



INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO

PRÁCTICA 4. IDENTIFICACIÓN DE PALABRAS, FRASES
Y DOCUMENTOS SIMILARES

R E P O R T E

MATERIA:

TECNOLOGÍAS DE LENGUAJE NATURAL

PRESENTA:

LUIS FERNANDO RODRÍGUEZ-DOMÍNGUEZ

PROFESOR:

ITURIEL ENRIQUE FLORES ESTRADA

INSTITUTO POLITÉCNICO NACIONAL



Índice general

1	Introducción	1
2	Desarrollo	3
2.1	Generación del Cuerpo de Documentos	3
2.2	Normalización de Documentos	4
2.2.1	Normalización para WordNet	4
2.2.2	Normalización para Embeddings	5
2.3	Similitud de Palabras con Synsets (WordNet)	5
2.4	Similitud de Documentos con Synsets (WordNet)	10
2.5	Similitud de Palabras con Embedding (GloVe)	11
2.6	Similitud de Documentos con Embedding (BERT)	12
3	Conclusiones Generales	14
3.1	Análisis de Resultados	14
3.1.1	Similitud de Palabras	14
3.1.2	Similitud de Documentos (Frases Representativas)	15
3.1.3	Impacto de la Extracción de Frases y Desafíos	16
3.2	Conclusiones Finales	16

Capítulo 1

Introducción

El presente reporte documenta la realización de la Práctica 4 correspondiente a la asignatura de Tecnologías de Lenguaje Natural. El objetivo principal de esta práctica es desarrollar un sistema capaz de identificar y cuantificar la similitud entre palabras, frases y documentos. Para ello, se aplicarán diversas técnicas de Procesamiento de Lenguaje Natural (NLP), abarcando tanto enfoques semánticos basados en recursos léxicos estructurados como WordNet [1], así como enfoques basados en representaciones vectoriales densas (embeddings) como GloVe [2] y BERT [3].

Para llevar a cabo este estudio, se utilizará como corpus de documentos las introducciones de cinco libros con temáticas similares, obtenidos del Proyecto Gutenberg [4]. Cada introducción será tratada como un documento individual sobre el cual se aplicarán las distintas metodologías de cálculo de similitud.

Antes de aplicar las técnicas de similitud, se realizará un proceso de normalización de texto sobre cada documento. Este paso es fundamental para reducir el ruido, estandarizar el texto y mejorar la calidad de la entrada a los algoritmos de similitud. Este proceso incluirá tareas como la segmentación, tokenización, etiquetado gramatical (POS tagging), lematización y eliminación de palabras vacías (stopwords), utilizando principalmente las herramientas proporcionadas por la biblioteca NLTK [5].

Se implementarán y evaluarán los siguientes enfoques y técnicas:

- **Similitud de Palabras con WordNet:** Se identificarán los verbos y sustantivos más frecuentes en cada documento y se encontrarán términos similares utilizando las métricas `path_similarity` y `wup_similarity` de WordNet.
- **Similitud de Palabras con GloVe:** Para los verbos más frecuentes, se calculará la similitud con otros términos del vocabulario de GloVe utilizando la similitud coseno sobre sus vectores de embedding.
- **Extracción de Frases Representativas:** Se utilizará el algoritmo TextRank [6], implementado a través de la biblioteca Sumy [7], para extraer la frase más representativa de cada introducción.
- **Similitud de Frases con WordNet:** Se compararán las frases representativas entre sí utilizando un método de agregación de similitudes `path_similarity` a nivel de synsets.

- **Similitud de Frases con BERT:** Se obtendrán embeddings para las frases representativas utilizando el modelo `bert-base-uncased` de la biblioteca Transformers [8], y se calculará la similitud coseno entre ellas.

Además de la obtención de las puntuaciones de similitud, un aspecto clave de la práctica será el análisis comparativo de los resultados obtenidos mediante los diferentes enfoques, discutiendo sus fortalezas, debilidades y la naturaleza de las similitudes que cada uno es capaz de capturar.

Capítulo 2

Desarrollo

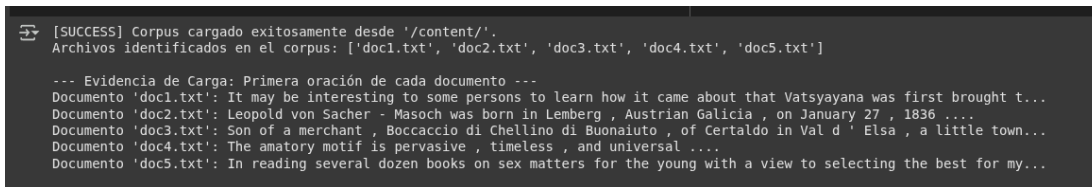
En este capítulo se detalla la implementación y ejecución de los diferentes módulos del programa desarrollado para la identificación de similitud entre palabras, frases y documentos. Se presentarán las metodologías empleadas y las evidencias de su funcionamiento, siguiendo los puntos establecidos en la rúbrica de la práctica.

2.1. Generación del Cuerpo de Documentos

El primer paso consistió en la creación de un corpus de documentos. Para esta práctica, se seleccionaron las introducciones de cinco libros del portal Project Gutenberg [4]. Se procuró que los libros tuvieran temáticas relacionadas para que el análisis de similitud fuera más significativo. Los libros seleccionados y sus respectivos identificadores en el corpus son:

- `doc1.txt`: Introducción de "The Kama Sutra of Vatsyayana" traducido por Richard Burton.
- `doc2.txt`: Introducción de "Venus in Furs" por Leopold von Sacher-Masoch.
- `doc3.txt`: Introducción de "The Decameron of Giovanni Boccaccio" por Giovanni Boccaccio.
- `doc4.txt`: Introducción de "The Love-potion" por Charles Nordmann.
- `doc5.txt`: Introducción de "Sex—The Unknown" por Havelock Ellis.

Cada introducción se guardó en un archivo de texto plano (`doc1.txt` a `doc5.txt`) y se colocó en el directorio de trabajo del Jupyter Notebook. La carga de estos archivos se realizó utilizando `PlaintextCorpusReader` de NLTK, configurado con `PunktSentenceTokenizer` para la segmentación inicial en oraciones. La Figura 2.1 muestra la salida del script (celda de ejecución [5] del notebook) que confirma la carga exitosa de los documentos y presenta la primera oración de cada uno como evidencia.



```
[SUCCESS] Corpus cargado exitosamente desde '/content/'.
Archivos identificados en el corpus: ['doc1.txt', 'doc2.txt', 'doc3.txt', 'doc4.txt', 'doc5.txt']

--- Evidencia de Carga: Primera oración de cada documento ---
Documento 'doc1.txt': It may be interesting to some persons to learn how it came about that Vatsyayana was first brought t...
Documento 'doc2.txt': Leopold von Sacher - Masoch was born in Lemberg , Austrian Galicia , on January 27 , 1836 ....
Documento 'doc3.txt': Son of a merchant , Boccaccio di Chellino di Buonaiuto , of Certaldo in Val d ' Elsa , a little town...
Documento 'doc4.txt': The amatory motif is pervasive , timeless , and universal ....
Documento 'doc5.txt': In reading several dozen books on sex matters for the young with a view to selecting the best for my...
```

Figura 2.1: Confirmación de la carga del corpus y primera oración de cada documento.

2.2. Normalización de Documentos

La normalización es un paso esencial para preparar el texto para los análisis de similitud. Se aplicaron diferentes técnicas según el enfoque posterior (WordNet o embeddings).

2.2.1. Normalización para WordNet

Para las tareas que involucran WordNet (puntos 3 y 4), se implementó la función `normalizar_documento_wordnet`. Esta función realiza los siguientes pasos:

1. Segmentación en oraciones (`nltk.sent_tokenize`).
2. Tokenización en palabras dentro de cada oración (`nltk.word_tokenize`).
3. Conversión a minúsculas.
4. Etiquetado gramatical (POS tagging) con `nltk.pos_tag`.
5. Conversión de etiquetas POS del formato Penn Treebank al formato de WordNet utilizando la función auxiliar `get_wordnet_pos`.
6. Filtrado de tokens no alfabéticos y eliminación de stopwords (utilizando la lista en inglés de NLTK).
7. Lematización de los tokens restantes utilizando `WordNetLemmatizer` y su etiqueta POS de WordNet correspondiente (solo para sustantivos, verbos, adjetivos y adverbios).

La justificación de estos pasos radica en que WordNet opera sobre lemas y requiere conocer la categoría gramatical de las palabras para acceder a los synsets correctos. La eliminación de stopwords y no alfabéticos reduce el ruido y enfoca el análisis en palabras con mayor carga semántica. La Figura 2.2 muestra la salida del proceso de normalización para WordNet (celda de ejecución [6] del notebook), presentando los primeros 10 tokens normalizados (lema, etiqueta_wordnet) de cada documento.

```

--- Normalizando documentos para análisis con WordNet ---
Procesando 'doc1.txt'...
Primeros 10 tokens normalizados para 'doc1.txt': [('interest', 'v'), ('person', 'n'), ('learn', 'v'), ('come', 'v'), ('vatsyayana', 'n'), ('first', 'n'), ('bring', 'v'), ('bear', 'v'), ('leopard', 'n')].
Procesando 'doc2.txt'...
Primeros 10 tokens normalizados para 'doc2.txt': [('leopard', 'n'), ('von', 'n'), ('bear', 'v'), ('leberg', 'n'), ('austrian', 'n'), ('galicia', 'n'), ('january', 'n'), ('son', 'n'), ('merchant', 'n'), ('boccaccio', 'n')].
Procesando 'doc3.txt'...
Primeros 10 tokens normalizados para 'doc3.txt': [('son', 'n'), ('merchant', 'n'), ('boccaccio', 'n'), ('chellino', 'n'), ('buonaiuto', 'n'), ('certaindo', 'n'), ('val', 'n'), ('amatory', 'n'), ('motif', 'n'), ('pervasive', 'a')].
Procesando 'doc4.txt'...
Primeros 10 tokens normalizados para 'doc4.txt': [('amatory', 'n'), ('motif', 'n'), ('pervasive', 'a'), ('timeless', 'n'), ('universal', 'n'), ('phase', 'n'), ('manifest', 'n'), ('read', 'v'), ('several', 'a'), ('dozen', 'n')].
Procesando 'doc5.txt'...
Primeros 10 tokens normalizados para 'doc5.txt': [('read', 'v'), ('several', 'a'), ('dozen', 'n'), ('book', 'n'), ('sex', 'n'), ('matter', 'n'), ('young', 'a'), ('view', 'n')].
[SUCCESS] Normalización para WordNet completada.

```

Figura 2.2: Evidencia de la normalización de documentos para análisis con WordNet.

2.2.2. Normalización para Embeddings

- **GloVe:** Requiere palabras en minúsculas. Para la búsqueda de palabras similares al verbo más frecuente (punto 5), se utilizó el lema del verbo (obtenido de la normalización para WordNet) y se buscó directamente en el modelo GloVe, que está pre-entrenado con texto en minúsculas.
- **BERT:** Utiliza su propio tokenizer especializado (WordPiece para `bert-base-uncased`). A BERT se le deben pasar las frases con un preprocesamiento mínimo. Para el punto 6, se usaron las frases representativas extraídas (ver Sección 2.4) directamente con el tokenizer de BERT.


La justificación es que los modelos de embedding tienen sus propios requisitos de tokenización y preprocesamiento, y aplicar normalizaciones agresivas podría ser contraproducente o innecesario.

2.3. Similitud de Palabras con Synsets (WordNet)

En esta sección, para cada uno de los cinco documentos, se identificó el verbo más común y el sustantivo más frecuente a partir de los tokens normalizados para WordNet. Luego, utilizando la función `obtener_palabras_similares_wordnet`, se encontraron los 5 términos más similares a cada una de estas palabras (verbo y sustantivo) empleando dos métricas de similitud de WordNet: `path_similarity` y `wup_similarity`.

La `path_similarity` se basa en la longitud de la ruta más corta entre dos synsets en la jerarquía de WordNet. La `wup_similarity` (Wu-Palmer) considera la profundidad del synset ancestro común más específico de los dos synsets, así como la profundidad de los propios synsets.

Las Figuras 2.3 a 2.7 muestran los resultados obtenidos de la celda de ejecución [8] del notebook para cada documento. Cada figura presenta el verbo/sustantivo más frecuente, su frecuencia, y las cinco palabras más similares según cada métrica con su respectiva puntuación de similitud.

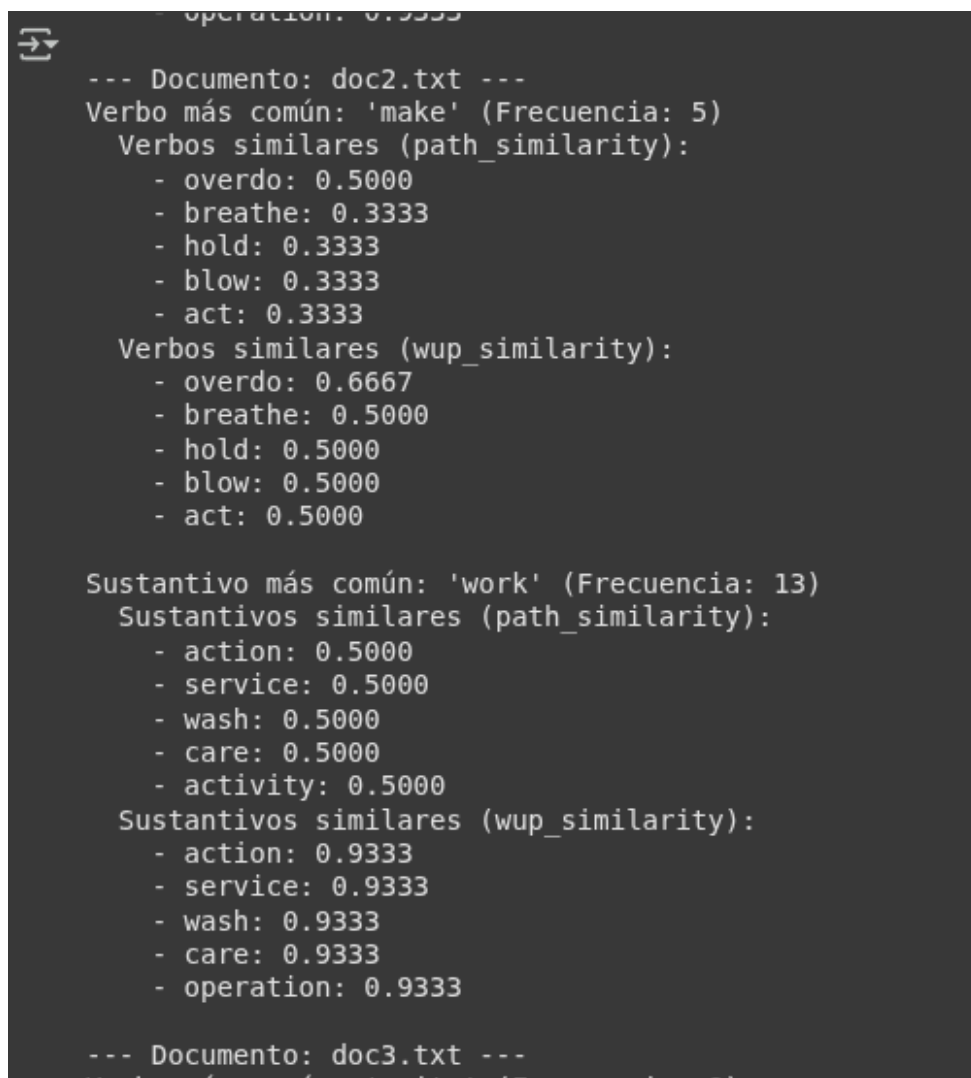


```
--- Calculando Similitud de Palabras con Synsets ---

--- Documento: doc1.txt ---
Verbo más común: 'call' (Frecuencia: 8)
  Verbos similares (path_similarity):
    - label: 0.5000
    - refer: 0.5000
    - dub: 0.5000
    - style: 0.5000
    - baptize: 0.5000
  Verbos similares (wup_similarity):
    - refer: 0.8571
    - dub: 0.8571
    - style: 0.8571
    - baptize: 0.8571
    - rename: 0.8571

Sustantivo más común: 'work' (Frecuencia: 11)
  Sustantivos similares (path_similarity):
    - action: 0.5000
    - service: 0.5000
    - wash: 0.5000
    - care: 0.5000
    - activity: 0.5000
  Sustantivos similares (wup_similarity):
    - action: 0.9333
    - service: 0.9333
    - wash: 0.9333
    - care: 0.9333
    - operation: 0.9333
```

Figura 2.3: Similitud de palabras con Synsets (WordNet) para doc1.txt.

A terminal window with a dark background and light gray text. It displays the output of a WordNet similarity analysis for the file 'doc2.txt'. The output is organized into sections for verbs and nouns, each showing the most common word and its similarity scores with related words.

```
operation: 0.9333
--- Documento: doc2.txt ---
Verbo más común: 'make' (Frecuencia: 5)
Verbos similares (path_similarity):
- overdo: 0.5000
- breathe: 0.3333
- hold: 0.3333
- blow: 0.3333
- act: 0.3333
Verbos similares (wup_similarity):
- overdo: 0.6667
- breathe: 0.5000
- hold: 0.5000
- blow: 0.5000
- act: 0.5000

Sustantivo más común: 'work' (Frecuencia: 13)
Sustantivos similares (path_similarity):
- action: 0.5000
- service: 0.5000
- wash: 0.5000
- care: 0.5000
- activity: 0.5000
Sustantivos similares (wup_similarity):
- action: 0.9333
- service: 0.9333
- wash: 0.9333
- care: 0.9333
- operation: 0.9333

--- Documento: doc3.txt ---
```

Figura 2.4: Similitud de palabras con Synsets (WordNet) para doc2.txt.

```
- operation: 0.9333

--- Documento: doc3.txt ---
Verbo más común: 'write' (Frecuencia: 8)
  Verbos similares (path_similarity):
    - draw: 0.5000
    - write_off: 0.5000
    - verse: 0.5000
    - dramatize: 0.5000
    - rewrite: 0.5000
  Verbos similares (wup_similarity):
    - draw: 0.8571
    - write_off: 0.8571
    - verse: 0.8571
    - dramatize: 0.8571
    - rewrite: 0.8571

Sustantivo más común: 'boccaccio' (Frecuencia: 23)
  Sustantivos similares (path_similarity):
    - poet: 0.5000
    - bard: 0.3333
    - Racine: 0.3333
    - Marlowe: 0.3333
    - elegist: 0.3333
  Sustantivos similares (wup_similarity):
    - poet: 0.9524
    - bard: 0.9091
    - Racine: 0.9091
    - Marlowe: 0.9091
    - elegist: 0.9091

--- Documento: doc4.txt ---
```

Figura 2.5: Similitud de palabras con Synsets (WordNet) para doc3.txt.

```
- Marlowe: 0.9091
- elegist: 0.9091

--- Documento: doc4.txt ---
Verbo más común: 'present' (Frecuencia: 2)
Verbos similares (path_similarity):
- bring_home: 0.5000
- show: 0.5000
- give: 0.3333
- peep: 0.3333
- flash: 0.3333
Verbos similares (wup_similarity):
- bring_home: 0.8000
- breathe: 0.4000
- hold: 0.4000
- blow: 0.4000
- act: 0.4000

Sustantivo más común: 'potion' (Frecuencia: 5)
Sustantivos similares (path_similarity):
- beverage: 0.5000
- elixir: 0.5000
- philter: 0.5000
- food: 0.3333
- cooler: 0.3333
Sustantivos similares (wup_similarity):
- elixir: 0.9474
- philter: 0.9474
- beverage: 0.9412
- elixir_of_life: 0.9000
- cooler: 0.8889
```

Figura 2.6: Similitud de palabras con Synsets (WordNet) para doc4.txt.

```

--- Documento: doc5.txt ---
Verbo más común: 'give' (Frecuencia: 8)
  Verbos similares (path_similarity):
    - infect: 0.5000
    - deliver: 0.5000
    - award: 0.5000
    - breathe: 0.3333
    - hold: 0.3333
  Verbos similares (wup_similarity):
    - infect: 0.6667
    - deliver: 0.6667
    - award: 0.6667
    - breathe: 0.5000
    - hold: 0.5000

Sustantivo más común: 'sex' (Frecuencia: 20)
  Sustantivos similares (path_similarity):
    - perversion: 0.5000
    - bondage: 0.5000
    - outercourse: 0.5000
    - safe_sex: 0.5000
    - conception: 0.5000
  Sustantivos similares (wup_similarity):
    - perversion: 0.9231
    - bondage: 0.9231
    - outercourse: 0.9231
    - safe_sex: 0.9231
    - conception: 0.9231

```

Figura 2.7: Similitud de palabras con Synsets (WordNet) para doc5.txt.

2.4. Similitud de Documentos con Synsets (WordNet)

Para esta tarea, primero se extrajo la frase más representativa de la introducción de cada uno de los cinco libros. Se utilizó el algoritmo TextRank, implementado a través de la biblioteca Sumy y su función `TextRankSummarizer`. La función `extraer_frase_representativa_textrank` se encargó de esta tarea, extrayendo una única oración por documento. La Figura 2.8 (primera parte de la salida de la celda de ejecución [9]) muestra las frases representativas extraídas para cada documento.

```

--- Extrayendo Frases Representativas con TextRank ---
Documento 'doc1.txt': "The date of the 'Jayamangla' is fixed between the tenth and thirteenth centuries A.D., because while treating of the sixty-four arts an example is
Documento 'doc2.txt': "By this is meant the desire on the part of the individual affected of desiring himself completely and unconditionally subject to the will of a pe
Documento 'doc3.txt': "Despite his complaints of the malevolence of his critics in the Proem to the Fourth Day of the Decameron, he had no lack of appreciation on the pe
Documento 'doc4.txt': "In its various mutations, its protean diversities, it is the love-potion, the philtre, the mystic concoction that, once quaffed, will instil love
Documento 'doc5.txt': "We give to young folks, in their general education, as much as they can grasp of science and ethics and art, and yet in their sex education, which

--- Calculando Similitud de Frases Representativas con Path Similarity (WordNet) ---
Frase base (de 'doc5.txt'): "We give to young folks, in their general education, as much as they can grasp of science and ethics and art, and yet in their sex education, which
Similitud con frase de 'doc2.txt': 0.2869
Similitud con frase de 'doc3.txt': 0.2379
Similitud con frase de 'doc4.txt': 0.2389
Similitud con frase de 'doc5.txt': 1.0000

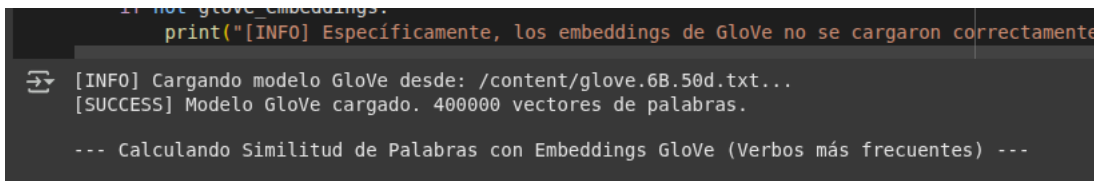
```

Figura 2.8: Frases representativas extraídas de cada documento mediante TextRank.

Posteriormente, se eligió la frase representativa del documento `doc5.txt` como base de comparación. Se calculó la similitud entre esta frase base y las frases representativas de los otros cuatro documentos (`doc1.txt` a `doc4.txt`, además de la comparación consigo misma). La similitud se midió utilizando la función `document_path_similarity_wordnet`. Esta función convierte cada frase en una lista de synsets (utilizando `doc_to_synsets_wordnet`) y luego calcula una similitud promedio basada en `path_similarity` entre los synsets de las dos frases de manera simétrica. Los resultados de estas comparaciones (segunda parte de la salida de la celda de ejecución [9]) se muestran en la Figura 2.8.

2.5. Similitud de Palabras con Embedding (GloVe)

En esta sección, se utilizó el modelo pre-entrenado de GloVe "Wikipedia 2014 + Gigaword 5" (vectores de 50 dimensiones, `glove.6B.50d.txt`). Se cargó el modelo GloVe mediante la función `cargar_glove_model`. La Figura 2.9 (primera parte de la salida de la celda de ejecución [10]) muestra el mensaje de confirmación de la carga exitosa del modelo GloVe, indicando que 400,000 vectores de palabras fueron cargados.



```
if not glove_embeddings:
    print("[INFO] Específicamente, los embeddings de GloVe no se cargaron correctamente")

[INFO] Cargando modelo GloVe desde: /content/glove.6B.50d.txt...
[SUCCESS] Modelo GloVe cargado. 400000 vectores de palabras.

--- Calculando Similitud de Palabras con Embeddings GloVe (Verbos más frecuentes) ---
```

Figura 2.9: Estado de la carga del modelo GloVe.

Con el modelo GloVe cargado, para cada documento, se tomó el verbo más frecuente (identificado en la Sección 2.3) y se utilizó la función `find_closest_glove_embeddings` para hallar los 5 términos más similares. Esta función calcula la similitud coseno entre el embedding del verbo objetivo y los embeddings de las demás palabras en el vocabulario de GloVe. La figura 2.10 (correspondientes a la segunda parte de la salida de la celda de ejecución [10]) presentan los resultados para cada documento, mostrando el verbo y sus 5 términos GloVe más similares con sus puntuaciones de similitud coseno.

```

--- Documento: doc1.txt ---
Verbo más común (lema): 'call'
Los 5 términos más similares a 'call' según GloVe (coseno):
- calls: 0.8872
- calling: 0.8769
- asking: 0.8592
- ask: 0.8572
- answer: 0.8378

--- Documento: doc2.txt ---
Verbo más común (lema): 'make'
Los 5 términos más similares a 'make' según GloVe (coseno):
- making: 0.9406
- take: 0.9393
- come: 0.9354
- give: 0.9349
- need: 0.9262

--- Documento: doc3.txt ---
Verbo más común (lema): 'write'
Los 5 términos más similares a 'write' según GloVe (coseno):
- writing: 0.8501
- read: 0.8217
- publish: 0.7848
- notes: 0.7766
- books: 0.7764

--- Documento: doc4.txt ---
Verbo más común (lema): 'present'
Los 5 términos más similares a 'present' según GloVe (coseno):
- same: 0.8586
- which: 0.8549
- of: 0.8475
- there: 0.8450
- however: 0.8408

--- Documento: doc5.txt ---
Verbo más común (lema): 'give'
Los 5 términos más similares a 'give' según GloVe (coseno):
- take: 0.9359
- make: 0.9349
- giving: 0.9313
- need: 0.8977
- put: 0.8923

```

Figura 2.10: Similitud de palabras con GloVe para el verbo más frecuente de `doc1.txt`.

2.6. Similitud de Documentos con Embedding (BERT)

Para calcular la similitud entre frases representativas utilizando BERT, primero se cargó el modelo `bert-base-uncased` y su tokenizer correspondiente desde la biblioteca Transformers. La Figura 2.11 (primera parte de la salida de la celda de ejecución [11]) muestra la confirmación de la carga exitosa del modelo y el tokenizer.

```

[INFO] Cargando tokenizer y modelo BERT (bert-base-uncased)...
/usr/local/lib/python3.11/dist-packages/huggingface_hub/utils/_auth.py:94: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (https://huggingface.co/settings/tokens), set it as secret in your Google Colab and restart
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.
warnings.warn(
[SUCCESS] Tokenizer y modelo BERT cargados.

```

Figura 2.11: Estado de la carga del modelo BERT y su tokenizer.

Una vez cargado el modelo, se utilizó la función `get_bert_sentence_embedding` para obtener un vector de embedding para cada una de las frases representativas extraídas en la Sección 2.4. Esta función tokeniza la frase, la pasa a través del modelo BERT y toma el embedding del token especial [CLS] de la última capa oculta como representación de la frase completa.

Finalmente, se tomó el embedding de la frase representativa del documento `doc1.txt` como base y se calculó la similitud coseno con los embeddings de las frases representativas de los otros cuatro documentos (y consigo misma). La Figura 2.12 (segunda parte de la salida de la celda de ejecución [11]) presenta estas similitudes.

```
--- Calculando Similitud de Frases con Embeddings BERT (Coseno) ---
Frase base (de 'doc1.txt'): "The date of the 'Jayamangla' is fixed between the tenth and thirteenth centuries A.D., because while treating of the sixty-four arts an exa
Similitud con frase de 'doc1.txt' ("The date of the 'Jayamangla' is fixed between the ..."): 1.0000
Similitud con frase de 'doc2.txt' ("By this is meant the desire on the part of the ind..."): 0.7630
Similitud con frase de 'doc3.txt' ("Despite his complaints of the malevolence of his c..."): 0.6534
Similitud con frase de 'doc4.txt' ("In its various mutations, its protean diversities,..."): 0.7457
Similitud con frase de 'doc5.txt' ("We give to young folks, in their general education..."): 0.7806
```

Figura 2.12: Similitud de la frase representativa de `doc1.txt` con las de otros documentos, usando embeddings BERT y similitud coseno.

Capítulo 3

Conclusiones Generales

Este experimento se centró en la comparación de diversos enfoques para medir la similitud entre palabras y documentos (representados por frases). Se emplearon técnicas basadas en conocimiento léxico estructurado (WordNet con métricas `path_similarity` y `wup_similarity`) y técnicas basadas en representaciones vectoriales densas (GloVe y BERT). El corpus consistió en las introducciones de cinco libros del Proyecto Gutenberg, con temáticas relacionadas al amor, la sexualidad y la literatura erótico-afectiva, lo que proveyó un contexto interesante para evaluar las distintas metodologías.

3.1. Análisis de Resultados

3.1.1. Similitud de Palabras

WordNet (`path_similarity` vs. `wup_similarity`)

Ambas métricas de WordNet identificaron palabras semánticamente cercanas dentro de su jerarquía taxonómica, principalmente sinónimos o términos conceptualmente relacionados. Por ejemplo, para el verbo 'call', ambas métricas sugirieron términos como 'refer', 'dub', y 'style'. Para el sustantivo 'work', se encontraron 'action', 'service', y 'care'. Una diferencia notable fue que `wup_similarity` consistentemente arrojó puntajes de similitud más altos en comparación con `path_similarity`. Esto se atribuye a que `wup_similarity` no solo considera la distancia en la jerarquía, sino también la profundidad de los synsets y su ancestro común más específico (LCS), lo que tiende a valorar más la cercanía conceptual específica. En general, `wup_similarity` pareció ofrecer una medida más intuitiva de la cercanía semántica.^{al} premiar la especificidad compartida. Ambas métricas demostraron ser efectivas para palabras bien establecidas en el lexicon de WordNet, como se vio con el término 'boccaccio' siendo relacionado con 'poet'.

GloVe (Similitud Coseno)

GloVe identificó palabras que tienden a co-ocurrir o que se utilizan en contextos similares dentro de su vasto corpus de entrenamiento (Wikipedia + Gigaword). Los tipos de palabras similares incluyeron:

- Formas flexionadas o derivadas del término original (e.g., 'call' → 'calls', 'calling').
- Palabras temáticamente relacionadas o conceptos asociados (e.g., 'call' → 'asking', 'answer'; 'write' → 'read', 'publish', 'books').

Un caso particular fue el verbo 'present' (del `doc4.txt`), para el cual GloVe devolvió palabras como 'same', 'which', 'of'. Esto sugiere que, para palabras muy comunes y polisémicas, el embedding de GloVe puede capturar un uso funcional o estructural muy general derivado de su alta frecuencia de co-ocurrencia con una amplia gama de palabras en el corpus, en lugar de un significado semántico específico y acotado. Comparado con WordNet, GloVe captura un tipo de similitud más asociativa y contextual, mientras que WordNet se enfoca en relaciones léxicas curadas.

Limitaciones Observadas en Similitud de Palabras

- **WordNet:** Su cobertura de vocabulario, aunque extensa, no es exhaustiva para neologismos o jerga. Su sensibilidad al contexto es limitada, operando principalmente a nivel de lema y su sentido más común. La calidad del preprocesamiento (POS tagging, lematización) es crucial.
- **GloVe:** Aunque su vocabulario es grande (400k palabras para `glove.6B.50d.txt`), pueden existir palabras fuera de vocabulario (OOV). Asigna un único vector por palabra, lo que dificulta el manejo de la polisemia, ya que el vector representa una amalgama de todos los contextos en los que aparece la palabra.

3.1.2. Similitud de Documentos (Frases Representativas)

Se extrajeron frases representativas de cada introducción utilizando TextRank y se compararon.

WordNet (`document_path_similarity`)

Se utilizó la frase representativa de `doc5.txt` ("We give to young folks... sex education...") como base. Las similitudes obtenidas fueron:

- vs. `doc2.txt` (Sacher-Masoch, deseo y sumisión): 0.2869 (la más alta, reflejando una conexión temática con la sexualidad).
- vs. `doc3.txt` (Boccaccio, literario/histórico): 0.2379 (la más baja, indicando menor afinidad temática).
- vs. `doc4.txt` (motivo amatorio, pociones): 0.2389 (baja, aunque relacionada con el amor/deseo, es más abstracta).

Los puntajes fueron relativamente bajos, lo cual es característico de promediar `path_similarity` entre múltiples synsets, pero el ordenamiento relativo fue coherente. El método de agregación (promediar las mejores similitudes synset a synset) puede simplificar en exceso, perdiendo la estructura sintáctica y el significado composicional de la frase.

BERT (Similitud Coseno sobre Embeddings [CLS])

Se utilizó la frase representativa de `doc1.txt` ("The date of the 'Jayamangla'...") como base. Las similitudes obtenidas fueron:

- vs. `doc5.txt` (educación sexual): 0.7806 (la más alta). Temáticamente, el Kama Sutra (`doc1`) y la educación sexual son los más cercanos.
- vs. `doc2.txt` (Sacher-Masoch): 0.7630 (segunda más alta). También temáticamente relevante.
- vs. `doc4.txt` (motivo amatorio): 0.7457.
- vs. `doc3.txt` (Boccaccio): 0.6534 (la más baja).

Los puntajes de BERT fueron notablemente más altos y ofrecieron una discriminación más clara y semánticamente coherente. La capacidad de BERT para capturar el contexto bidireccional, utilizando el embedding del token [CLS] como representación de la frase completa, se reflejó en una evaluación de similitud que parece más alineada con la comprensión humana. El orden de similitud obtenido con BERT fue muy intuitivo dada la temática de los libros.

3.1.3. Impacto de la Extracción de Frases y Desafíos

La calidad de la frase representativa extraída por TextRank es crucial; si no es un buen resumen, la comparación de similitud será defectuosa. En este caso, las frases extraídas parecieron ser resúmenes razonables. Un desafío interpretativo surgió con el verbo 'present' y GloVe, destacando que la similitud basada en co-ocurrencia no siempre equivale a similitud semántica directa. La elección del "primer synset.^{en} WordNet es una simplificación; métodos más avanzados podrían incluir desambiguación del sentido de la palabra.

3.2. Conclusiones Finales

La práctica permitió una comparación efectiva de diferentes enfoques para la medición de similitud en NLP.

- Para la **similitud de palabras individuales**, WordNet es valioso por sus relaciones léxicas explícitas y curadas, siendo `wup_similarity` a menudo más intuitiva que `path_similarity`. GloVe, por su parte, destaca en encontrar similitudes asociativas y contextuales basadas en el uso del lenguaje en grandes corpus. La elección entre ellos depende del tipo específico de similitud que se busque.
- Para la **similitud de frases o documentos**, BERT demostró una capacidad superior. Su habilidad para procesar el contexto completo de una secuencia resulta en mediciones de similitud que se alinean más estrechamente con la intuición humana y la coherencia temática, superando a los métodos basados en la agregación de similitudes de palabras individuales como el empleado con WordNet.

Se evidencia un compromiso entre interpretabilidad y poder representacional. WordNet es más transparente pero limitado en cobertura y sensibilidad contextual. Los embeddings como GloVe y, especialmente, BERT, capturan matices más ricos del lenguaje pero pueden ser más cajas negras computacionalmente más intensivos (aunque aquí se usaron modelos pre-entrenados, su inferencia aún consume recursos).

En resumen, los métodos basados en embeddings contextuales como BERT son herramientas muy potentes para tareas de similitud semántica a nivel de secuencias de texto. Los recursos léxicos como WordNet siguen siendo útiles para análisis a nivel de palabra y para comprender relaciones semánticas estructuradas. La selección de la técnica adecuada dependerá siempre de los requisitos específicos de la tarea, los recursos disponibles y la naturaleza de la similitud que se pretende identificar.

Bibliografía

- [1] G. A. Miller, “Wordnet: a lexical database for english,” in *Communications of the ACM*, vol. 38, pp. 39–41, ACM, 1995.
- [2] J. Pennington, R. Socher, and C. D. Manning, “Glove: Global vectors for word representation,” in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.
- [3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Association for Computational Linguistics, 2019.
- [4] Project Gutenberg, “Project Gutenberg,” 2024. Disponible en: <https://www.gutenberg.org/>.
- [5] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python: analyzing text with the natural language toolkit*. O’Reilly Media, Inc.", 2009.
- [6] R. Mihalcea and P. Tarau, “Textrank: Bringing order into text,” in *Proceedings of the 2004 conference on empirical methods in natural language processing*, pp. 404–411, 2004.
- [7] M. Matulák and other contributors, “Sumy: Module for automatic summarization of texts and html pages..” <https://github.com/miso-belica/sumy>, 2024. Consultado en 2024.
- [8] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, and A. M. Rush, “Transformers: State-of-the-art natural language processing,” in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, (Online), pp. 38–45, Association for Computational Linguistics, Oct. 2020.