NLP

CHELO RICHART

Índice

NLP

WORD2VEC

LSTM

NLP

El Procesamiento del Lenguaje Natural (NLP, por sus siglas en inglés) es **una rama de la inteligencia artificial** y la lingüística computacional que se enfoca en la **interacción entre las máquinas y el lenguaje humano.**

El objetivo del NLP es permitir que las ordenadores comprendan, procesen y generen lenguaje humano de manera efectiva.

El NLP se utiliza en una amplia gama de aplicaciones,

- el análisis de sentimientos
- la traducción automática
- el reconocimiento de voz
- la generación de texto
- la extracción de información
- la búsqueda semántica.

Para lograr estos objetivos, el NLP utiliza una combinación de técnicas lingüísticas, estadísticas y de aprendizaje automático.

El NLP es una tecnología en constante evolución y desarrollo, y se espera que su uso se expanda aún más en el futuro, especialmente en áreas como la asistencia virtual, la automatización de procesos empresariales y la atención médica.

WORD2VEC

Word2vec

Es un modelo de aprendizaje profundo que se utiliza para **representar palabras como vectores** numéricos en un espacio de alta dimensión.

Fue desarrollado por Tomas Mikolov y su equipo en Google en 2013.

El objetivo de Word2vec es **aprender la representación semántica de las palabras utilizando técnicas de aprendizaje no supervisado**.

El modelo utiliza una red neuronal para aprender las representaciones de las palabras utilizando información estadística de un corpus de texto.

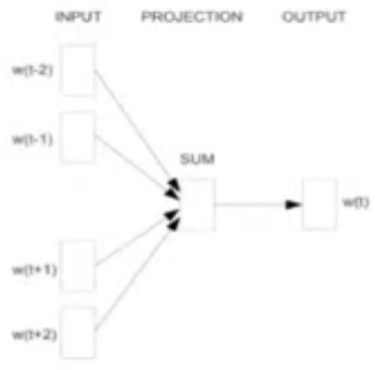
Word2vec tiene dos arquitecturas principales:

Continuous Bag of Words (CBOW) y Skip-Gram.

En el modelo CBOW, el objetivo es predecir la palabra central en una ventana de contexto de palabras circundantes.

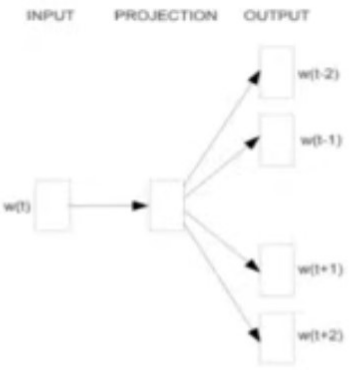
En el modelo Skip-Gram, el objetivo es predecir las palabras circundantes a partir de la palabra central.

https://datascientest.com/es/nlp-word-embedding-word2vec-es



CBOW

Predecir la palabra central



Skip-gram

Predecir el contexto

El proceso de entrenamiento de Word2vec implica alimentar el modelo con un corpus de texto y ajustar los parámetros de la red neuronal para **maximizar la probabilidad** de que una palabra dada aparezca en un contexto dado.

Después del entrenamiento, cada palabra se representa como un vector numérico en el espacio de alta dimensión, donde las palabras con significados similares están más cerca entre sí.

Sesgos

Extreme *she* occupations

1. homemaker	2. nurse	3. receptionist
4. librarian	5. socialite	6. hairdresser
7. nanny	8. bookkeeper	9. stylist
10. housekeeper	11. interior designer	12. guidance counselor

Extreme *he* occupations

1. maestro	2. skipper	3. protege
4. philosopher	5. captain	6. architect
7. financier	8. warrior	9. broadcaster
10. magician	11. figher pilot	12. boss

Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings (2016), Bolukbasi et al.

Dudas

- ¿Las palabras homónimas tienen el mismo embedding?
- ¿Cómo quitamos las stopwords?
- ¿Qué es mejor CBOW o Skip-Gram?
- El tamaño de la ventana de contexto determina cuántas palabras antes y después de una palabra se incluyen como palabras de contexto de la palabra dada. Según la nota de los autores, el valor recomendado es 10 para skip-gram y 5 para CBOW

Gensim

Gensim es una biblioteca de Python utilizada para modelar y analizar datos de texto utilizando técnicas de procesamiento de lenguaje natural (NLP). Fue desarrollado por Radim Řehůřek en 2008.

Gensim se utiliza principalmente para crear modelos de representación de documentos y palabras en vectores numéricos para su posterior análisis. La biblioteca implementa algoritmos populares de NLP, como Word2Vec.

Gensim también proporciona herramientas para preprocesamiento de texto, como tokenización, lematización y eliminación de stopwords. Además, la biblioteca ofrece una interfaz fácil de usar para trabajar con grandes conjuntos de datos de texto, lo que hace que el procesamiento de grandes volúmenes de datos sea más eficiente.

Además, la biblioteca es de código abierto y cuenta con una gran comunidad de usuarios y desarrolladores que contribuyen a su mejora continua.

Otra duda

https://stackoverflow.com/questions/57090378/is-google-word2vec-pertrained-model-cbow-or-skipgram

Nos toca prácticar…

RNN

Problema de CBOW y Skip-gram

"El concierto fue sumamente aburrido durante los primeros 15 minutos mientras la banda calentaba, pero luego fue tremendamente emocionante".

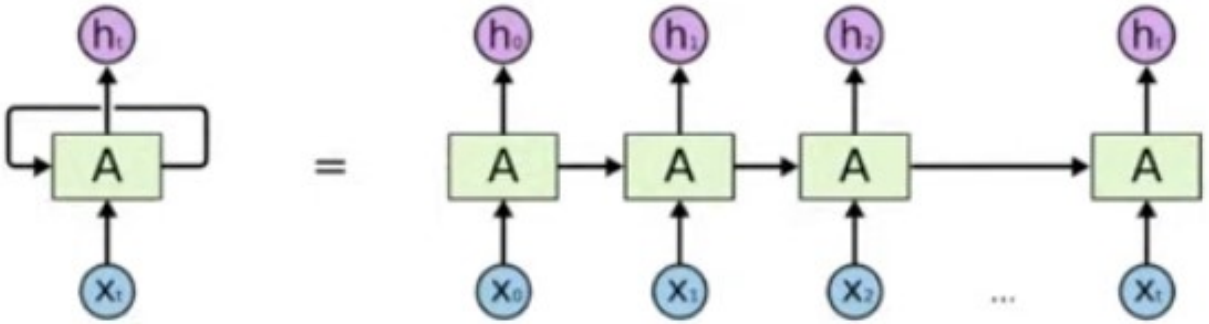
Red Neuronal Recurrente (RNN)

- Secuencias: frases, precios de acciones diarias, medidas de sensores
- Las RNNs procesan secuencias y retienen información en la “memoria” (estado)

Reproducción automática desactivada

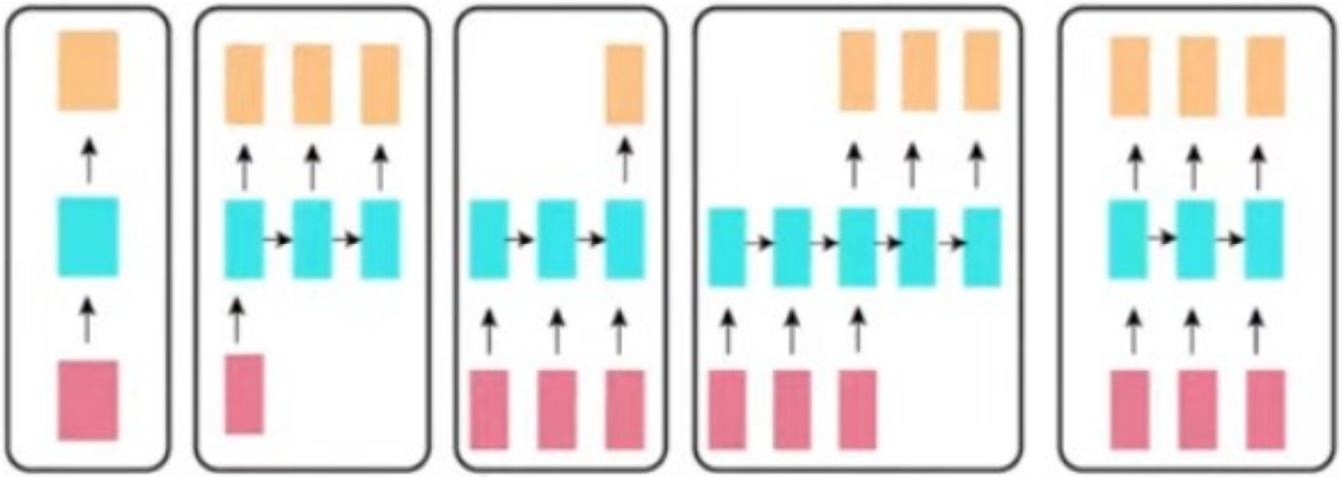
@NLP_en_ES

RNNs



@NLP_en_ES

Tipos de RNNs



@NLP_en_ES

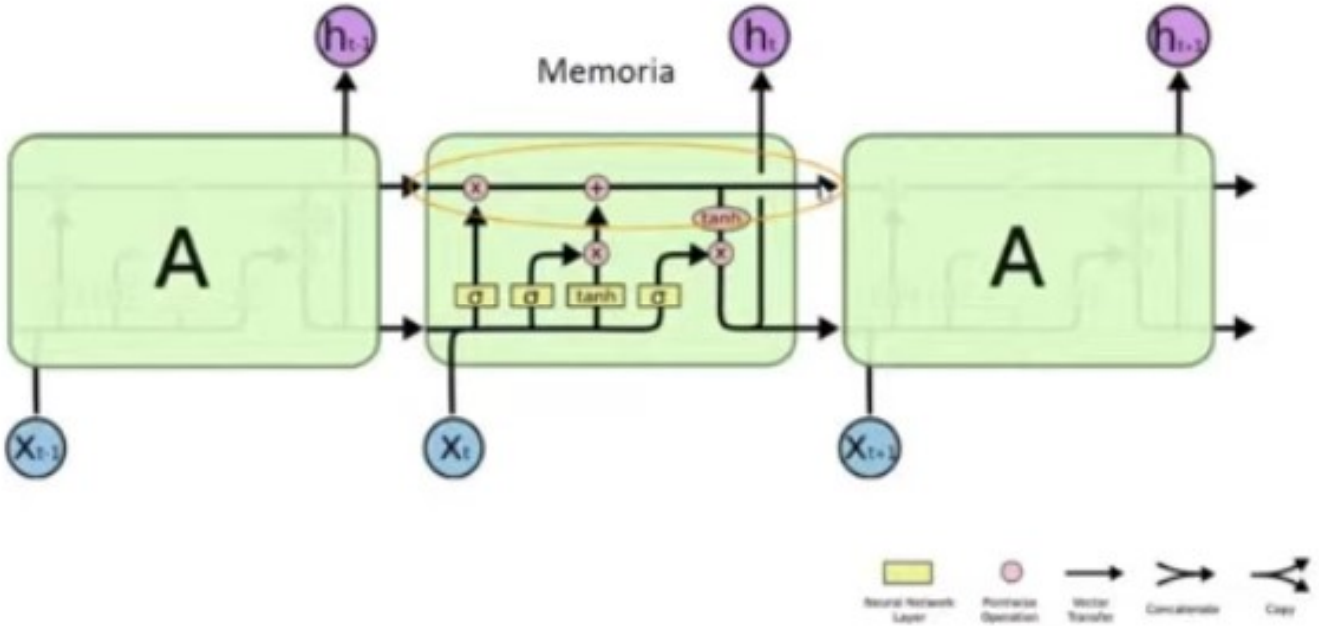
Problemas de las RNNs

- El **gradiente** indica el ajuste a realizar en los pesos con respecto a la variación en el error.
- "**Gradientes Explosivos**" o *Exploding Gradients:* el algoritmo asigna una importancia exageradamente alta a los pesos. En este caso el problema se puede resolver fácilmente truncando los gradientes *(Gradient Clipping).*
- "**Gradientes Desaparecidos**" o *Vanishing Gradients:* los valores de los gradientes son demasiado pequeños y el modelo deja de aprender o aprende muy despacio.
- Sucede a menudo cuando las **secuencias son relativamente largas.**

Problemas de las RNNs

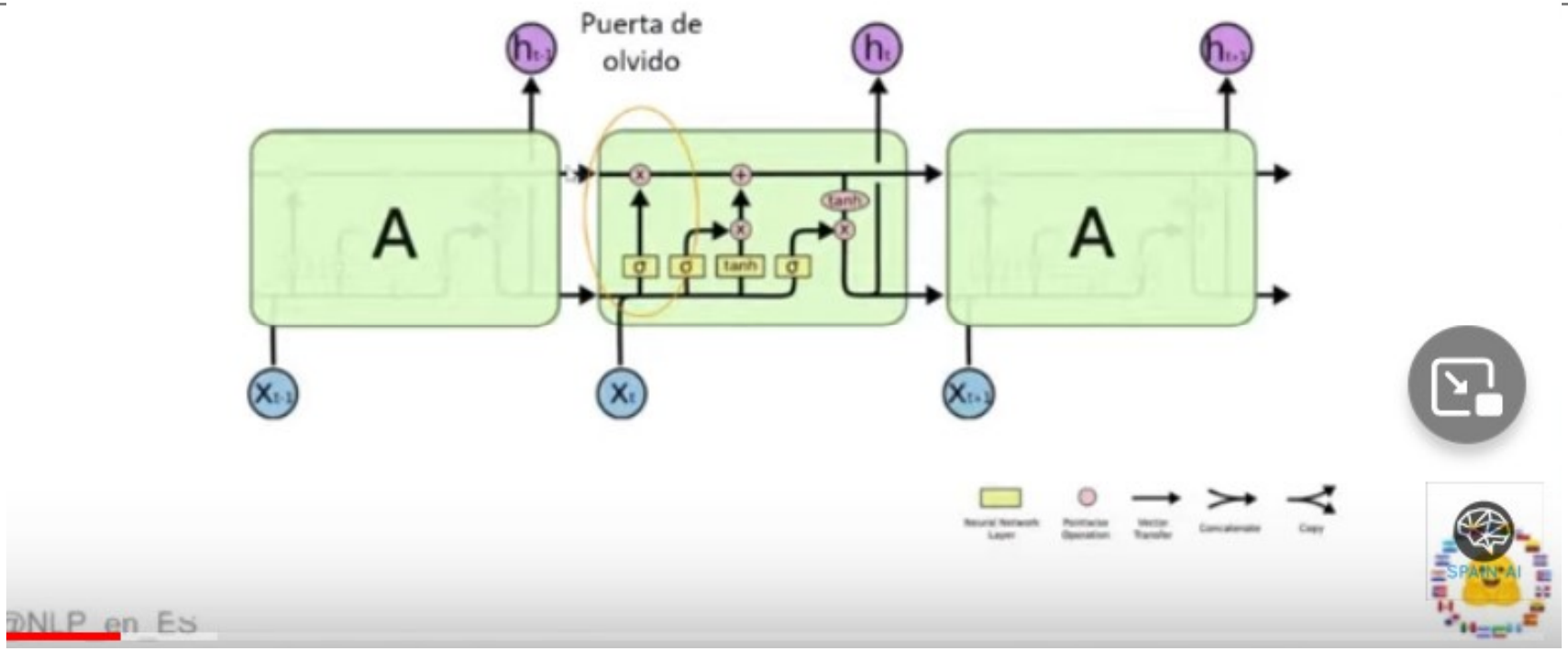
@NLP_en_ES

LSTMs

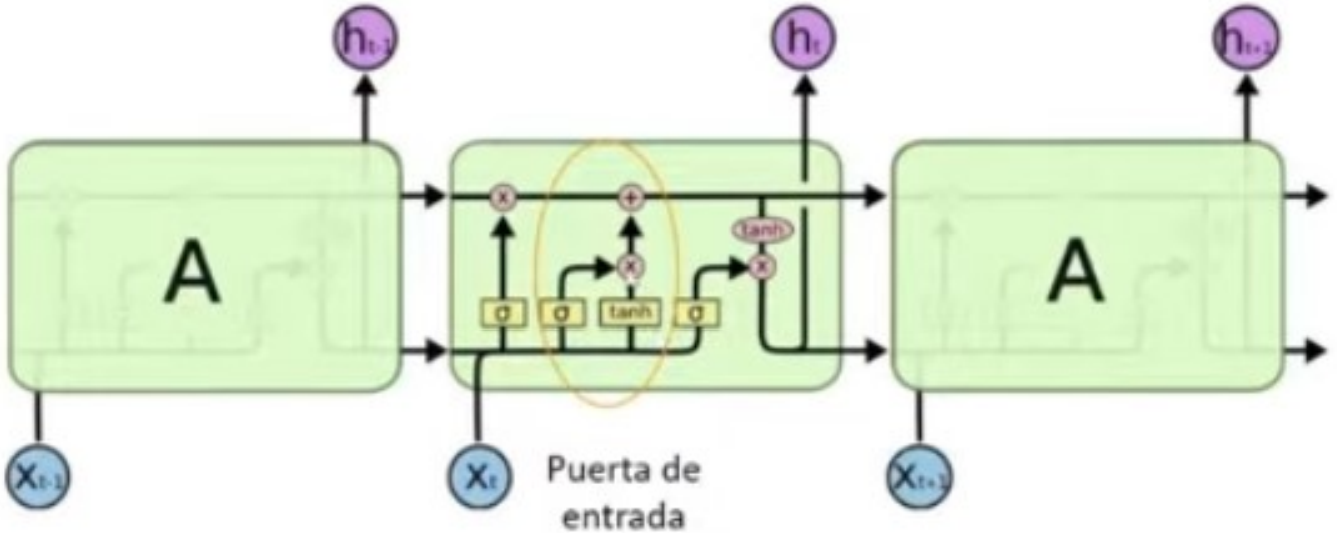


@NLP_en_Es

LSTMs

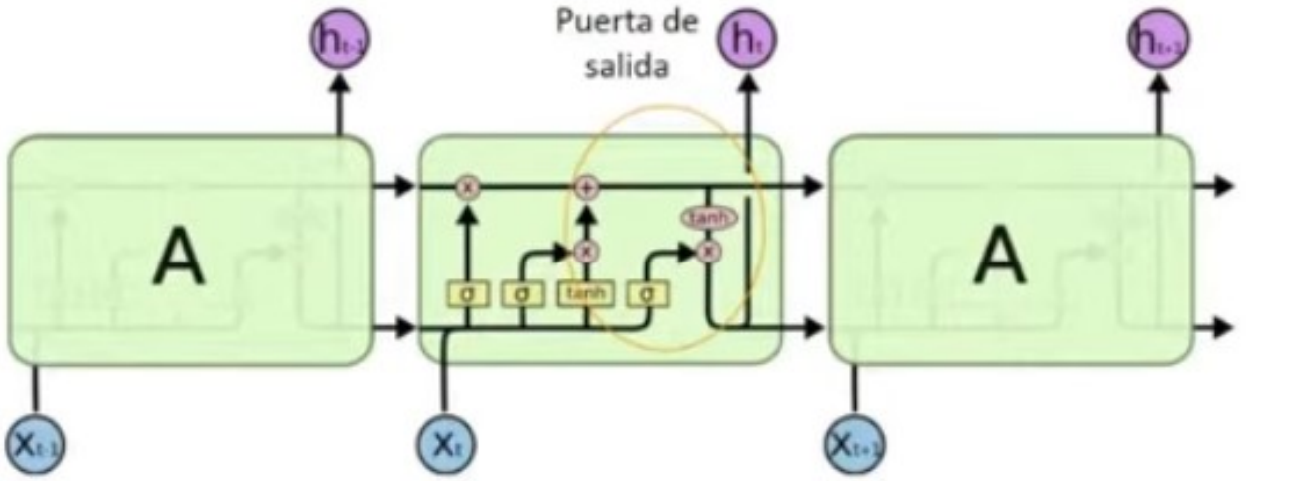


LSTMs



@NLP_en_Es

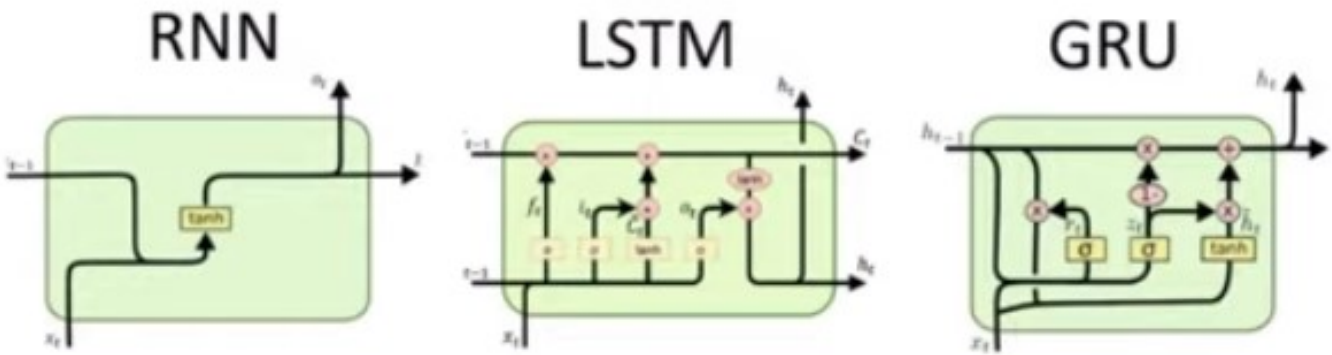
LSTMs



@NLP_en_Es

RNNS, LSTMs y GRUs

- **RNNs:** problemas relacionados con “Gradientes Desaparecidos”
- **LSTMs:** mantienen los valores de los gradientes suficientemente altos y, por lo tanto, el entrenamiento es más rápido y la precisión mejora
- **GRUs:** rendimiento similar, computacionalmente más eficientes



@NLP_en_ES

Conclusión

- Las LSTM son una solución muy prometedora para problemas relacionados con secuencias y series temporales. Pero...
- Los cálculos se hacen en serie lo que ralentiza mucho el entrenamiento.
- ¿Qué pasa con secuencias muy largas?

Resumen

Las RNN **(Redes Neuronales Recurrentes)** son un tipo de red neuronal que se utiliza para procesar **datos secuenciales**, como texto, audio, series temporales, entre otros. A diferencia de las redes neuronales convencionales, que procesan los datos de manera aislada y no tienen memoria del pasado**, las RNN tienen memoria y pueden procesar los datos secuenciales de manera iterativa.**

En una RNN, **cada unidad o celda tiene una entrada y una salida, así como una memoria interna.**

La salida de cada unidad se utiliza como entrada para la siguiente unidad de la secuencia, y la memoria interna se actualiza con información relevante del pasado. Esto permite a las RNN **capturar patrones y dependencias a lo largo del tiempo**, lo que las hace especialmente útiles para tareas de procesamiento de lenguaje natural, como el modelado de lenguaje, la traducción automática, el etiquetado de partes del discurso y el análisis de sentimientos.

Sin embargo, las RNN también tienen limitaciones, como la dificultad de entrenarlas para capturar patrones a largo plazo y la aparición de gradientes explosivos o desvanecidos durante el entrenamiento, lo que puede afectar el rendimiento del modelo.

Para superar estas limitaciones, se han propuesto diversas variantes de las RNN, como las **LSTM** (Memoria a Largo Plazo) y las **GRU** (Unidades de Recurrencia Gated), que han demostrado ser más efectivas en muchas aplicaciones de procesamiento de lenguaje natural.