



INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE COMPUTO



Nombre de los alumnos: SANDOVAL MUÑOZ EMANUEL, TREJO

SIERRA HECTOR, VALERIO LOPEZ JOSE EDUARDO

Matriculas: 2022670104, 2021630065, 2022670002

Maestr@: CAMACHO VAZQUEZ VANESSA ALEJANDRA

Materia: MACHINE LEARNING

Grupo: 6CV1

Trabajo: Construcción de características textuales en Python

Código:

Primer Código:

```
from sklearn.feature_extraction.text import TfidfVectorizer

# Datos de ejemplo (lista de documentos)
corpus = ['Este es un ejemplo de texto',
          'Otro ejemplo de texto',
          'Algunos ejemplos son buenos' ]

# Crear un objeto TfidfVectorizer
tfidf_vectorizer = TfidfVectorizer(ngram_range=(1,2)) #considera...
# Ajustar el vectorizador y transformar los datos de texto
tfidf_matrix = tfidf_vectorizer.fit_transform(corpus)
# Obtener el vocabulario (características)
feature_names = tfidf_vectorizer.get_feature_names_out()
# Imprimir el vocabulario
print("Vocabulario (Características):")
print(feature_names)
# Imprimir la matriz TF-IDF
print("\nMatriz TF-IDF:")
print(tfidf_matrix.toarray())
```

Ejecución:

```
Vocabulario (Características):
['algunos' 'algunos ejemplos' 'buenos' 'de' 'de texto' 'ejemplo'
 'ejemplo de' 'ejemplos' 'ejemplos son' 'es' 'es un' 'este' 'este es'
 'otro' 'otro ejemplo' 'son' 'son buenos' 'texto' 'un' 'un ejemplo']

Matriz TF-IDF:
[[0.         0.         0.         0.25504351 0.25504351 0.25504351
  0.25504351 0.         0.         0.33535157 0.33535157 0.33535157
  0.33535157 0.         0.         0.         0.         0.25504351
  0.33535157 0.33535157]
 [0.         0.         0.         0.34385143 0.34385143 0.34385143
  0.34385143 0.         0.         0.         0.         0.
  0.         0.45212331 0.45212331 0.         0.         0.34385143
  0.         0.         ]
 [0.37796447 0.37796447 0.37796447 0.         0.         0.
  0.         0.37796447 0.37796447 0.         0.         0.
  0.         0.         0.         0.37796447 0.37796447 0.
  0.         0.         ]]
```

Procesamiento de tweets

1	===== Frase 1 tokens =====		
2	id: 1	Palabra: te	Lema: tú
3	id: 2	Palabra: odio	Lema: odio
4	id: 3	Palabra: tren	Lema: tren
5	id: 6	Palabra: te	Lema: tú
6	id: 7	Palabra: odio	Lema: odio
7	id: 11	Palabra: odio	Lema: odio
8	id: 13	Palabra: todos	Lema: <i>todo</i>
9	id: 16	Palabra: hora	Lema: hora
10	id: 18	Palabra: ni	Lema: ni
11	id: 19	Palabra: viajé	Lema: viajar
12	id: 22	Palabra: 2014-05-06	Lema: 2014-05-06
13	id: 23	Palabra: 15:33:16	Lema: 15:33:16
14	id: 24	Palabra: ya	Lema: ya
15	id: 25	Palabra: pasaron	Lema: pasar
16	id: 26	Palabra: más	Lema: más
17	id: 28	Palabra: 4	Lema: 4
18	id: 29	Palabra: personas	Lema: persona
19	id: 32	Palabra: pelotudo	Lema: pelotudo
20	id: 33	Palabra: sigue	Lema: seguir
21	id: 34	Palabra: haciendo	Lema: hacer
22	id: 35	Palabra: 6	Lema: 6
23	id: 36	Palabra: juegos	Lema: juego
24	id: 38	Palabra: fotocopias	Lema: fotocopia
25	===== Frase 2 tokens =====		
26	id: 2	Palabra: puta	Lema: puta
27	id: 3	Palabra: madre	Lema: madre

Código de procesamiento

```
import stanza
import re

stanza.download('es') # Descargar el modelo en español

nlp = stanza.Pipeline(lang='es', processors='tokenize,mwt,pos,lemma')
#Eliminación de menciones
patron_1= re.compile(r'@[A-Za-z][A-Za-z0-9_]*?')
#Eliminación de URL
patron_2= re.compile(r'(http|https)://[A-Za-z0-9_-]*[A-Za-z]*((/[A-Za-z0-9_]*)?)?')
#Patron para eliminar caracteres especiales
patron_3= re.compile(r'[:\~\^\(\)\_\.\!;\.\!>!\(\)\_<\/>\/\,\.\!-\$%\#\!')
#Patron para palabras vacias
patron_4 = re.compile(r'\b(v|e|o|u|con|de|en|por|para|a|e|l|la|los|las|un|una|unos|unas|al|del)\b')
#Diccionario para emojis
# Diccionario para convertir emojis en palabras
emojis_a_palabras = {
    ':-/': 'inseguro',
    't.t': 'llorando',
    ':)': 'feliz',
    ':(': 'triste',
    '-.-': 'disconforme',
    ':b': 'sacar_lengua',
    ':o': 'sorprendido',
    ':f': 'ceño',
    ':s': 'inseguro',
    ':p': 'sacar_lengua'
}

# Ruta al archivo de texto
archivo = r"C:\Users\lemanu\Documents\Escuela\PROGRAMACION\python\Machine\Construcción de vectores TF-IDF\tweets_emo_negativas.txt"
ar_salida="resultado.txt"
```

```
# Leer el contenido del archivo
with open(archivo, 'r', encoding='utf-8') as file:
    texto = file.read().lower()

# Reemplazar emojis por palabras
for emoji, palabra in emojis_a_palabras.items():
    texto = re.sub(re.escape(emoji), palabra, texto)

# Procesar el texto con Stanza
doc = nlp(texto)

# Abrir el archivo resultados.txt en modo escritura
with open("TWEETS_EMOCIONES_NEGATIVAS.txt", "w", encoding="utf-8") as archivo_resultados:
    # Definimos una variable de número de oración
    noracion = 0
    for sent in doc.sentences:
        noracion += 1
        archivo_resultados.write(f'==== Frase {noracion} tokens ==== \n')
        for word in sent.words:
            #Guardamos la línea como se solicita en el documento id - palabra - lema
            #Para quitar las menciones
            if patron_1.match(word.text):
                continue
            if patron_2.match(word.text):
                continue
            if patron_3.match(word.text):
                continue
            if patron_4.match(word.text):
                continue
            archivo_resultados.write(f'id: {word.id}.\ljust(8) + f'Palabra: {word.text}.\ljust(25) + f'Lema: {word.lemma}.\n')
```

VECTORES DE VALORES TF-IDF DE BIGRAMAS

Construcción de vectores TF-IDF · vectores_tweets_BI_emonegativas.txt

Pr...

truccion.py × vectores_tweets_BI_emonegativas.txt × Codigo2.py × construccion_2.py × tweets_emo_negativas.txt ×

> .venv library root

Codigo2.py 02/05/2024 08
Construccion.py 02/05/20
construccion_2.py 02/05/2
tweets_emo_negativas.txt
TWEETS_EMOCIONES_NE
vectores_tweets_BI_emone

External Libraries
Scratches and Consoles

```
00 06,00 11,00 18,00 20,00 25,00 32,00 35,00 36,00 44,00 46,00 54,00 55,00 lema,01 10,01 24,01 29,01 36,01 42,01 44,01
45,01 50,01 lema,02 09,02 16,02 18,02 21,02 22,02 23,02 28,02 38,02 39,02 50,03 00,03 01,03 03,03 07,03 13,03 20,03 29,03
33,03 34,03 42,03 46,03 49,03 53,03 57,03 lema,04 01,04 03,04 27,04 29,04 31,04 35,04 53,05 06,05 07,05 08,05 09,05 10,05
11,05 13,05 14,05 15,05 16,05 17,05 18,05 19,05 20,05 21,05 22,05 23,05 25,05 27,05 29,05 30,05 31,05 53,05 lema,06 06,06
01,06 10,06 22,06 23,06 25,06 26,06 28,06 39,06 46,06 lema,07 05,07 12,07 17,07 52,07 lema,08 18,08 36,08 lema,09 17,09
24,09 25,09 lema,10 20,10 30,10 32,10 34,10 lema,10 palabra,10 tokens,100 palabra,101 palabra,102 palabra,103 palabra,104
palabra,105 palabra,106 palabra,107 palabra,108 palabra,109 palabra,11 18,11 21,11 35,11 41,11 46,11 lema,11 palabra,11
tokens,110 palabra,111 palabra,112 palabra,113 palabra,114 palabra,115 palabra,116 palabra,117 palabra,118 palabra,119
palabra,12 03,12 05,12 51,12 57,12 lema,12 palabra,12 tokens,120 palabra,121 palabra,122 palabra,123 palabra,124
palabra,125 palabra,126 palabra,127 palabra,128 palabra,129 palabra,13 16,13 23,13 25,13 31,13 35,13 41,13 57,13 lema,13
palabra,13 tokens,130 palabra,131 palabra,132 palabra,133 palabra,134 palabra,135 palabra,136 palabra,137 palabra,138
palabra,139 palabra,14 41,14 49,14 54,14 lema,14 palabra,14 tokens,140 palabra,141 palabra,142 palabra,143 palabra,144
palabra,145 palabra,146 palabra,147 palabra,148 palabra,149 palabra,15 02,15 04,15 11,15 12,15 23,15 26,15 27,15 28,15
33,15 39,15 43,15 51,15 56,15 lema,15 palabra,15 tokens,150 palabra,151 palabra,152 palabra,153 palabra,154 palabra,155
palabra,156 palabra,157 palabra,158 palabra,159 palabra,16 00,16 01,16 05,16 08,16 16,16 18,16 20,16 22,16 27,16 43,16
45,16 46,16 48,16 53,16 59,16 lema,16 palabra,16 tokens,160 palabra,161 palabra,162 palabra,163 palabra,164 palabra,165
palabra,166 palabra,167 palabra,168 palabra,169 palabra,17 03,17 05,17 06,17 10,17 12,17 16,17 17,17 21,17 24,17 27,17
34,17 35,17 39,17 52,17 53,17 57,17 lema,17 palabra,17 tokens,170 palabra,172 palabra,173 palabra,174 palabra,175
```

Vectores TF-IDF guardados exitosamente.

Vocabulario (Características):

['00 06' '00 11' '00 18' ... 'época lema' 'últimos lema' 'única lema']

Matriz TF-IDF

(0, 1286) 1.0
(1, 2630) 0.542091277452448
(1, 4222) 0.5593028900547976
(1, 3849) 0.5593028900547976
(1, 1728) 0.2837209919316424
(2, 2395) 0.5626387061885085
(2, 2827) 0.5626387061885085
(2, 3579) 0.5626387061885085
(2, 1728) 0.22430572639940216
(3, 2619) 0.5650826384030044
(3, 4259) 0.5650826384030044
(3, 3886) 0.5650826384030044
(3, 1728) 0.20504837313789662
(4, 2630) 0.542091277452448
(4, 4222) 0.5593028900547976
(4, 3849) 0.5593028900547976

VECTORES DE VALORES TF-IDF DE TRIGRAMAS

```
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer

# Cargar los tweets preprocesados
ruta_datos = ('C:\\Users\\lemanu\\Documents\\Escuela\\PROGRAMACION\\python\\Machine\\Construcción de vectores TF-IDF\\TWEETS_EMOCIONES_NEGATIVAS.txt')
with open(ruta_datos, 'r', encoding='utf-8') as file:
    tweets = file.readlines()

# Crear un objeto TfidfVectorizer para bigramas
#tfidf_vectorizer = TfidfVectorizer(ngram_range=(2, 2))
# Crear un objeto TfidfVectorizer para trigramas
tfidf_vectorizer = TfidfVectorizer(ngram_range=(3, 3))

# Ajustar el vectorizador y transformar los datos de texto
tfidf_matrix = tfidf_vectorizer.fit_transform(tweets)

# Obtener el vocabulario (características)
feature_names = tfidf_vectorizer.get_feature_names_out()
# Convertir la matriz TF-IDF a DataFrame para mejor manejo
df_tfidf = pd.DataFrame(tfidf_matrix.toarray(), columns=tfidf_vectorizer.get_feature_names_out())

# Guardar los vectores TF-IDF en un archivo
df_tfidf.to_csv( path_or_buf: "vectores_tweets_Tri_emonegativas.txt", index=False)

print("Vectores TF-IDF guardados exitosamente.")

#Imprimir Vocabulario
print("Vocabulario (Características): ")
print(feature_names)

#Matriz
print("\nMatriz TF-IDF")
print(tfidf_matrix)
```

Construccion.py x Codigo2.py x vectores_tweets_Tri_emonegativas.txt x construccion_2.py x tweets

00 06 28,00 11 18,00 18 15,00 20 lema,00 25 23,00 32 22,00 35 31,00 35 32,00 36 38,00 44 lema,00 46 38,00 54 27,00
55 lema,00 lema 13,00 lema 21,01 10 lema,01 24 12,01 29 lema,01 36 38,01 42 31,01 44 24,01 44 lema,01 45 54,01
50 47,01 lema 02,01 lema 09,01 lema 11,01 lema 2014,01 lema 21,02 09 lema,02 16 59,02 18 19,02 21 36,02 22
lema,02 23 09,02 28 lema,02 38 51,02 38 59,02 39 01,02 39 42,02 50 lema,03 00 44,03 01 44,03 03 07,03 07 lema,03
13 35,03 20 24,03 29 lema,03 33 lema,03 34 55,03 42 lema,03 46 lema,03 49 56,03 53 21,03 57 30,03 lema 15,03 lema
17,03 lema 22,04 01 29,04 03 lema,04 27 46,04 29 37,04 31 18,04 35 lema,04 53 36,05 06 lema,05 07 lema,05 08
lema,05 09 lema,05 10 lema,05 11 46,05 11 lema,05 13 lema,05 14 lema,05 15 lema,05 16 lema,05 17 lema,05 18
lema,05 19 lema,05 20 lema,05 21 lema,05 22 lema,05 23 lema,05 25 lema,05 27 lema,05 29 lema,05 30 lema,05 31
lema,05 53 lema,05 lema 12,05 lema 15,05 lema 16,05 lema 18,05 lema 19,06 00 55,06 01 lema,06 10 32,06 22 lema,06
23 lema,06 25 lema,06 26 lema,06 28 lema,06 39 lema,06 46 lema,06 lema 20,06 lema 2014,07 05 53,07 12 lema,07 17
lema,07 52 lema,07 lema 03,07 lema 16,07 lema 2014,08 18 lema,08 36 34,08 lema 17,08 lema 2014,09 17 lema,09 24
24,09 25 01,09 lema 02,09 lema 2014,09 lema 21,09 lema 22,10 20 lema,10 30 lema,10 32 lema,10 34 28,10 lema 10,10
lema 16,10 lema 2014,10 lema 21,10 palabra 2014,10 palabra adrenalina,10 palabra afecta,10 palabra algo,10
palabra altura,10 palabra bandaas,10 palabra cocaCola,10 palabra como,10 palabra cuenta,10 palabra cuestan,10
palabra esperan,10 palabra ganada,10 palabra graduar,10 palabra hacer,10 palabra hemorroides,10 palabra humano,10
palabra inglés,10 palabra lema,10 palabra mejor,10 palabra mi,10 palabra pero,10 palabra pie,10 palabra que,10
palabra quisiera,10 palabra seguía,10 palabra ser,10 palabra si,10 palabra sólo,10 palabra tarde,10 palabra te,10
palabra tenemos,10 palabra tener,10 palabra tuve,10 palabra ver,10 palabra vuelta,100 palabra 15,100 palabra
como,100 palabra muy,100 palabra no,100 palabra toda,101 palabra estos,101 palabra extraño,101 palabra le,101
palabra piba,101 palabra razón,101 palabra tengo,102 palabra dije,102 palabra le,102 palabra que,103 palabra
bélgica,103 palabra hace,103 palabra no,103 palabra sale,104 palabra 2014,104 palabra caro,104 palabra caso,104

Vectores TF-IDF guardados exitosamente.

Vocabulario (Características):

['00 06 28' '00 11 18' '00 18 15' ... 'época lema época'
'últimos lema último' 'única lema único']

Matriz TF-IDF

(1, 5488) 0.5308345328786249
(1, 5109) 0.5308345328786249
(1, 3587) 0.6606280325622461
(2, 3967) 0.5584003975861113
(2, 4839) 0.5584003975861113
(2, 3523) 0.6134965296978833
(3, 5525) 0.5672461881864367
(3, 5146) 0.5672461881864367
(3, 3598) 0.5970456632251133
(4, 5488) 0.5308345328786249
(4, 5109) 0.5308345328786249
(4, 3587) 0.6606280325622461
(5, 3967) 0.5584003975861113
(5, 4839) 0.5584003975861113
(5, 3523) 0.6134965296978833
(6, 251) 0.5906662844490117
(6, 3017) 0.3595923235222003
(6, 3967) 0.510786991454904
(6, 4839) 0.510786991454904
(7, 5513) 0.4975865636039678
(7, 5134) 0.4975865636039678
(7, 409) 0.5999708151441974

VECTORES DE VALORES TF-IDF DE CUATRIGRAMAS

```
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer

# Cargar los tweets preprocesados
ruta_datos = (r'C:\Users\emanu\Documents\Escuela\PROGRAMACION\python\Machine\Construcción de vectores TF-IDF\TWEETS_EMOCIONES_NEGATIVAS.txt')
with open(ruta_datos, 'r', encoding='utf-8') as file:
    tweets = file.readlines()

# Crear un objeto TfidfVectorizer para bigramas
#tfidf_vectorizer = TfidfVectorizer(ngram_range=(2, 2))
# Crear un objeto TfidfVectorizer para trigramas
#tfidf_vectorizer = TfidfVectorizer(ngram_range=(3, 3))
# Crear un objeto TfidfVectorizer para cuatrigramas
tfidf_vectorizer = TfidfVectorizer(ngram_range=(4, 4))

# Ajustar el vectorizador y transformar los datos de texto
tfidf_matrix = tfidf_vectorizer.fit_transform(tweets)

# Obtener el vocabulario (características)
feature_names = tfidf_vectorizer.get_feature_names_out()

# Convertir la matriz TF-IDF a DataFrame para mejor manejo
df_tfidf = pd.DataFrame(tfidf_matrix.toarray(), columns=tfidf_vectorizer.get_feature_names_out())

# Guardar los vectores TF-IDF en un archivo
df_tfidf.to_csv( path_or_buf= "vectores_tweets_Cuatri_emonegativas.txt", index=False)

print("Vectores TF-IDF guardados exitosamente.")

#Imprimir Vocabulario
print("Vocabulario (Características): ")
print(feature_names)

#Matriz
print("\nMatriz TF-IDF")
print(tfidf_matrix)
```

```
tores TF-IDF C:\
00 06 28 lema,00 11 18 lema,00 18 15 lema,00 20 lema 16,00 25 23 lema,00 32 22 lema,00 35 31 lema,00 35 32 lema,00
36 38 lema,00 44 lema 03,00 46 38 lema,00 54 27 lema,00 55 lema 06,00 lema 13 25,00 lema 21 35,01 10 lema 16,01
24 12 lema,01 29 lema 04,01 36 38 lema,01 42 31 lema,01 44 24 lema,01 44 lema 03,01 45 54 lema,01 50 47 lema,01
lema 02 39,01 lema 09 25,01 lema 11 21,01 lema 2014 06,01 lema 21 44,02 09 lema 21,02 16 59 lema,02 18 19 lema,02
21 36 lema,02 22 lema 21,02 23 09 lema,02 28 lema 21,02 38 51 lema,02 38 59 lema,02 39 01 lema,02 39 42 lema,02
50 lema 15,03 00 44 lema,03 01 44 lema,03 03 07 lema,03 07 lema 03,03 13 35 lema,03 20 24 lema,03 29 lema 17,03
33 lema 17,03 34 55 lema,03 42 lema 19,03 46 lema 22,03 49 56 lema,03 53 21 lema,03 57 30 lema,03 lema 15 04,03
lema 15 28,03 lema 17 12,03 lema 22 33,04 01 29 lema,04 03 lema 15,04 27 46 lema,04 29 37 lema,04 31 18 lema,04
35 lema 18,04 53 36 lema,05 06 lema 2014,05 07 lema 2014,05 08 lema 2014,05 09 lema 2014,05 10 lema 2014,05 11 46
lema,05 11 lema 2014,05 13 lema 2014,05 14 lema 2014,05 15 lema 2014,05 16 lema 2014,05 17 lema 2014,05 18 lema
17,05 18 lema 2014,05 19 lema 2014,05 20 lema 2014,05 21 lema 2014,05 22 lema 2014,05 23 lema 2014,05 25 lema
2014,05 27 lema 2014,05 29 lema 2014,05 30 lema 2014,05 31 lema 2014,05 53 lema 07,05 lema 12 51,05 lema 15 39,05
lema 16 16,05 lema 18 12,05 lema 19 16,06 00 55 lema,06 01 lema 2014,06 10 32 lema,06 22 lema 2014,06 23 lema
2014,06 25 lema 2014,06 26 lema 2014,06 28 lema 00,06 39 lema 23,06 46 lema 17,06 lema 20 18,06 lema 2014 05,07
05 53 lema,07 12 lema 21,07 17 lema 18,07 52 lema 23,07 lema 03 03,07 lema 16 45,07 lema 2014 05,08 18 lema 16,08
36 34 lema,08 lema 17 35,08 lema 2014 05,09 17 lema 22,09 24 24 lema,09 25 01 lema,09 lema 02 23,09 lema 2014
05,09 lema 21 02,09 lema 22 28,10 20 lema 23,10 30 lema 17,10 32 lema 06,10 34 28 lema,10 lema 16 01,10 lema 2014
05,10 lema 21 56,10 palabra 2014 05,10 palabra adrenalina lema,10 palabra afecta lema,10 palabra algo lema,10
palabra altura lema,10 palabra bandaas lema,10 palabra cocacola lema,10 palabra como lema,10 palabra cuenta
lema,10 palabra cuestan lema,10 palabra esperan lema,10 palabra ganada lema,10 palabra graduar lema,10 palabra
hacer lema,10 palabra hemorroides lema,10 palabra humano lema,10 palabra inglés lema,10 palabra mejor lema,10
palabra mi lema,10 palabra pero lema,10 palabra pie lema,10 palabra que lema,10 palabra quisiera lema,10 palabra
seguía lema,10 palabra ser lema,10 palabra si lema,10 palabra sólo lema,10 palabra tarde lema,10 palabra te
lema,10 palabra tenemos lema,10 palabra tener lema,10 palabra tuve lema,10 palabra ver lema,10 palabra vuelta
lema,100 palabra 15 23,100 palabra como lema,100 palabra muy lema,100 palabra no lema,100 palabra toda lema,101
palabra estos lema,101 palabra extraño lema,101 palabra le lema,101 palabra piba lema,101 palabra razón lema,101
palabra tengo lema,102 palabra dije lema,102 palabra le lema,102 palabra que lema,103 palabra bética lema,103
```

Vectores TF-IDF guardados exitosamente.

Vocabulario (Características):

```
['00 06 28 lema' '00 11 18 lema' '00 18 15 lema' ... 'una lema es una'  
'unk armar lema unk' 'unk itear lema unk']
```

Matriz TF-IDF

```
(1, 5954) 0.6263715492594969  
(1, 4563) 0.7795246514884938  
(2, 5679) 0.6731190113726502  
(2, 4499) 0.7395341753622384  
(3, 5991) 0.6887831656014666  
(3, 4574) 0.7249674136014823  
(4, 5954) 0.6263715492594969  
(4, 4563) 0.7795246514884938  
(5, 5679) 0.6731190113726502  
(5, 4499) 0.7395341753622384  
(6, 251) 0.6032621846340617  
(6, 2534) 0.6032621846340617  
(6, 5679) 0.5216794736052769  
(7, 5979) 0.5058690884515811  
(7, 408) 0.609957566290134  
(7, 2675) 0.609957566290134  
(8, 5468) 0.5058690884515811  
(8, 610) 0.609957566290134  
(8, 2833) 0.609957566290134  
(9, 5654) 0.5314298874699189  
(9, 760) 0.5989917673489801  
(9, 2941) 0.5989917673489801  
(10, 6043) 0.5773502691896257  
(10, 809) 0.5773502691896257
```

VECTORES DE VALORES TF-IDF DE UNITRIGRAMAS

[illegible]

Vectores TF-IDF guardados exitosamente.

Vocabulario (Características):

['00' '01' '02' ... 'últimos' 'única' 'único']

Matriz TF-IDF

```
(0, 1605) 0.7071067811865476
(0, 902) 0.7071067811865476
(1, 1634) 0.677961005644783
(1, 1065) 0.13053086611390755
(1, 1574) 0.6994865358903438
(1, 1264) 0.1304782624133122
(1, 1004) 0.1304782624133122
(2, 1248) 0.9920350081246972
(2, 1065) 0.07274400571505724
(2, 1264) 0.07271469001363991
(2, 1004) 0.07271469001363991
(3, 1621) 0.993387869484056
(3, 1065) 0.06630139629677019
(3, 1264) 0.06627467695517934
(3, 1004) 0.06627467695517934
(4, 1634) 0.677961005644783
(4, 1065) 0.13053086611390755
(4, 1574) 0.6994865358903438
(4, 1264) 0.1304782624133122
(4, 1004) 0.1304782624133122
(5, 1248) 0.9920350081246972
(5, 1065) 0.07274400571505724
(5, 1264) 0.07271469001363991
```