

Proyecto NLP: Pipeline Completo

Clasificación de Sentimientos con Análisis Exhaustivo

Gustavo Andrés González Pineda - 21438

Maria Marta Ramírez Gil - 21342

```
positivos_templates = [
    "excelente producto calidad precio recomiendo totalmente",
    "muy satisfecho compra superó expectativas",
    "increíble servicio rápido bien empaquetado perfecto",
    "fantástico mejor compra año definitivamente volveré",
    "maravilloso alta calidad vale pena inversión",
    "genial cumple funciones perfectamente satisfecho",
    "buenísimo llegó tiempo perfecto estado recomiendo",
    "espectacular justo buscaba muy contento",
    "magnífico producto premium calidad excepcional",
    "extraordinario supera descripción muy feliz compra",
    "perfecto exactamente esperaba cinco estrellas",
    "impresionante calidad construcción durabilidad excelente",
    "sobresaliente mejor opción mercado sin duda",
    "fabuloso diseño elegante funcional práctico",
    "brillante solución problema perfecto uso diario",
    "estupendo relación calidad precio insuperable",
    "formidable cumple promesas vendedor honesto",
    "sensacional mejor inversión hecho recomiendo ampliamente",
    "increíblemente bueno resistente duradero confiable",
    "tremendamente satisfecho volveré comprar definitivamente"
```

Carga y Creación del Corpus

Dataset expandido con 3005 documentos generados.

Distribución de Clases

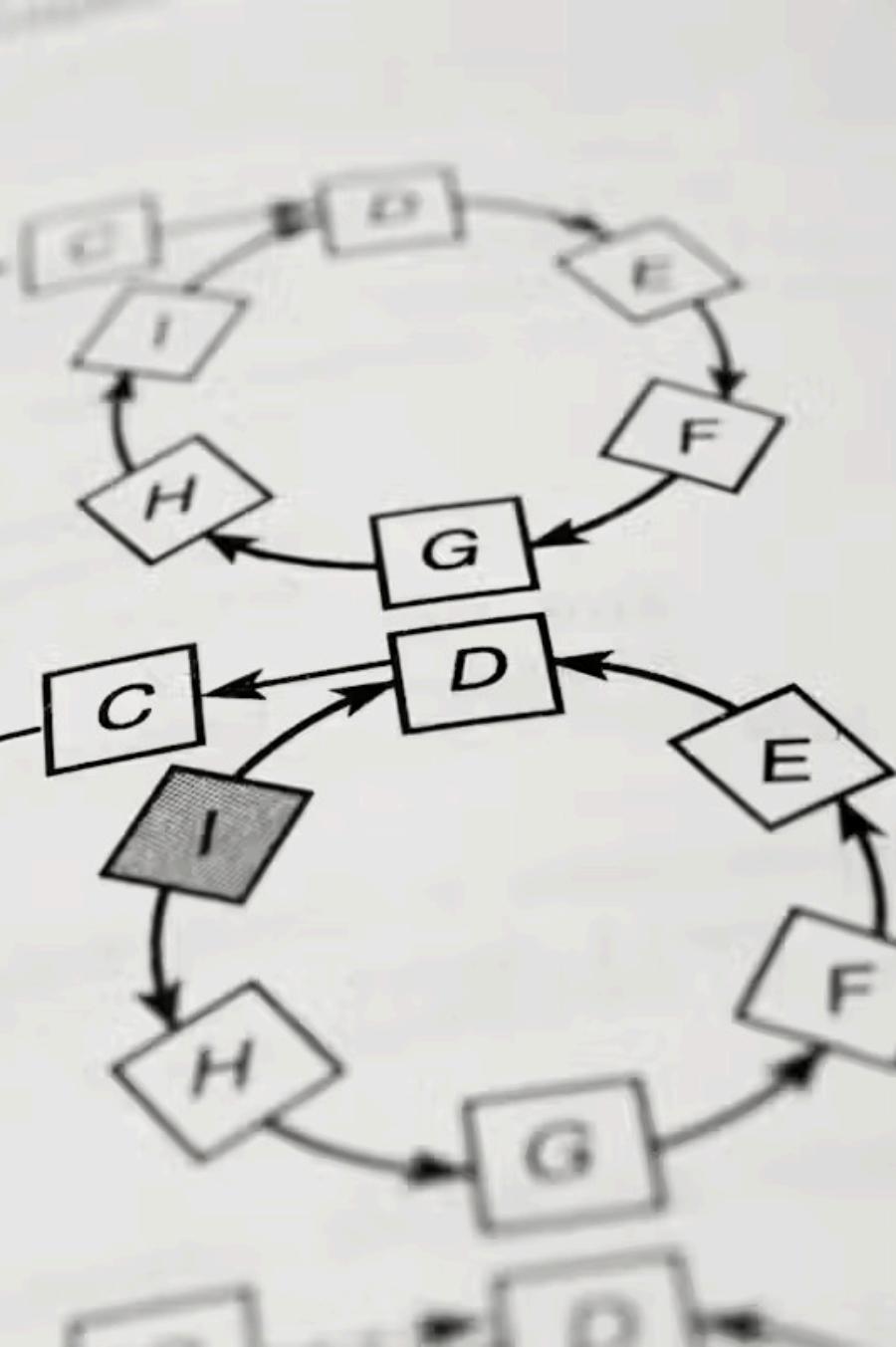
- Neutro: 1005 (33.4%)
- Positivo: 1000 (33.3%)
- Negativo: 1000 (33.3%)

Balance de clases equilibrado para análisis preciso.

Preprocesamiento del Texto

Estadísticas del Corpus

- Longitud promedio (tokens): 5.1
- Longitud máxima: 7
- Longitud mínima: 3



Algoritmo de Levenshtein

Análisis de similitudes y detección de errores ortográficos.

Vocabulario Total

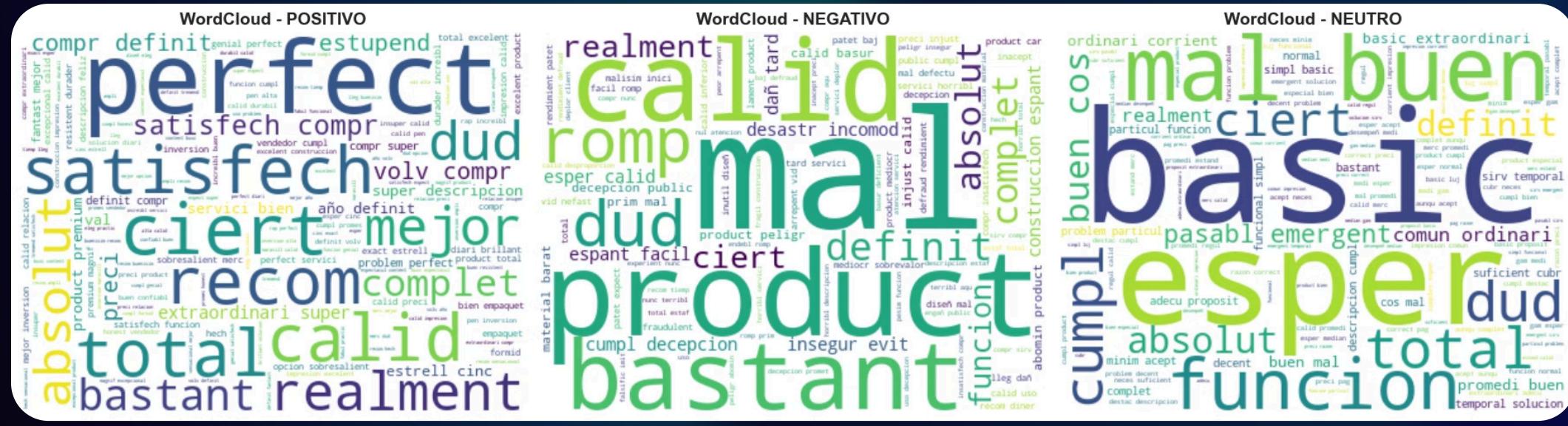
178 palabras únicas.

Ejemplo de Similitudes (posibles errores ortográficos)

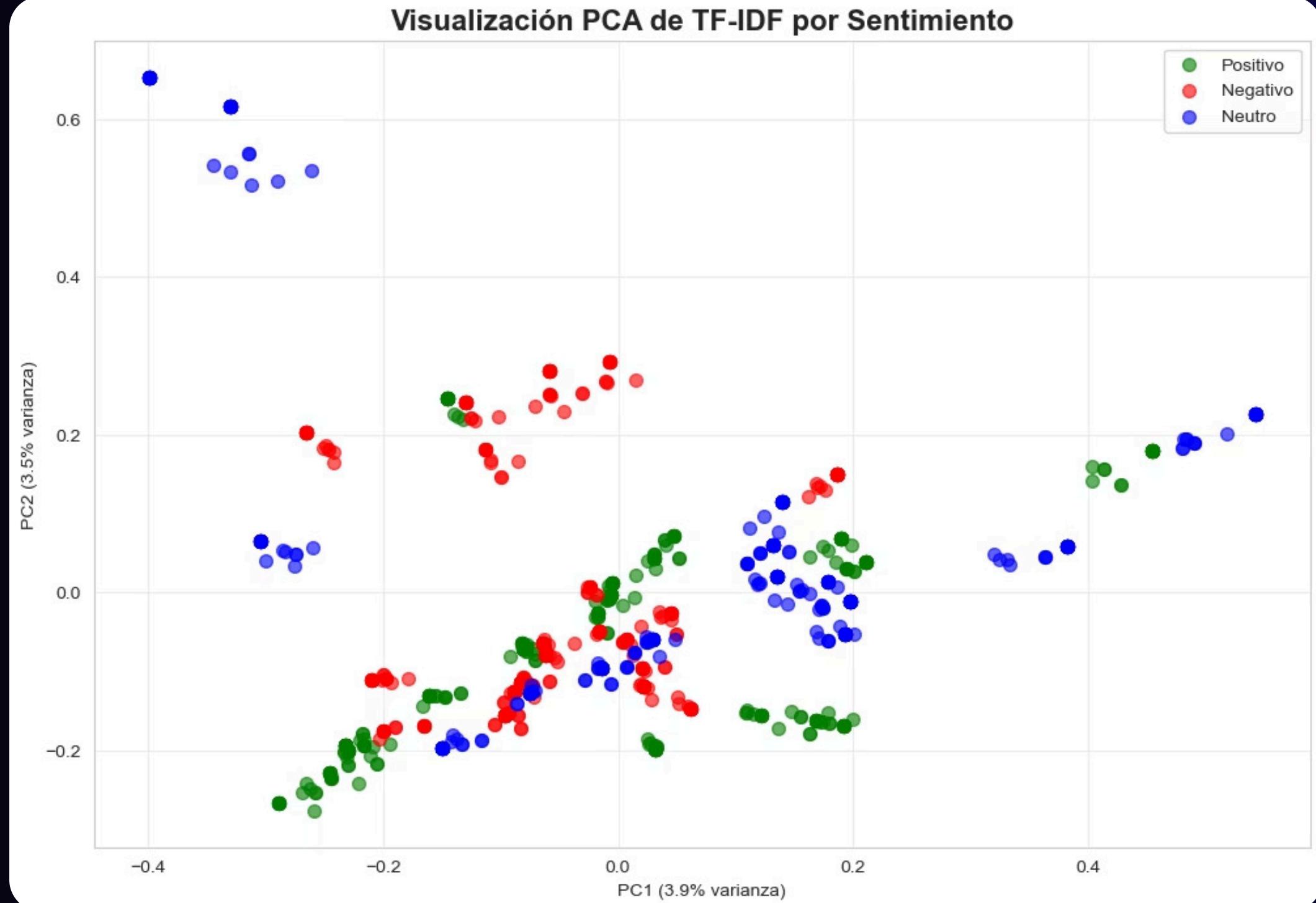
'excelente' → [('excellent', 1)]

Visualización de Representaciones

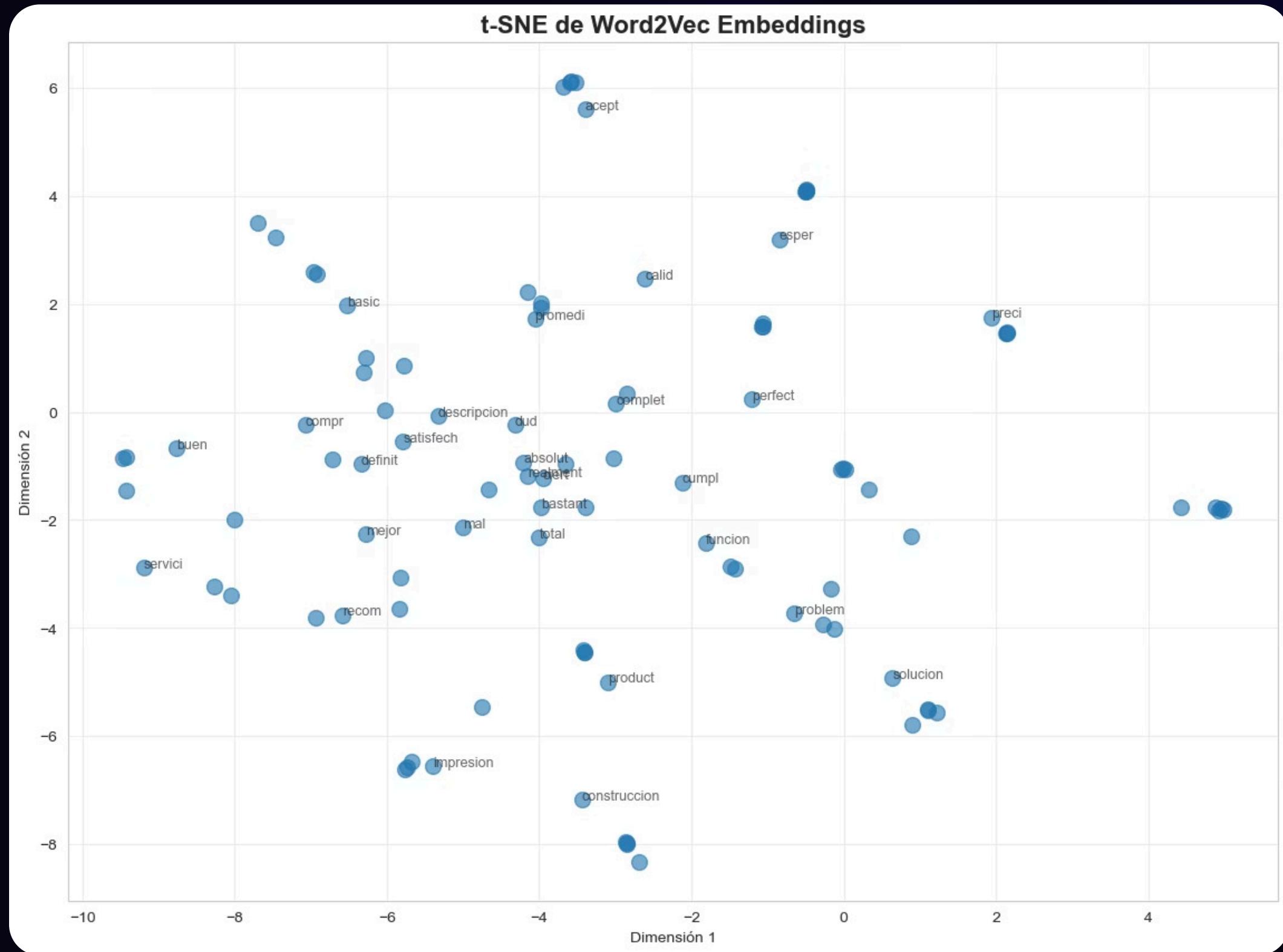
WordCloud



PCA de TF-IDF

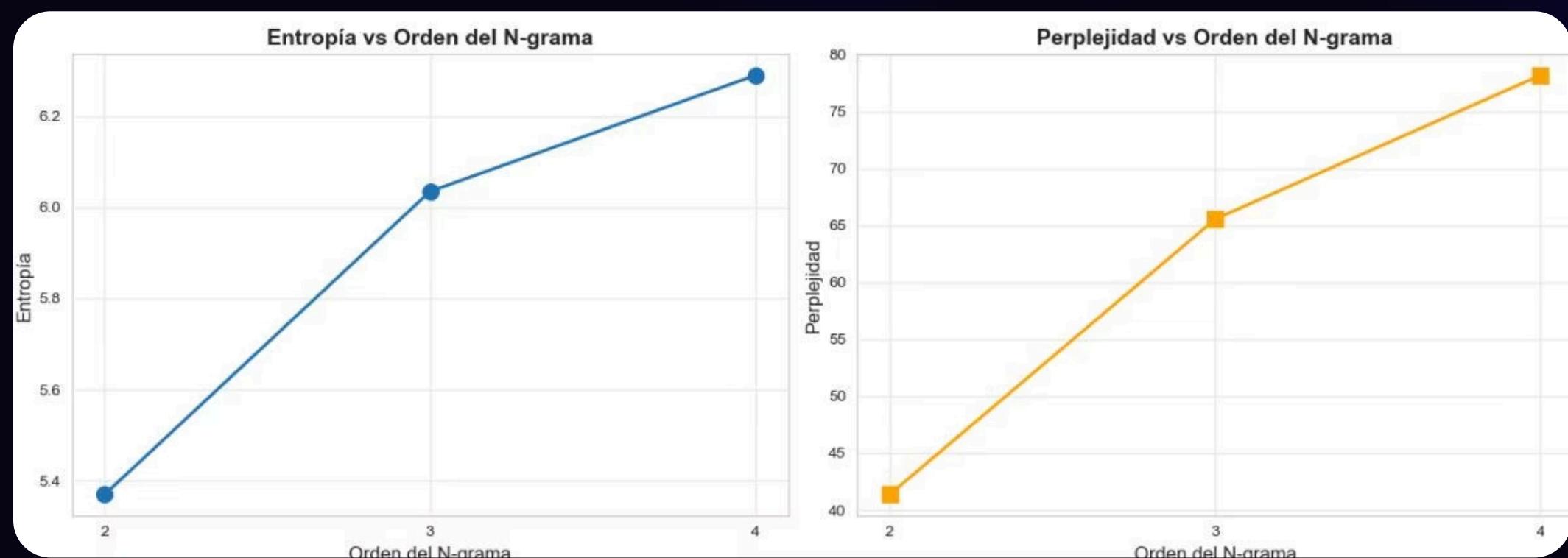
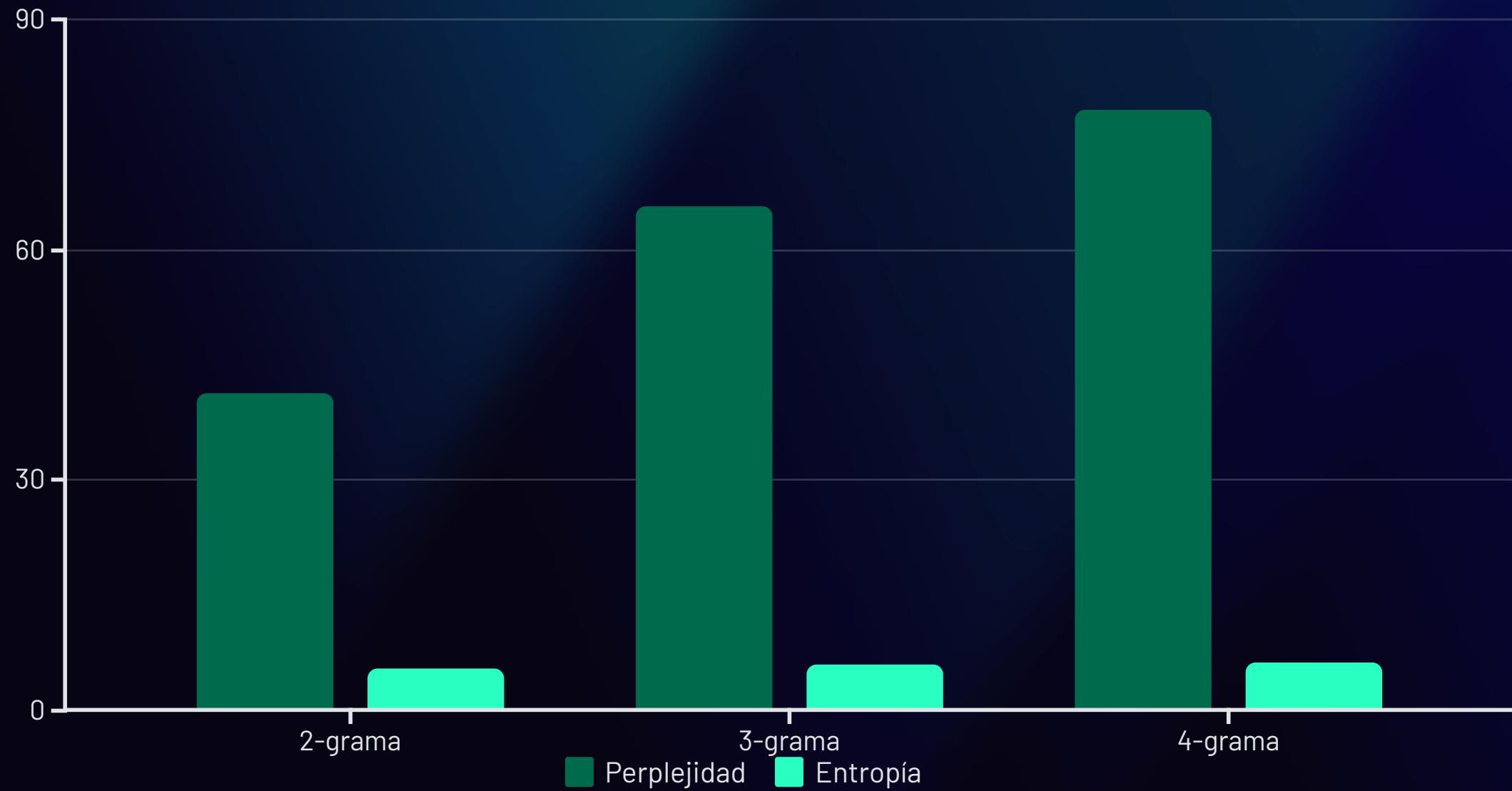


t-SNE de Word2Vec



7. Modelos Probabilísticos - N-Gramas

Fase 3: Entrenando modelos N-gramas.



8. Clasificación Supervisada

Fase 4: Entrenando modelos con 2404 datos de entrenamiento y 601 de prueba.

1.0

Naive Bayes

Accuracy: 1.0000, F1-Score: 1.0000

1.0

SVM

Accuracy: 1.0000, F1-Score: 1.0000

1.0

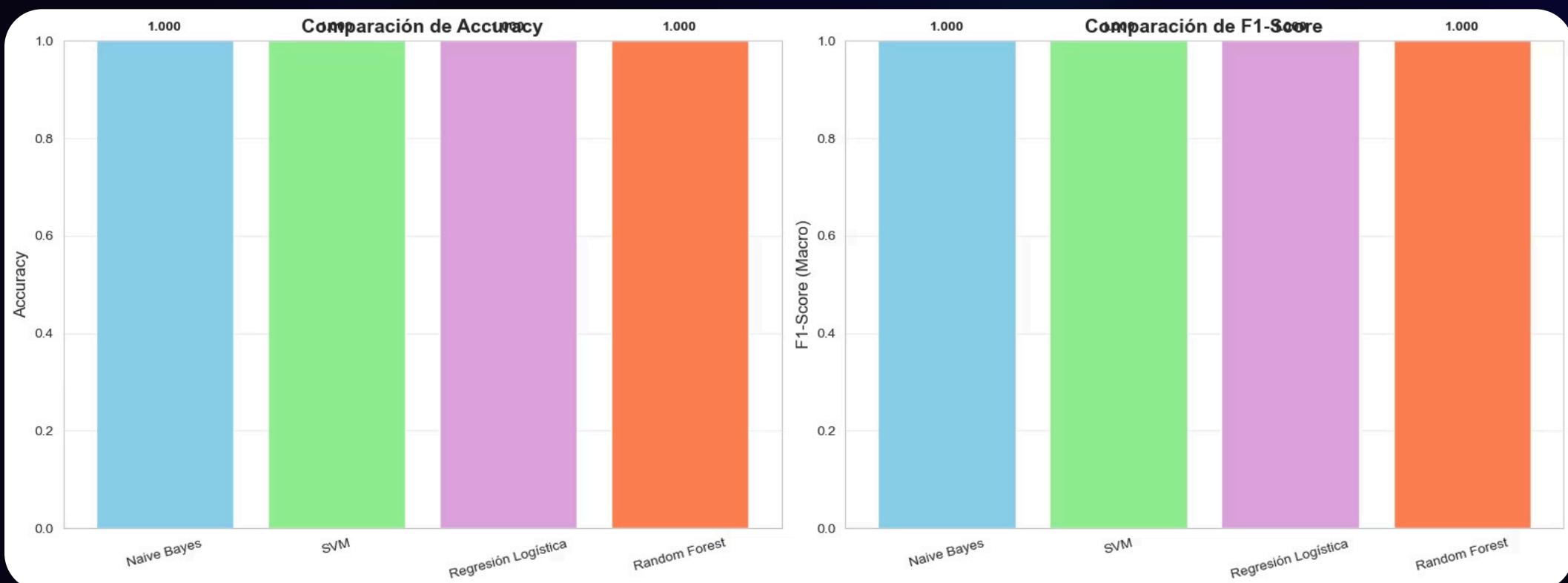
Regresión Logística

Accuracy: 1.0000, F1-Score: 1.0000

1.0

Random Forest

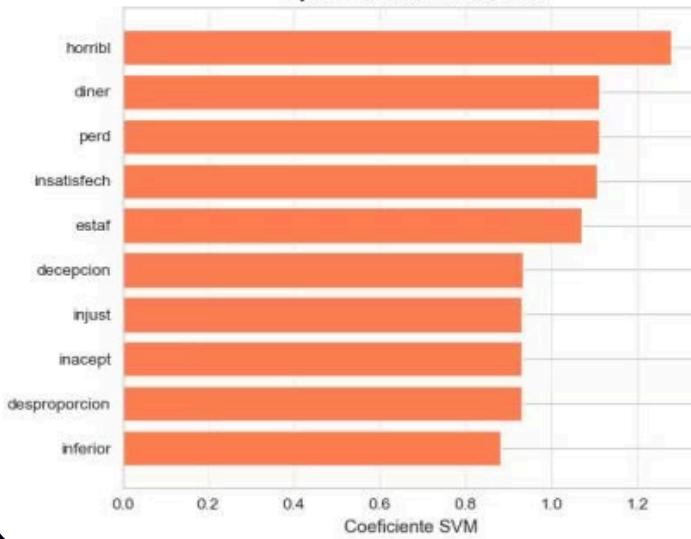
Accuracy: 1.0000, F1-Score: 1.0000



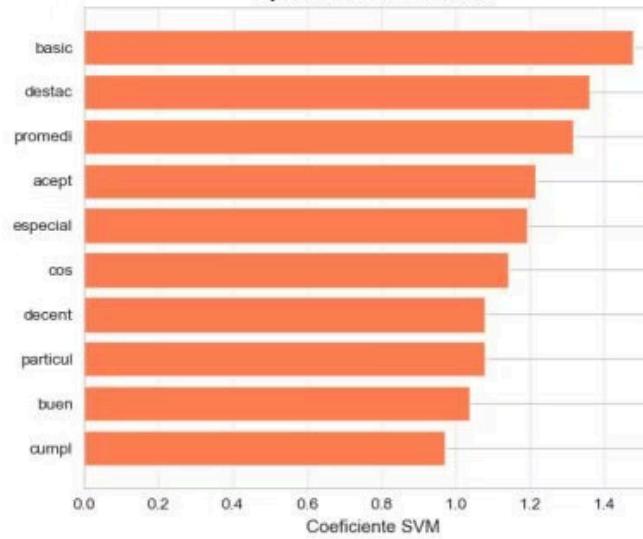
Análisis de Características Importantes

====

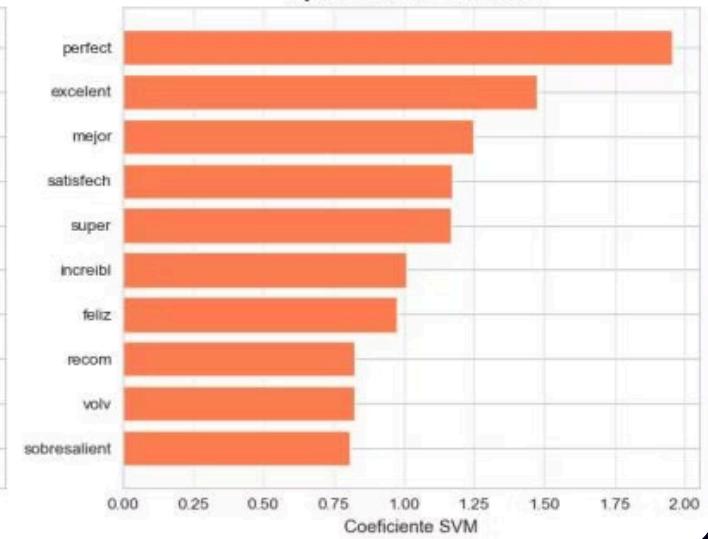
Top 10 Palabras - NEGATIVO



Top 10 Palabras - NEUTRO



Top 10 Palabras - POSITIVO



Sistema de Predicción

Ejemplo:

```
textos_prueba = [  
    "Este producto es absolutamente fantástico, lo recomiendo totalmente",  
    "Muy decepcionado con la compra, no funciona bien para nada",  
    "El producto está bien, nada especial pero cumple su función",  
    "Excelente calidad precio, superó mis expectativas completamente",  
    "Pésima experiencia, llegó roto y el servicio es horrible"  
]
```

```
resultado = predecir_sentimiento(texto)
```

```
=====
```

Texto 1: Este producto es absolutamente fantástico, lo recomiendo totalmente

```
=====
```

Tokens procesados: product absolut fantast recom total...

PREDICCIONES:

- Naive Bayes: positivo
- SVM: positivo
- Regresión Logística: positivo
- Random Forest: negativo

PROBABILIDADES:

positivo : 0.656	<div style="width: 65.6%;"></div>
negativo : 0.255	<div style="width: 25.5%;"></div>
neutro : 0.088	<div style="width: 8.8%;"></div>