

# Introducción a la minería de texto



# ¿Qué es la minería de texto?

---

## Minería de datos

- Extraer conocimiento útil a partir de datos en estado puro
- Se apoya en modelos matemáticos y estadísticos
- Usa tanto análisis descriptivo, modelos supervisados y modelos no supervisados.
- Ejemplos de metodologías son
  - Clustering
  - Clasificadores
  - Regresión
  - Reducción de dimensionalidad
  - Redes neuronales



## Minería de texto

- Extraer conocimiento útil a partir de texto en estado puro
- Se apoya en metodologías de la minería de datos
- Usa tanto análisis descriptivo, modelos supervisados y no supervisados
- Ejemplos de metodologías son
  - Análisis semántico latente
  - Análisis de sentimiento
  - TF-IDF
  - Representación vectorial de palabras



# Glosario de conceptos

---

- **Corpus:** Fuente de texto amplia, se suele utilizar para entrenamiento.
- **Token:** palabras idénticas en el texto
- **Tokenizador:** Algoritmo diseñado para extraer los tokens de un texto
- **Web Scraping:** extracción del texto contenido en el código fuente de una página web
- **Stemming:** Técnica que consiste en cortar palabras
- **Lema:** la raíz de la palabra
- **Lematizador:** Algoritmo para encontrar el lema de las palabras
- **Stopwords:** corresponden a artículos, preposiciones u otros caracteres
- **Representación vectorial:** Las palabras de acuerdo a un corpus transformadas en un vector de características.
- **NLP:** Procesamiento de lenguaje natural.
- **Análisis de sentimiento:** Técnica que busca predecir el sentimiento asociado a un texto.
- **Word embeddings:** Métodos para obtener Word vectors a partir del modelado de las palabras
- **Word vectors:** Representación en un vector de números reales de una palabra

# Trabajando textos

---

Un token es cada palabra idéntica en un texto. Por ejemplo la oración “*Mi amigo tiene un gato y una gata*” los tokens son ('Mi', 'Amigo', 'tiene', 'un', 'gato', 'y', 'una', 'gata')

Los tokens pueden ser obtenidos separando cada una de las palabras por un separador, por ejemplo un espacio.

El espacio es el convencional, sin embargo de cualquier forma va a depender de la estructura del texto, por ejemplo saltos de línea (\n) u otros caracteres pueden hacer engorrosa la labor de limpiar el texto.

# TF-IDF

En general quisiéramos saber más acerca de qué palabras están asociadas con un conjunto de texto. En análisis iniciales nos guiamos por la frecuencia de ocurrencia de los tokens, sin embargo esto puede llevar a análisis erróneos pues existen palabras que por muy frecuentes que sean, no aportan pues son inherentes al contexto. El análisis **TF-IDF** (*Text Frequency – Inverse Document Frequency*) es una medida que calcula la frecuencia de ocurrencia de una palabra, compensada por la cantidad de documentos (por ejemplo tweets, frases, etc) que contienen esa palabra.

Si  $t$  es el término buscado,  $\#D$  el número de documentos,  $\#W$  el número total de palabras, el TF-IDF se compone como.

$$TF = \frac{\#t}{\#W}$$
$$IDF = \log \frac{\#D}{1 + d \in D: t \in d}$$

$$TF - IDF = TF \times IDF$$

Existen variantes como un TF booleano o escalado o bien dividida la frecuencia por el número total de palabras.

# Análisis de sentimiento

El análisis de sentimiento es una metodología que busca determinar la intención o sentimiento del usuario en función al texto escrito. La forma más sencilla de plantear el problema es un clasificador usando la ocurrencia de variables como predictor.

Por ejemplo si tenemos un sentimiento positivo hacia la frase “Buen precio” y uno negativo hacia “mala atención”, los datos son como

Sentimiento	Buen	precio	mala	atención
1	1	1	0	0
0	0	0	1	1

Luego basta incorporarlo a un clasificador como un árbol de decisión, de esta forma se crea un mecanismo que decide la intención del usuario en función de las palabras que utiliza

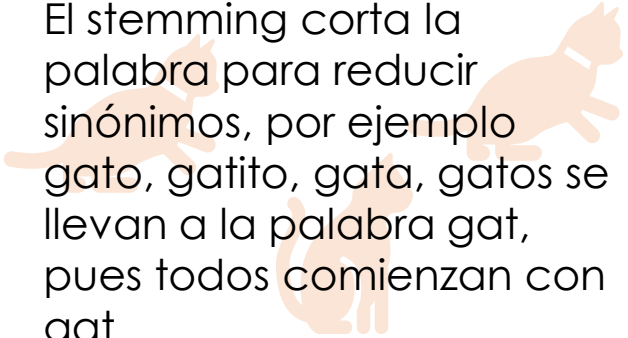
# Tratando el texto

---

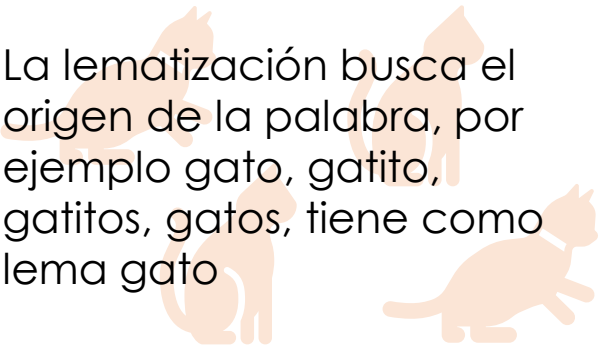
Dado que cada token constituirá una variable, es de vital importancia limpiar lo más posible el texto antes de entregarlo a un modelo. En general una pauta básica es:

- Eliminar caracteres de puntuación (#, @, %, etc)
- Eliminar espacios en blanco a todos los tokens
- Volver a analizar el texto
- Llevar todas las letras a minúsculas.
- Quitar los tildes de las palabras.
- Volver a analizar el texto
- Eliminar stopwords o tokens irrelevantes.
- Volver a analizar el texto.
- Aplicar stemming o lematización
- Volver a analizar el texto

Además es recomendable la extracción de características adicionales a la oración (sustantivos, adjetivos, complemento directo, verbos, etc.)



El stemming corta la palabra para reducir sinónimos, por ejemplo gato, gatito, gata, gatos se llevan a la palabra gat, pues todos comienzan con gat



La lematización busca el origen de la palabra, por ejemplo gato, gatito, gatitos, gatos, tiene como lema gato

# Análisis semántico latente (LSA)

---

Técnica que en función de los contextos o documentos en los que aparecen palabras, se busca obtener la dimensión o tema que relacionan estos conceptos en los documentos. El concepto es muy similar al PCA, en el sentido que busca obtener los temas subyacentes en las palabras. En efecto se buscan las palabras que tienen más peso en cada uno de los vectores que se generan para identificar el tema que las relacionan.

Para esto se usa la matriz de tópicos, donde las columnas son los documentos y las filas las palabras y se aplica la descomposición en valores singulares

$$M = U\Sigma V^*$$

$M$  es una matriz de  $m \times n$ , donde  $m$  es la cantidad de tokens,  $n$  la de documentos.  $U$  es una matriz de  $m \times m$ ,  $\Sigma$  es una matriz diagonal de  $m \times n$  y  $V$  es una matriz de  $m \times n$ . Los elementos de la diagonal de  $\Sigma$  se conocen como valores singulares, y las columnas de las otras matrices son los vectores singulares.

En la práctica se aplica de forma análoga al PCA con la descomposición espectral.





# Representación vectorial

---

Hasta ahora no hemos tomado algunos de los mayores desafíos en el análisis de texto, algunos de estos no tocados

- Qué pasa con los modismos?
- Cómo se trabaja el sarcasmo en los comentarios?
- Qué ocurre con palabras de contexto?
- Qué pasa si palabras no tienen lema?
- Cómo lidiar con las faltas de ortografía y lenguaje de RRSS?

Una forma de trabajar estos problemas es reenfocar el análisis y no trabajar directamente con las palabras, sino más bien con su contexto.

Como medio para llegar a esto se utilizan modelos que en un corpus, modelan la probabilidad de ocurrencia de cada palabra, dada las palabras que la rodean (o incluso símbolos, caracteres o emoticons), el vector de características obtenidos para modelar la probabilidad de ocurrencia de una palabra, se le conoce como **representación vectorial** de las palabras

# Representación vectorial

Se pueden obtener estos vectores de palabras de varias formas, sin embargo las más populares últimamente son las metodologías word2vec y globe. Veremos el word2vec, el cual consiste en modelar palabras o contexto en función del mismo texto.



Esto es sólo un ejemplo

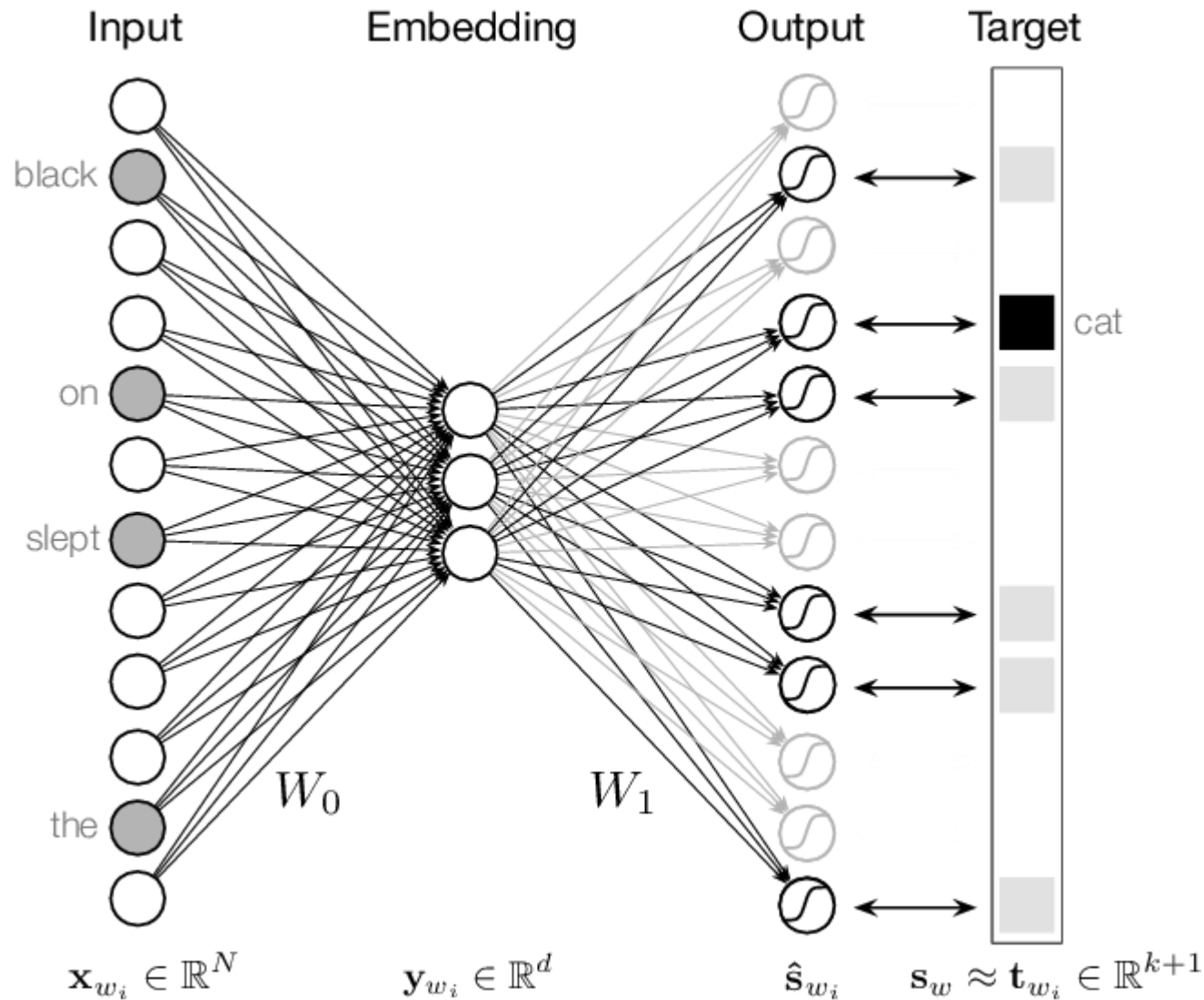
Cuando se modela la palabra del centro, se conoce como algoritmo CBOW (Continuous Bag Of Words), se toman las palabras Esto, es, un, ejemplo como predictores para la palabra sólo



Esto es sólo un ejemplo

Cuando se modelan las palabras de contexto en función de la del centro, el modelo se conoce como Skipgram. En este ejemplo sólo es predictor de las palabras Esto, es, un, ejemplo

# Representación vectorial



El modelo anterior se suele pasar a un perceptrón multicapa, cuya salida es la probabilidad de ocurrencia de las palabras objetivos. En la figura de la izquierda notar que los inputs son palabras de contexto y la salida son las probabilidades de ocurrencia de las palabras. Los Word vectors son la capa oculta de esta red neuronal, es decir si esos datos se los pasáramos a un clasificador como una regresión, tendría una buena capacidad de predecir nuestra palabra target.

# Aplicaciones

Dentro de las utilidades que tiene el vector de palabras es la comparación de similitudes entre palabras, en efecto al ser vectores si existe una distancia entre las palabras. Por lo general se usa la métrica de similitud coseno entre los vectores

$$\cos \theta = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|}$$

Esto se puede usar para hallar los términos que no corresponden con otros y también para buscar sinónimos a las palabras, como por ejemplo jergas, funcionando como traductor.

Notar que al ser vectores de valores continuo se puede aplicar clusters de palabras con mayor facilidad.

Por último tiene aplicaciones al análisis de sentimiento usando los vectores, sin embargo hay formas y formas de aplicarlo.

