

```
In [61]: import nltk
import numpy as np
import pandas as pd
from nltk.corpus import stopwords
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import pairwise_distances
from sklearn.metrics.pairwise import cosine_similarity
```

```
In [62]: nltk.download('stopwords')
doc = [
    "I es un campo en rápido avance que implica el desarrollo de máquinas inteligen
    "El aprendizaje automático es un subconjunto de la IA que se centra en la capac
    "El aprendizaje profundo es un subcampo del aprendizaje automático que utiliza
    "La IA se aplica en diversos sectores, como la sanidad, el transporte y el ocio
    "Las consideraciones éticas desempeñan un papel importante en el desarrollo de
    "La inteligencia artificial puede revolucionar muchos aspectos de la sociedad."
    "Los sistemas de IA, como los chatbots y los asistentes virtuales, son cada vez
    "El procesamiento del lenguaje natural es una rama de la IA que permite a las m
    "El objetivo de la visión por computadora es que las máquinas comprendan la inf
]
```

```
[nltk_data] Downloading package stopwords to
[nltk_data] C:\Users\chapu\AppData\Roaming\nltk_data...
[nltk_data] Package stopwords is already up-to-date!
```

```
In [63]: # preprocesamiento.
doc = [x.lower() for x in doc]
```

```
In [64]: # parámetros
max_features = 18
min_df = 1
max_df = 3
stop_words = stopwords.words('english')
```

```
In [65]: vectorizer = TfidfVectorizer(max_features=max_features,
                                     min_df=min_df,
                                     max_df=max_df,
                                     stop_words=stop_words)

X = vectorizer.fit_transform(doc).toarray()
```

```
In [66]: #? Calcular la similitud del coseno entre los documentos
cosine_similarities = cosine_similarity(X, X)
```

```
In [67]: df_similitud = pd.DataFrame(cosine_similarities, index=[f"Doc{i+1}" for i in range(
                                     columns=[f"Doc{i+1}" for i in range(len(doc))])
```

```
In [68]: import matplotlib.pyplot as plt
import seaborn as sns

plt.figure(figsize=(10, 8))
sns.heatmap(cosine_similarities, annot=True, cmap="Reds", xticklabels=[f"Linea {i+1}"
                                                                    for i in range(len(doc))])
```

```
plt.title("Mapa de calor de similitudes")  
plt.show()
```

