Pontificia Universidad JAVERIANA Colombia

Procesamiento de Lenguaje Natural

Tópicos Avanzados en Analítica

Maestría en Analítica para la Inteligencia de Negocios

Sergio Alberto Mora Pardo - H2 2023

Procesamiento de Lenguaje Natural (NLP)

Clase 1 - Palabras a números (I)

Contenido:

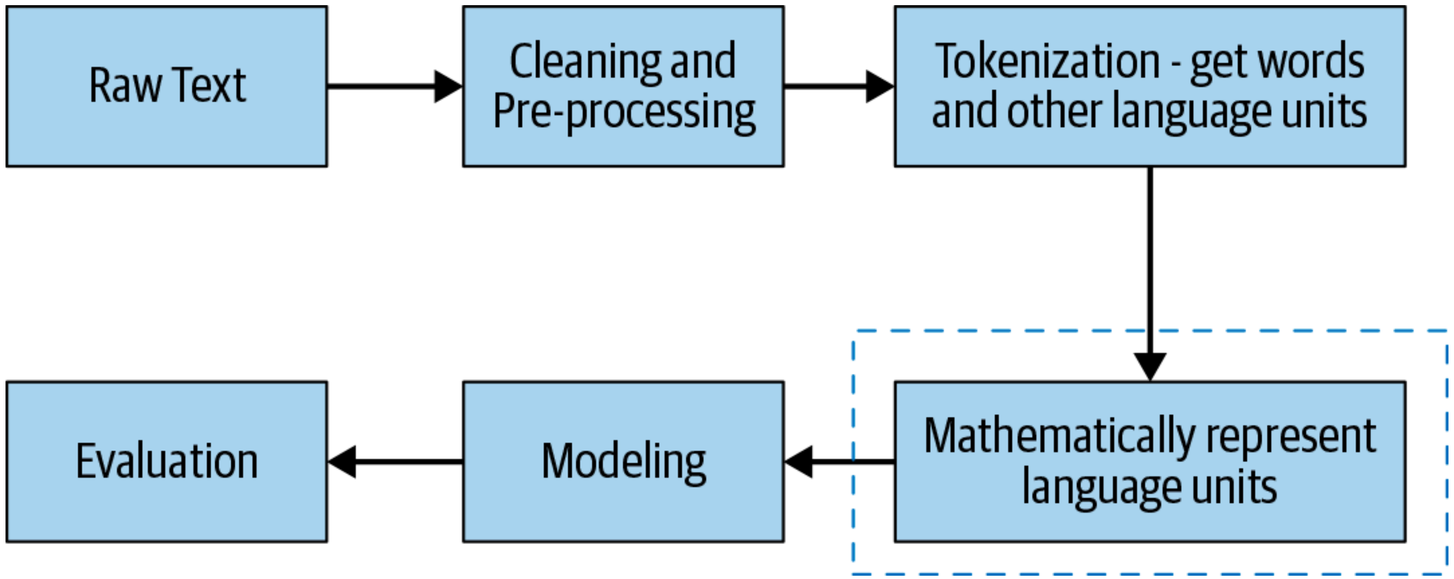
1. Modelos de Espacio Vectorial
2. Enfoques Básicos de Vectorización
3. Word Embedding
4. Embedding Visualization

Pontificia Universidad
JAVERIANA
Colombia

Text representation

Representación de Texto

Feature Engineering



Representación de Texto

Feature Engineering

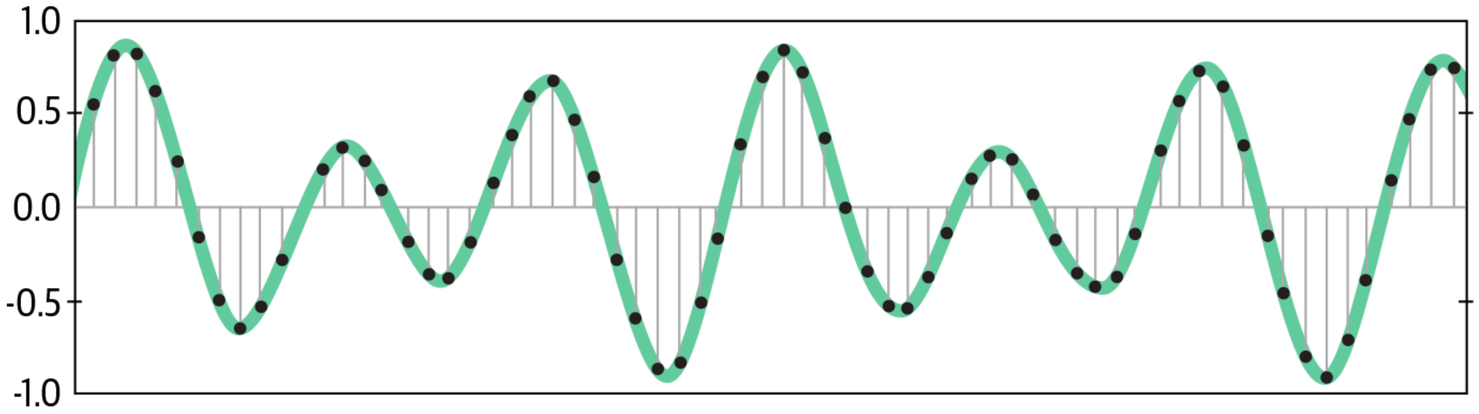
What We See

```
08 02 22 97 38 15 00 40 00 75 04 05 07 78 52 12 50 77 91 08
49 49 99 40 17 81 18 57 60 87 17 40 98 43 69 48 04 56 62 00
81 49 31 73 55 79 14 29 93 71 40 67 53 88 30 03 49 13 36 65
52 70 95 23 04 60 11 42 69 24 68 56 01 32 56 71 37 02 36 91
22 31 16 71 51 67 63 89 41 92 36 54 22 40 40 28 66 33 13 80
24 47 32 60 99 03 45 02 44 75 33 53 78 36 84 20 35 17 12 50
32 98 81 28 64 23 67 10 26 38 40 67 59 54 70 66 18 38 64 70
67 26 20 68 02 62 12 20 95 63 94 39 63 08 40 91 66 49 94 21
24 55 58 05 66 73 99 26 97 17 78 78 96 83 14 88 34 89 63 72
21 36 23 09 75 00 76 44 20 45 35 14 00 61 33 97 34 31 33 95
78 17 53 28 22 75 31 67 15 94 03 80 04 62 16 14 09 53 56 92
16 39 05 42 96 35 31 47 55 58 88 24 00 17 54 24 36 29 85 57
86 56 00 48 35 71 89 07 05 44 44 37 44 60 21 58 51 54 17 58
19 80 81 68 05 94 47 69 28 73 92 13 86 52 17 77 04 89 55 40
04 52 08 83 97 35 99 16 07 97 57 32 16 26 26 79 33 27 98 66
88 36 68 87 57 62 20 72 03 46 33 67 46 55 12 32 63 93 53 69
04 42 16 73 38 25 39 11 24 94 72 18 08 46 29 32 40 62 76 36
20 69 36 41 72 30 23 88 34 62 99 69 82 67 59 85 74 04 36 16
20 73 35 29 78 31 90 01 74 31 49 71 48 86 81 16 23 57 05 54
01 70 54 71 83 51 54 69 16 92 33 48 61 43 52 01 89 19 67 48
```

What Computers See

Representación de Texto

Feature Engineering




```
[-1274, -1252, -1160, -986, -792, -692, -614, -429, -286, -134, -57, -41,
-169, -456, -450, -541, -761, -1067, -1231, -1047, -952, -645, -489, -448,
-397, -212, 193, 114, -17, -110, 128, 261, 198, 390, 461, 772, 948, 1451,
1974, 2624, 3793, 4968, 5939, 6057, 6581, 7302, 7640, 7223, 6119, 5461,
4820, 4353, 3611, 2740, 2004, 1349, 1178, 1085, 901, 301, -262, -499,
-488, -707, -1406, -1997, -2377, -2494, -2605, -2675, -2627, -2500, -2148,
-1648, -970, -364, 13, 260, 494, 788, 1011, 938, 717, 507, 323, 324, 325,
350, 103, -113, 64, 176, 93, -249, -461, -606, -909, -1159, -1307, -1544]
```

Text Representation

Feature Engineering

1. Space Vector Models

   Todas las representaciones de texto son SVM.

2. Basic Vectorization Approaches

3. Word Embeddings

4. Visualizing Embeddings***

***opcional

Text Representation

Space Vector Models

Space Vector Models

Representación de unidades de texto

↓

Vectores numéricos

(Modelo matemático o algebraico)

Caracteres

Fonemas

Documentos

Palabras

Frases

Oraciones

Párrafos

Text Representation

Space Vector Models

Space Vector Models

Representación de unidades de texto

↓

Vectores numéricos

(Modelo matemático o algebraico)

Distancia del coseno:

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{||\mathbf{A}||_2 ||\mathbf{B}||_2} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} \mathrm{A}_i^2} \sqrt{\sum_{i=1}^{n} \mathrm{B}_i^2}}$$

Donde A_i y B_i son las i-ésimas componentes de los vectores A y B.

Text Representation

Feature Engineering

1. Space Vector Models

   Todas las representaciones de texto son SVM.

2. Basic Vectorization Approaches

3. Word Embeddings

4. Visualizing Embeddings***

***opcional

Text Representation

Basic Vectorization Approaches

Basic Vectorization Approaches

Ej. asignación de cada palabra en el vocabulario (V) del corpus de texto aun ID única (valor entero)

D1	Perro muerde a hombre.
D2	Hombre muerde a perro.
D3	El perro come carne.
D4	El hombre come comida.

Vocabulario del corpus:

[perro, muerde, hombre, come, carne, comida]

↓

Vector de tamaño 6

Texto procesado:
minúsculas, sin puntuación, etc..

Texto tokenizado:
Cadena de texto dividida en tokens.

Text Representation

Basic Vectorization Approaches

One-Hot Encoding

Vocabulario basado en asignación de Id por palabras.

W *Palabra del texto*

V *Conjunto del vocabulario del corpus*

W_{id} *Indice de palabras del texto, donde W_{id} entre {1, |V|}*

Representación binaria de cada palabra en un vector de tamaño |V|

Index = W_{id}

Text Representation

Basic Vectorization Approaches

One-Hot Encoding

Vocabulario basado en asignación de Id por palabras.

W *Palabra del texto*

V *Conjunto del vocabulario del corpus*

W_{id} *Indice de palabras del texto, donde W_{id} entre {1, |V|}*

perro = 1, muerde = 2, hombre = 3, carne = 4, comida = 5, come = 6

Ej.: D1: “perro muerde a hombre”

[[1 0 0 0 0 0] perro
[0 1 0 0 0 0] muerde
[0 0 1 0 0 0]] hombre

Ej.: D4: “hombre come comida”

[[0 0 1 0 0 0] hombre
[0 0 0 0 1 0] comida
[0 0 0 0 0 1]] come

Text Representation

Basic Vectorization Approaches

One-Hot Encoding

Vocabulario basado en asignación de Id por palabras.

Pros

Intuitiva y fácil de implementar

Text Representation

Basic Vectorization Approaches

One-Hot Encoding

Vocabulario basado en asignación de Id por palabras.

Pros

Intuitiva y fácil de implementar

Contra

1. Tamaño del vector proporcional al tamaño del vocabulario

2. No proporcional una longitud fija intra documentos.

3. Palabras como unidades atómicas y no tiene noción de (des)similitud.

Tienen la misma distancia: [correr, corre, manzana]

4. No maneja un esquema fuera del vocabulario (OOV).

Ej. Pasarle la palabra 'fruta' al modelo.

Text Representation

Basic Vectorization Approaches

Bag of Words (BOW)

Calificamos cada palabra en V por su recuento de ocurrencias en el documento.

↓

Recuento de palabras

W *Palabra del texto*

V *Conjunto del vocabulario del corpus*

W_{id} *Indice de palabras del texto, donde W_{id} entre {1, |V|}*

perro = 1, muerde = 2, hombre = 3, carne = 4, comida = 5, come = 6

Ej.: D1: “perro muerde a hombre”

[1 1 1 0 0 0]

perro
muerde
hombre

Ej.: D4: “hombre come comida”

[0 0 1 0 1 1]

hombre
comida
come

Text Representation

Basic Vectorization Approaches

Bag of Words (BOW)

Calificamos cada palabra en V por su recuento de ocurrencias en el documento.

Recuento de palabras

Pros

1. Intuitiva y fácil de implementar

2. Captura la similitud semántica de los documentos → *Ej. espacio euclidiano entre D_1 y D_2 es 0. En comparación con D_1 y D_4 que es 2.*

3. Codificación de longitud fija para cualquier oración arbitraria

Text Representation

Basic Vectorization Approaches

Bag of Words (BOW)

Calificamos cada palabra en V por su recuento de ocurrencias en el documento.

↓

Recuento de palabras

Contra

1. Tamaño del vector aumenta con el tamaño del vocabulario. → *Hace necesario limitar vocabulario a **n** número de palabras más frecuentes.*

2. No Captura la similitud semántica de palabras que significan lo mismo → *Ej.: “I ran”, ”I run” y ”I ate”*

3. No tiene forma de manejar palabras fuera del vocabulario.

4. El orden de las palabras se pierde → *D1 y D2 tendrán la misma representación en este esquema.*

Text Representation

Basic Vectorization Approaches

"n-gram feature selection."

N-grams

*Dividiremos el texto en grasientos de **n** palabras contiguas (o tokens).*

Captura de contexto

Recuento de palabras

Vocabulario del corpus:

{perro muerde, muerde hombre, hombre muerde, muerde perro, perro come, come carne, hombre come, come comida}

Vocabulario anterior:

[perro, muerde, hombre, come, carne, comida]

Vector de tamaño 8

Texto procesado:
minúsculas, sin puntuación, etc..

Texto tokenizado:
Cadena de texto dividida en tokens.

Text Representation

Basic Vectorization Approaches

"n-gram feature selection."

N-grams

*Dividiremos el texto en grasientos de **n** palabras contiguas (o tokens).*

Captura de contexto

Recuento de palabras

W *Fragmento de **n** palabras*

V *Conjunto de fragmentos del corpus*

W_{id} *Indice de fragmentos del texto, donde W_{id} entre {1, |V|} -> ej. bigramas*

{perro muerde, muerde hombre, hombre muerde, muerde perro, perro come, come carne, hombre come, come comida}

Ej.: D1: "perro muerde a hombre"

[1,1,0,0,0,0,0,0]

perro
muerde
hombre

Ej.: D4: "hombre muerde a perro"

[0,0,1,1,0,0,0 ,0]

hombre
muerde
perro

Text Representation

Basic Vectorization Approaches

"n-gram feature selection."

N-grams

*Dividiremos el texto en grasientos de **n** palabras contiguas (o tokens).*

Captura de contexto

Recuento de palabras

Pros y Contras

1. Captura alguna información del contexto y orden de palabras.

2. Espacio vectorial captura similitud semántica.

*3. A medida que aumenta **n**, la dimensionalidad solo aumenta.*

4. Aún sin una forma de abordar el problema de OOV.

Text Representation

Basic Vectorization Approaches

t : *Término*

d : *Documento*

TF-IDF

Término de frecuencia - frecuencia de documento inversa, cuantifica la importancia de una palabra en relación con otras palabras en el documento y en el corpus.

↓

Importancia de palabras

Frecuencia de término (TF).

Frecuencia de un término o una palabra en un documento.

$$\mathrm{TF}\left(t, d\right) = \frac{(\text{Number of occurrences of term } t \text{ in document } d)}{(\text{Total number of terms in the document } d)}$$

Frecuencia de documento inversa (IDF).

Mide la importancia de término en el corpus

$$\mathrm{IDF}\left(t\right) = \log_e \frac{(\text{Total number of documents in the corpus})}{(\text{Number of documents with term } t \text{ in them })}$$

Text Representation

Basic Vectorization Approaches

TF-IDF

Término de frecuencia - frecuencia de documento inversa, cuantifica la importancia de una palabra en relación con otras palabras en el documento y en el corpus.

↓

Importancia de palabras

Word	TF score	IDF score	TF-IDF score
dog	⅓ = 0.33	$\log_2(4/3) = 0.4114$	0.4114 * 0.33 = 0.136
bites	⅙ = 0.17	$\log_2(4/2) = 1$	1* 0.17 = 0.17
man	0.33	$\log_2(4/3) = 0.4114$	0.4114 * 0.33 = 0.136
eats	0.17	$\log_2(4/2) = 1$	1* 0.17 = 0.17
meat	1/12 = 0.083	$\log_2(4/1) = 2$	2* 0.083 = 0.17
food	0.083	$\log_2(4/1) = 2$	2* 0.083 = 0.17

TF-IDF = TF * IDF

La representación del vector TF-IDF es el mismo puntaje de TF-IDF, para cada término en ese documento.

Ejemplo, D1:

Perro	muerde	hombre	come	carne	alimento
0.136	0.17	0.136	0	0	0

scikit-learn.org. TFIDF vectorizer documentation. Last accessed June 15, 2020.

Text Representation

Basic Vectorization Approaches

TF-IDF

Término de frecuencia - frecuencia de documento inversa, cuantifica la importancia de una palabra en relación con otras palabras en el documento y en el corpus.

↓

Importancia de palabras

Pros y Contras

1. *Mejor que los otros métodos de vectorización vistos anteriormente.*
2. *Sufre de la maldición de la alta dimensionalidad.*

TF-IDF en SKLEARN:

1. *Ligera modificación en la fórmula IDF.*
2. *Disposiciones para dividir por cero.*
3. *No ignora por completo los términos que aparecen en todos los documentos.*

scikit-learn.org. TFIDF vectorizer documentation. Last accessed June 15, 2020.

Text Representation

Basic Vectorization Approaches

Representaciones discretas:

Tratan a las unidades del lenguaje (palabras, n-grams, etc.) como unidades atómicas.

Lo que dificulta la capacidad para captar relaciones entre palabras.

Dimensionalidad:

Los vectores generalmente son dispersos y de alta dimensión.

La dimensionalidad aumenta on el tamaño del vocabulario, siendo la mayoría de los valores cero.

1. *Dificulta la capacidad de aprendizaje.*
2. *Alta dimensionalidad, hace ineficientes los modelos*

OOV:

No pueden manejar palabras fuera del vocabulario (OOV).

Text Representation

Feature Engineering

1. Space Vector Models
2. Basic Vectorization Approaches
3. Word Embeddings
4. Visualizing Embeddings***

***opcional