

# Procesamiento del Lenguaje Natural

Rodrigo S. Cortez Madrigal



```
In [1]: import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer

import plotly
from plotly import graph_objs as go
from plotly import express as px
from plotly.subplots import make_subplots

# Plotly configuration
plotly.offline.init_notebook_mode(connected=True)
```

Obtener la matriz de ocurrencias, y por extracción booleana, obtener los documentos que tengan la palabra guerra, bombas y casa.

```
In [2]: docs = ["No sé con qué armas se peleará la tercera guerra mundial, pero la cuarta se peleará con palos y piedras",
"El fin de la segunda guerra mundial llegó con las bombas atómicas lanzadas en Japón.",
"La casa se está incendiando porque le cayeron bombas."]

words = ["guerra", "bombas", "casa"]
```

```
In [3]: # Obtener la matriz de ocurrencias, y por extracción booleana, obtener los documentos que tengan la palabra guerra, bombas y casa.

def get_occurrences(docs, words):
    """
    Get the occurrences of words in documents
    """
    # Create a CountVectorizer object
    vectorizer = CountVectorizer(vocabulary=words)

    # Fit and transform the documents
    X = vectorizer.fit_transform(docs)

    # Convert to array and create dataframe with words as columns
    occurrences = pd.DataFrame(X.toarray(), columns=vectorizer.get_feature_names_out())

    return occurrences

def get_documents_with_words(docs, words):
    """
    Get the documents that contain the words
    """
    # Create a CountVectorizer object
    vectorizer = CountVectorizer(vocabulary=words)

    # Fit and transform the documents
    X = vectorizer.fit_transform(docs)

    # Get the boolean mask of documents that contain the words
    mask = X.toarray().astype(bool)

    # Get the documents that contain the words
    documents_with_words = [doc for doc, m in zip(docs, mask) if all(m)]

    return documents_with_words

def get_word_frequencies(docs):
    """
    Get the word frequencies in the documents
    """
    # Create a CountVectorizer object
    vectorizer = CountVectorizer()

    # Fit and transform the documents
    X = vectorizer.fit_transform(docs)

    # Get the word frequencies
    word_frequencies = X.toarray().sum(axis=0)

    # Get the words
    words = vectorizer.get_feature_names_out()

    # Create a DataFrame with the word frequencies
    df = pd.DataFrame(word_frequencies, index=words, columns=["Frequency"])

    return df

print("-- * 50)
print(f"Occurrences matrix: {get_occurrences(docs, words)}")
print("-- * 50)
print(f"Documents with words: {get_documents_with_words(docs, words)}")
print("-- * 50)

# Plot occurrence matrix
fig = make_subplots(rows=1, cols=1)
fig.add_trace(go.Heatmap(z=get_occurrences(docs, words).values, x=get_occurrences(docs, words).columns, y=docs, row=1, col=1))
fig.update_layout(title_text="Occurrences Matrix")
fig.show()

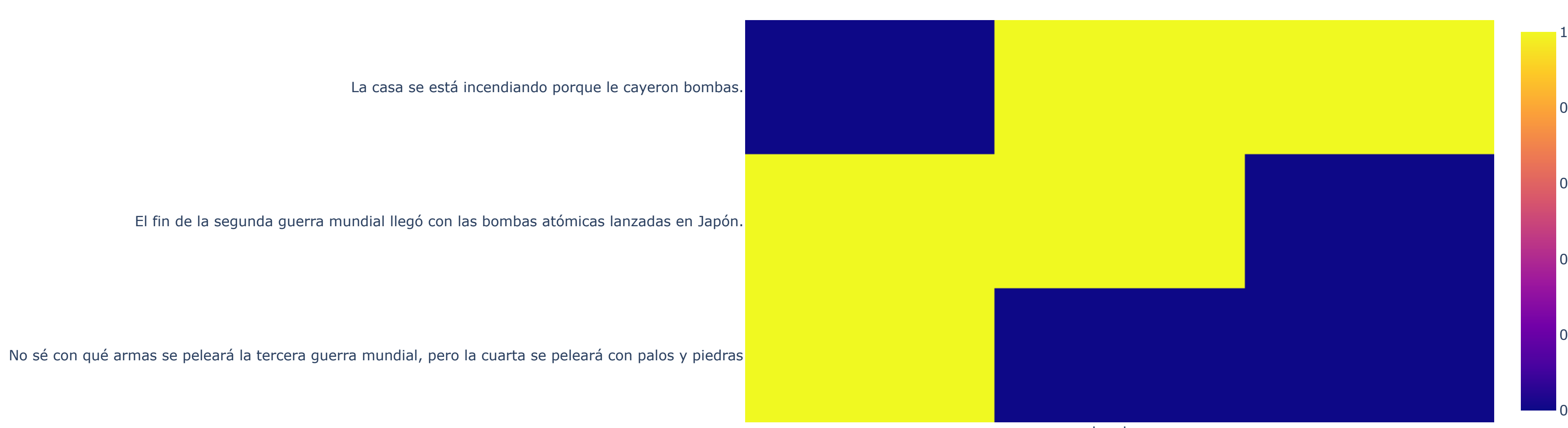
for doc in docs:
    print(f"Document: {doc}")
    print(f"Word frequencies: {get_word_frequencies([doc])}")
    print("-- * 50)
    # Plot
    fig = make_subplots(rows=1, cols=1)
    fig.add_trace(go.Bar(x=get_word_frequencies([doc]).index, y=get_word_frequencies([doc])["Frequency"], row=1, col=1))
    fig.update_layout(title_text=f"Word Frequencies in Document: {doc}")
    fig.show()
```

Occurrences matrix:

	guerra	bombas	casa
0	1	0	0
1	1	1	0
2	0	1	1

Documents with words: []

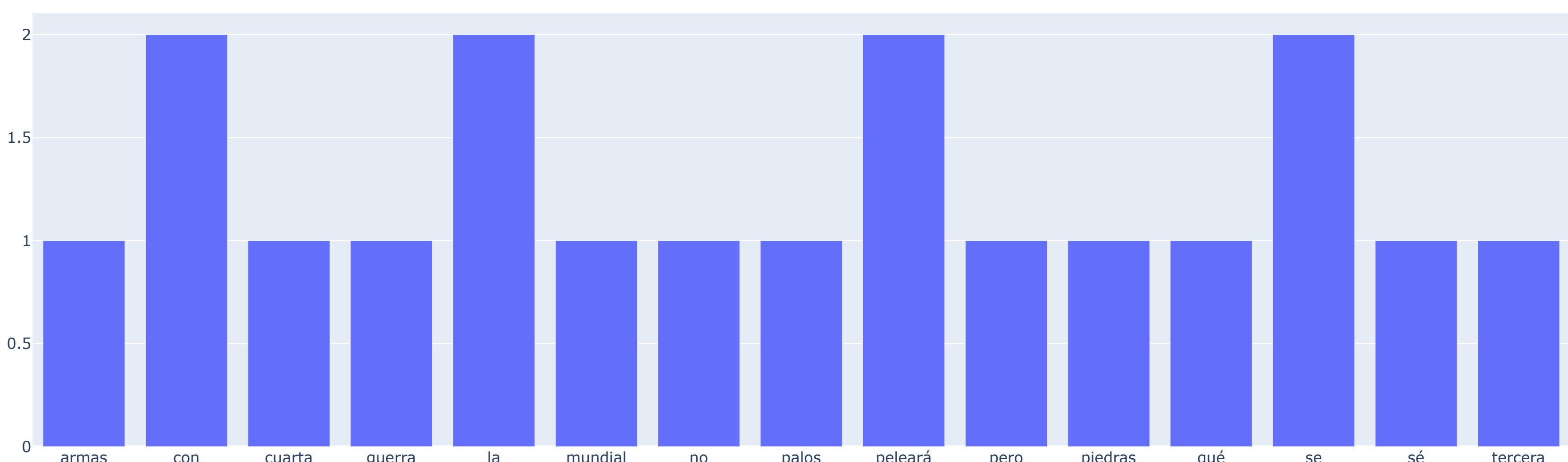
Occurrences Matrix



Document: No sé con qué armas se peleará la tercera guerra mundial, pero la cuarta se peleará con palos y piedras

Word frequencies:	Frequency
armas	1
con	2
cuarta	1
guerra	1
la	2
mundial	1
no	1
palos	1
peleará	2
pero	1
piedras	1
qué	1
se	2
sé	1
tercera	1

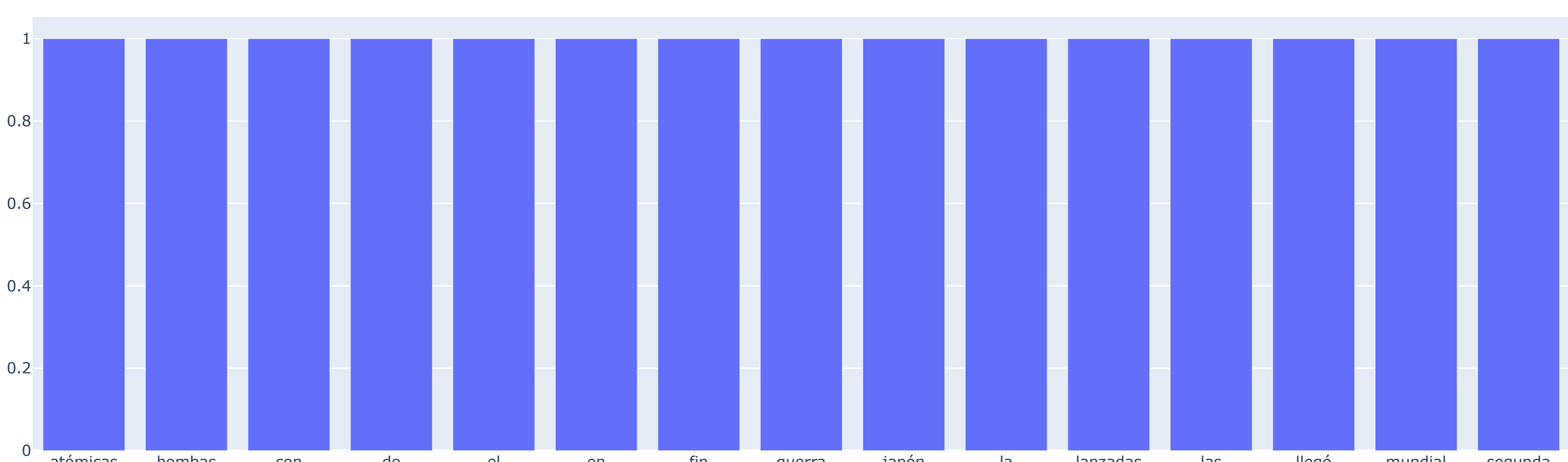
Word Frequencies in Document: No sé con qué armas se peleará la tercera guerra mundial, pero la cuarta se peleará con palos y piedras



Document: El fin de la segunda guerra mundial llegó con las bombas atómicas lanzadas en Japón.

Word frequencies:	Frequency
atómicas	1
bombas	1
con	1
de	1
el	1
en	1
fin	1
guerra	1
japón	1
la	1
lanzadas	1
las	1
llegó	1
mundial	1
segunda	1

Word Frequencies in Document: El fin de la segunda guerra mundial llegó con las bombas atómicas lanzadas en Japón.



Document: La casa se está incendiando porque le cayeron bombas.

Word frequencies:	Frequency
bombas	1
casa	1
cayeron	1
está	1
incendiando	1
la	1
le	1
porque	1
se	1

Word Frequencies in Document: La casa se está incendiando porque le cayeron bombas.

