Procesamiento de Lenguaje Natural (NLP)

Clase 2: Vectorización de Texto

Prof. D.Sc. BARSEKH-ONJI Aboud

Faculty of Engineering
Universidad Anáhuac México Sur

23 de octubre de 2025

Agenda de la Clase

1. Repaso y El Siguiente Paso
2. Modelo Bag-of-Words (Bolsa de Palabras)
3. Ponderación TF-IDF
4. Ejemplo Práctico en MATLAB
5. Conclusiones y Próximos Pasos

Agenda

1. Repaso y El Siguiente Paso
2. Modelo Bag-of-Words (Bolsa de Palabras)
3. Ponderación TF-IDF
4. Ejemplo Práctico en MATLAB
5. Conclusiones y Próximos Pasos

¿Dónde nos quedamos?

Resumen de la Clase 1

Aprendimos a tomar texto "sucio" y, mediante el **preprocesamiento**, lo convertimos en una lista de tokens limpios y estandarizados.

Texto Crudo

"Los alumnos estan aprendiendo..." $\downarrow$ *Limpieza* $\downarrow$ **Tokens Procesados**

```
'alumn', 'aprend', 'procesamiento', ...
```

El Problema Actual

Nuestros tokens son limpios, ¡pero siguen siendo palabras! Los modelos de Machine Learning no entienden palabras, solo entienden **números**.

El Puente entre Palabras y Matemáticas

Objetivo de Hoy: La Vectorización

El proceso de convertir texto en una representación numérica (vectores o matrices) se llama **vectorización** o *feature extraction*.

Tokens Limpios $\xrightarrow{\text{Vectorización}}$ Matriz Numérica

Hoy aprenderemos dos técnicas fundamentales para lograr esto: **Bag-of-Words** y **TF-IDF**.

Agenda

1. Repaso y El Siguiente Paso
2. Modelo Bag-of-Words (Bolsa de Palabras)
3. Ponderación TF-IDF
4. Ejemplo Práctico en MATLAB
5. Conclusiones y Próximos Pasos

Bag-of-Words (BoW): Una Idea Sencilla

Analogía

Imagina que tomas todas las palabras de un documento, las metes en una bolsa, las revuelves y luego cuentas cuántas veces aparece cada una. **El orden no importa, solo la frecuencia.**

El proceso tiene dos pasos clave:

1. **Crear un Vocabulario:** Construir una lista de todas las palabras únicas que aparecen en *todos* nuestros documentos.

Bag-of-Words (BoW): Una Idea Sencilla

Analogía

Imagina que tomas todas las palabras de un documento, las metes en una bolsa, las revuelves y luego cuentas cuántas veces aparece cada una. **El orden no importa, solo la frecuencia.**

El proceso tiene dos pasos clave:

1. **Crear un Vocabulario:** Construir una lista de todas las palabras únicas que aparecen en *todos* nuestros documentos.
2. **Contar Frecuencias:** Para cada documento, contar cuántas veces aparece cada palabra del vocabulario.

Bag-of-Words: Ejemplo Visual

Supongamos que tenemos dos documentos (después de preprocesar):

- **Doc 1:** "gato persigue raton"
- **Doc 2:** "perro persigue gato"

Bag-of-Words: Ejemplo Visual

Supongamos que tenemos dos documentos (después de preprocesar):

- **Doc 1:** "gato persigue raton"
- **Doc 2:** "perro persigue gato"

1. **Vocabulario único:** {gato, persigue, raton, perro}

Bag-of-Words: Ejemplo Visual

Supongamos que tenemos dos documentos (después de preprocesar):

- **Doc 1:** "gato persigue raton"
- **Doc 2:** "perro persigue gato"

1. **Vocabulario único:** {gato, persigue, raton, perro}
2. **Contar frecuencias para crear vectores:**

	gato	**persigue**	**raton**	**perro**
Doc 1	1	1	1	0
Doc 2	1	1	0	1

¡Hemos convertido el texto en una matriz numérica! Cada fila es un documento.

Limitaciones del Modelo BoW

El Problema: Todas las palabras pesan lo mismo

En el modelo BoW simple, la palabra "excelente" tiene la misma importancia que la palabra "producto" si ambas aparecen una vez.

Consideremos estas frases:

- "El producto es **excelente**."
- "El producto es **terrible**."

Intuitivamente, sabemos que "excelente" y "terrible" son más importantes para determinar la opinión que "producto".

¿Cómo podemos dar más peso a las palabras más significativas?

La respuesta es **TF-IDF**.

Agenda

TF-IDF: Dando Peso a lo Importante

Definición

TF-IDF significa *Term Frequency - Inverse Document Frequency*. Es una puntuación numérica que refleja qué tan importante es una palabra para un documento dentro de una colección de documentos.

La puntuación se compone de dos partes:

1. TF (Frecuencia del Término):

- ¿Qué tan seguido aparece la palabra en **un documento**?
- Si una palabra aparece mucho, es importante *para ese documento*.

2. IDF (Frecuencia Inversa de Documento):

- ¿Qué tan rara es la palabra en **toda la colección**?
- Si una palabra aparece en muchos documentos (como "producto"), es menos informativa y recibe un peso bajo.

La Intuición del TF-IDF

$$\text{Puntuación TF-IDF} = \text{TF} \times \text{IDF}$$

La Puntuación TF-IDF es alta cuando...

Una palabra aparece **muchas veces** dentro de un documento (*TF alto*), pero **pocas veces** en el resto de los documentos de la colección (*IDF alto*).

- Esto resalta las palabras que son **distintivas y características** de un documento en particular.
- Por ejemplo, en un artículo sobre Ïnteligencia Artificial", la palabra "neuronal"tendrá un TF-IDF alto. En un conjunto de noticias, la palabra "el" tendrá un TF-IDF muy bajo.

Agenda

1. Repaso y El Siguiente Paso
2. Modelo Bag-of-Words (Bolsa de Palabras)
3. Ponderación TF-IDF
4. Ejemplo Práctico en MATLAB
5. Conclusiones y Próximos Pasos

Creando una Matriz TF-IDF en MATLAB

Objetivo

Convertir un conjunto de comentarios de texto en una matriz numérica TF-IDF, lista para ser usada en un modelo de IA.

Creando una Matriz TF-IDF en MATLAB

```
% 1. Datos de ejemplo: 3 comentarios de un producto
textos = [
    "La bateria dura mucho, excelente producto.",
    "La pantalla es grande y la camara es buena.",
    "La bateria no dura nada, un producto terrible."
];

% 2. Preprocesamos los textos (usando los pasos de la Clase 1)
documentos = tokenizedDocument(lower(textos));
documentos = erasePunctuation(documentos);
listaStopWords = ["y", "la", "el", "es", "un", "una"];
documentos = removeWords(documentos, listaStopWords);
documentos = normalizeWords(documentos, 'Style', 'stem');

% 3. Crear el modelo de Bag-of-Words
bolsa = bagOfWords(documentos);

% 4. Convertir los documentos a una matriz TF-IDF
matrizTfidf = tfidf(bolsa);
```

Análisis del Resultado

Paso 1: El Vocabulario Creado

La función ‘bagOfWords‘ crea el vocabulario automáticamente.

```
>> disp(bolsa.Vocabulary)

'bateri'    'buen'    'camar'    'dur'    'excelent' ...
'grand'    'much'    'nad'    'pantall'    'product'    'terribl'
```

Análisis del Resultado

Paso 2: La Matriz Numérica Final

La función ‘tfidf‘ genera la matriz, donde cada fila es un comentario y cada columna es una palabra del vocabulario.

```
>> disp(full(matrizTfidf))

%       bateri    buen     camar     dur     ...
ans =
   0.2588      0       0     0.2588    ...   (Comentario 1)
        0   0.3536  0.3536        0    ...   (Comentario 2)
   0.2182      0       0     0.2182    ...   (Comentario 3)
```

Interpretando los Números de la Matriz

¿Qué significa cada número?

Cada valor en la matriz representa la **puntuación TF-IDF** de una palabra (columna) para un documento específico (fila). Un número más alto significa que la palabra es más importante o distintiva para ese documento.

Vamos a analizar un ejemplo:

- La palabra `'bateri'` tiene un valor alto en la Fila 1 y Fila 3, pero **cero** en la Fila 2.
  - **¿Por qué?** Porque la palabra "batería" solo aparece en el primer y tercer comentario. El cero indica su ausencia en el segundo.
- La palabra `'excelent'` (en la Fila 1) probablemente tendrá una puntuación TF-IDF más alta que `'product'`.
  - **¿Por qué?** Aunque ambas aparecen una vez (TF similar), `'product'` aparece en dos documentos, haciéndola menos única (IDF más bajo). En cambio, `'excelent'` es una palabra muy distintiva de ese primer comentario.

Interpretando los Números de la Matriz

La Gran Idea

No solo contamos palabras; hemos calculado su **relevancia ponderada**. Esto es lo que permite a un modelo de IA enfocarse en las palabras que realmente definen el significado de un texto.

Agenda

Resumen y Conclusiones

- **Vectorizar es Esencial:** Es el proceso de convertir palabras en números, un paso obligatorio para aplicar Machine Learning al texto.
- **Bag-of-Words (BoW):** Es un modelo simple y efectivo que representa documentos basándose en la frecuencia de las palabras.
- **TF-IDF:** Es una técnica de ponderación que mejora a BoW al dar más importancia a las palabras que son realmente distintivas de un documento.
- **Logro de Hoy:** ¡Ya sabemos cómo transformar texto crudo en una **matriz numérica significativa**!

Próximos Pasos

La Próxima Frontera: Aplicaciones Reales

Ahora que nuestros datos de texto están en un formato que una máquina puede "entender", podemos empezar a hacer tareas útiles con ellos.
En la próxima etapa, se dará un primer paso en las aplicaciones de NLP: realizando un **Análisis de Sentimientos** para clasificar automáticamente si un comentario es positivo, negativo o neutral.