-decoder

[LINK](#)



El modelo que se entrena es el "completo", con el encoder y decoder.

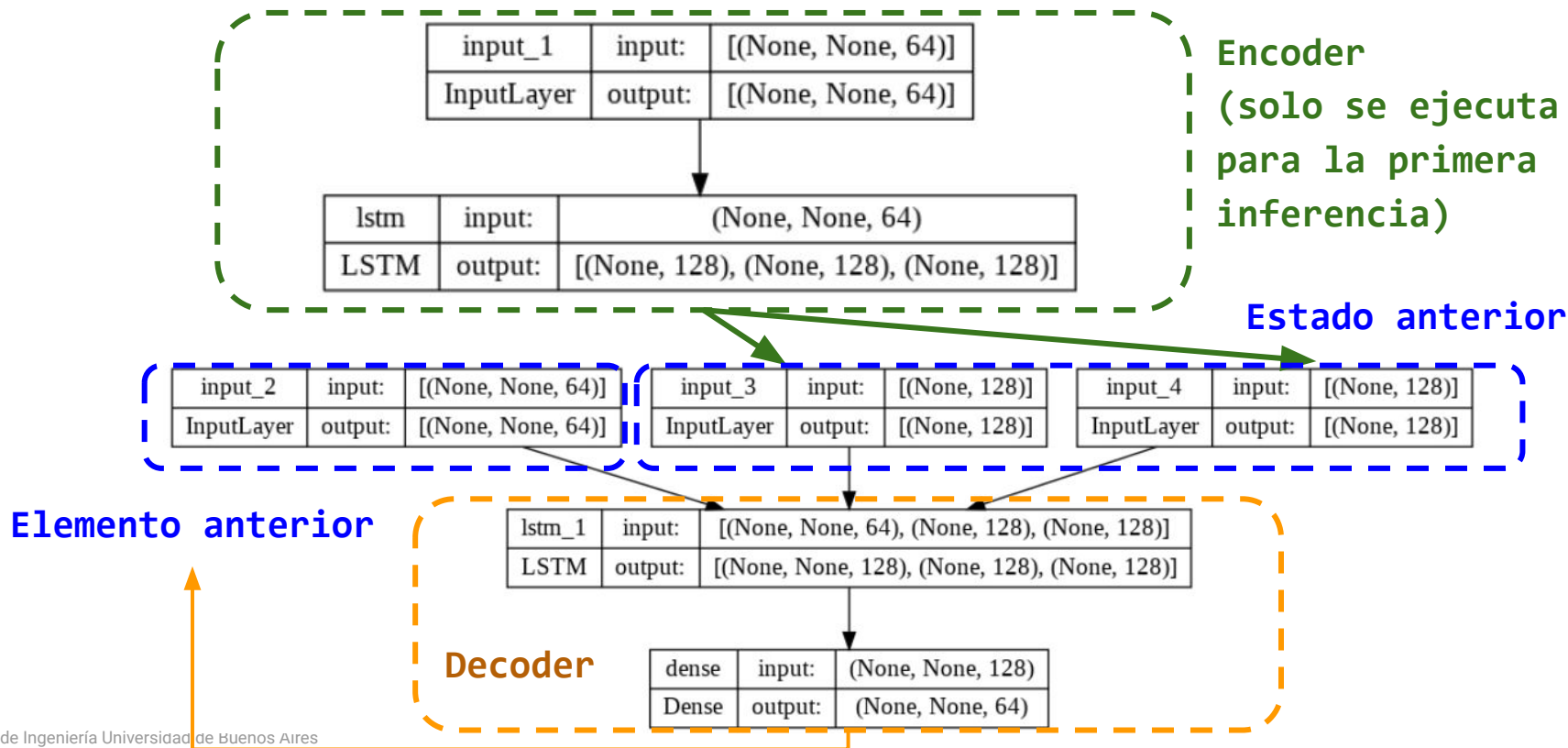


LSTM Decoder

[LINK](#)



Para la inferencia se utiliza por separado el encoder y el decoder.
El decoder funciona de manera auto-regresiva.





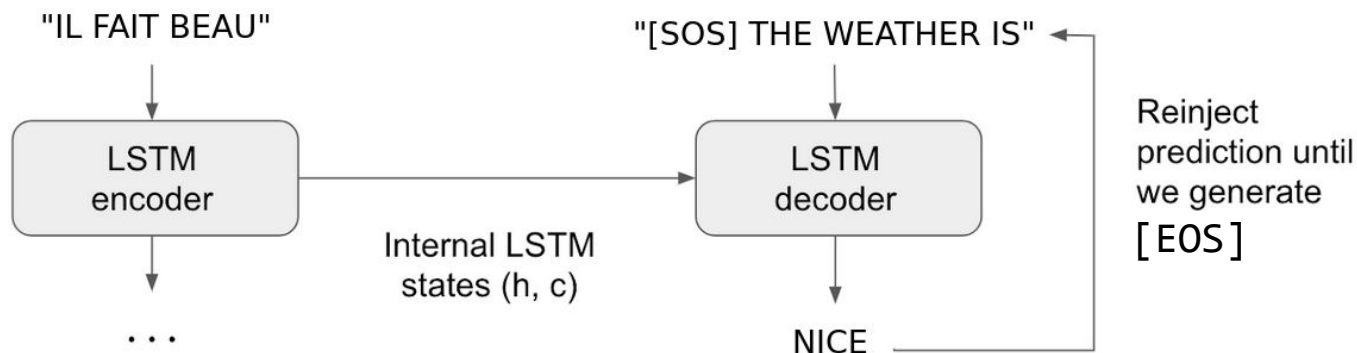
Link al Colab



LINK



Cuando hablamos de un encoder-decoder NLP se agrega un grado de dificultad más, ya que las secuencias no necesariamente tienen el mismo tamaño y que hay que vectorizar las sentencias de entrada

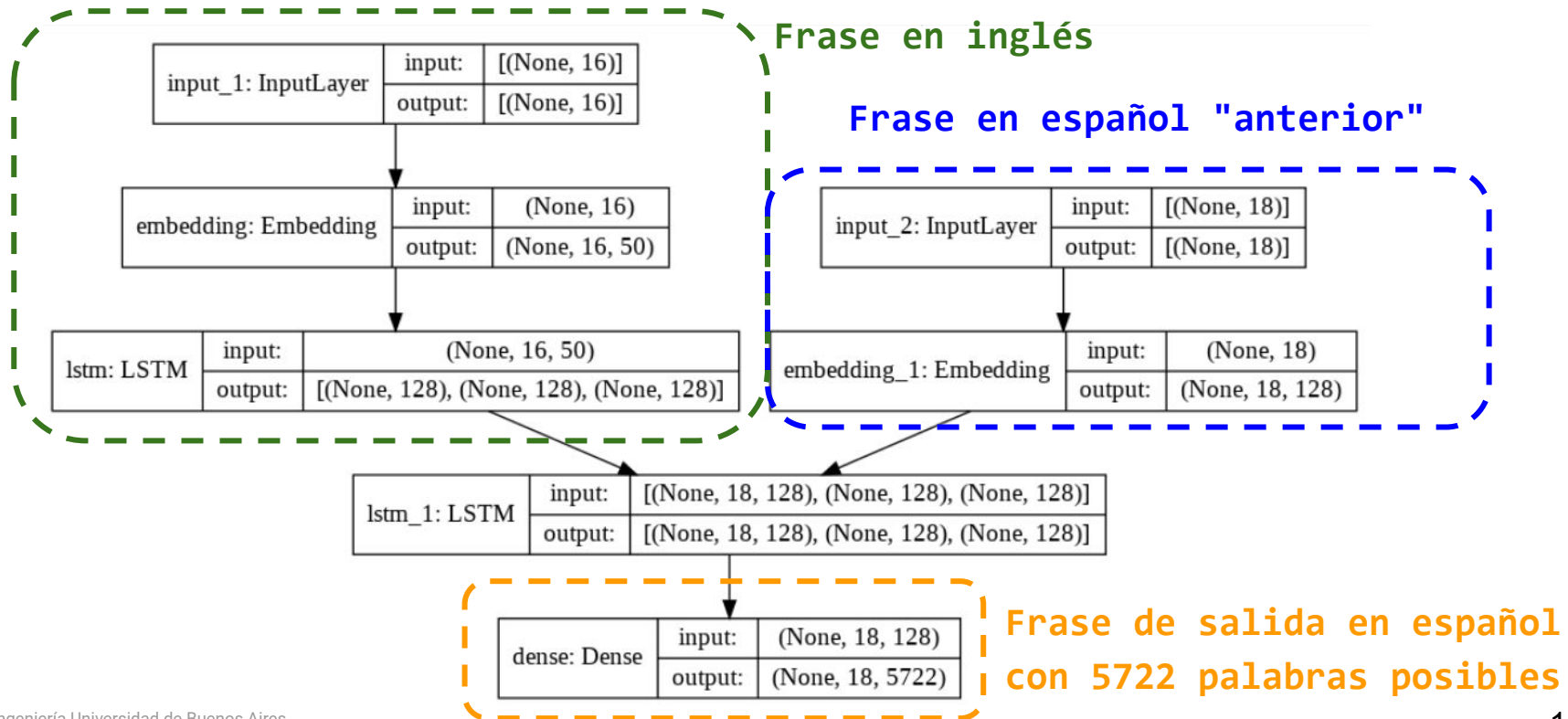


Para solucionar el problema de secuencias de distinto tamaño se define una máxima longitud y luego se acota con los tokens de inicio y fin de sentencia (<sos>/<eos>)

Traductores



En este ejemplo realizaremos un traductor de inglés a español, vectorizando las sentencias de entrada con Embeddings



Inferencia del traductor



El encoder inicializa el contexto (h1,c1) con la entrada del decoder en <sos>, luego la salida es realimentada.

```
('A deal is a deal.',  
'Un trato es un trato. <eos>',  
'<sos> Un trato es un trato.')
```

Input: Tom is naked.
Response: tom es un noche

Ensayo real, formó una
oración coherente pero no era
el resultado solicitado

Step 1:

A deal is a deal -> Encoder -> enc(h1,c1)

enc(h1,c1) + <sos> -> Decoder -> Un + dec(h1,c1)

step 2:

dec(h1,c1) + Un -> Decoder -> trato + dec(h2,c2)

step 3:

dec(h2,c2) + trato -> Decoder -> es + dec(h3,c3)

step 4:

dec(h3,c3) + es -> Decoder -> un + dec(h4,c4)

step 5:

dec(h4,c4) + un -> Decoder -> trato + dec(h5,c5)

step 6:

dec(h5,c5) + trato. -> Decoder -> <eos> + dec(h6,c6)



Link al Colab



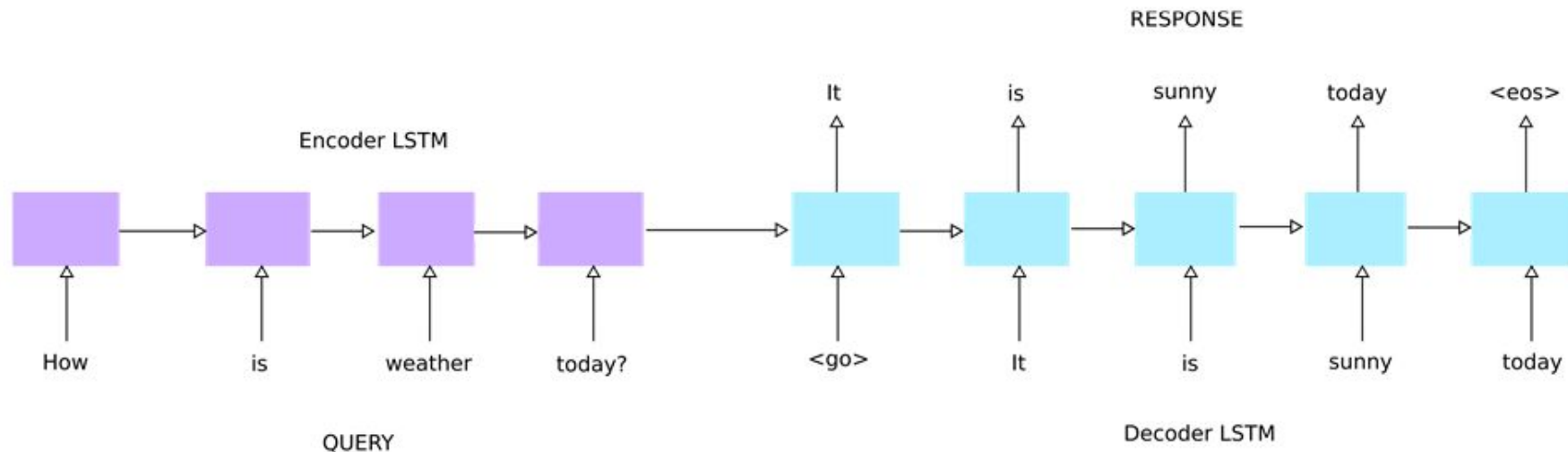
[LINK](#)

Question and answers (QA) ~ Bot

[LINK](#)



Es hora de armar un Bot conversacional, que responda a preguntas que nosotros le hagamos (QA). Para ello utilizaremos un dataset "modesto" por lo que no se espera alcanzar resultados muy prometedores



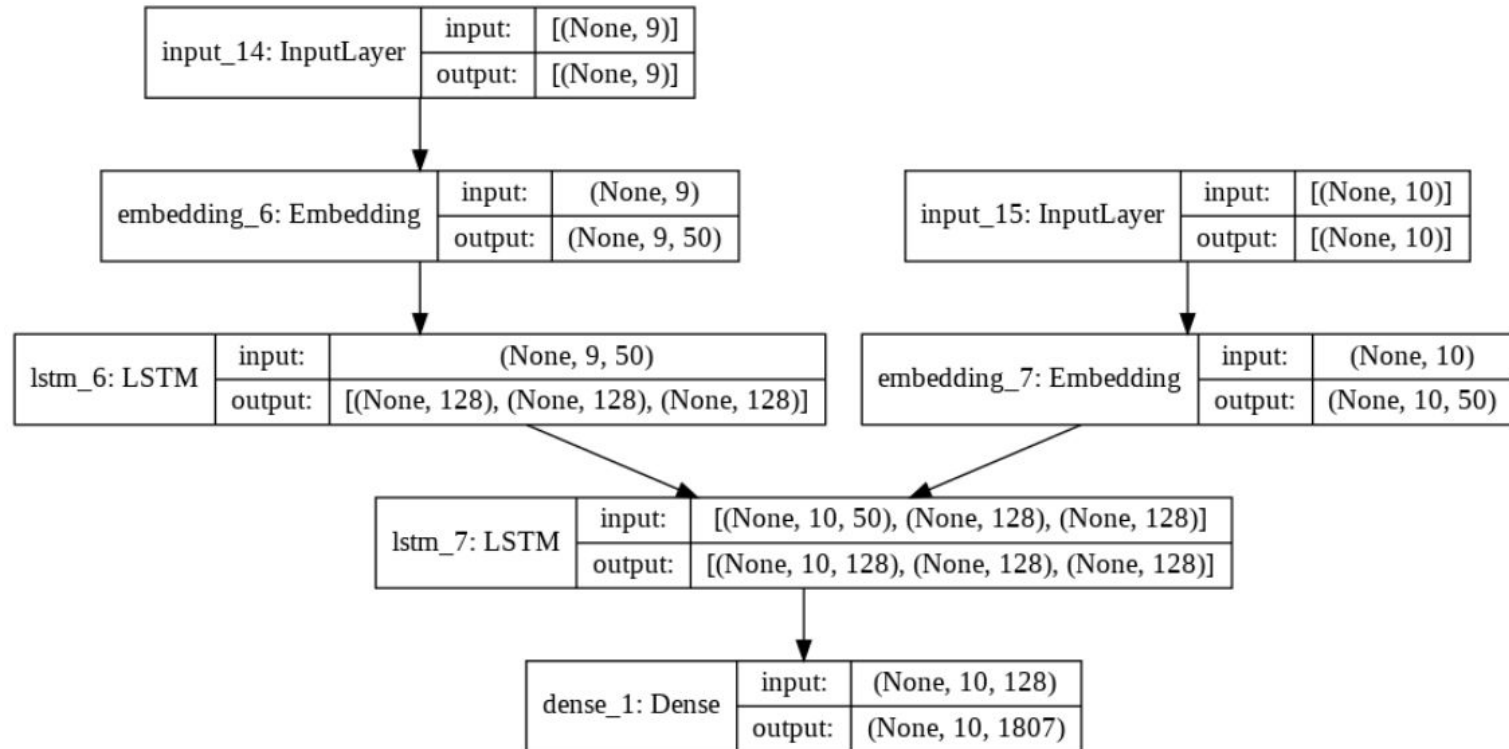
Input: i like to go to the beach
Response: i am a vegan

Question and answers (QA) Bot

[LINK](#)



Un ejemplo de una arquitectura utilizando los Embeddings de Glove (dim 50)





Construir QA Bot basado en el ejemplo del traductor pero con un dataset QA

[LINK](#)

Recomendaciones para evitar esto...



Recomendaciones:

MAX_VOCAB_SIZE = 8000
max_length ~ 10
Embeddings 300 Fasttext
n_units = 128
LSTM Dropout 0.2
Epochs 30~50

Preguntas interesantes:

Do you read?
Do you have any pet?
Where are you from?

Frameworks para crear modelos seq2seq [LINK](#)



Algunos frameworks/librerías que traen modelos e interfaces preparadas para armar rápidamente un sistema basado en NLP.



DeepPavLov



Hugging face



ParlAI