Estudio de la similitud de artículos científicos mediante técnicas de Procesamiento de Lenguaje Natural

Durán López Francisco Javier

fjdurlop0@gmail.com

Prof. Sánchez Velázquez Octavio Augusto

Facultad de Ingeniería

Universidad Nacional Autónoma de México

17 de agosto de 2021

Abstract

It is known that each time, scientific articles are more open and with fewer restrictions thanks to the internet, therefore the volume of these increases very quickly and researchers need tools that help automate repetitive processes, reducing the time they need to do an academic or professional job. This article seeks to provide researchers with a fast and efficient method to obtain articles similar to those considered for a certain project or line of research. To meet the objective, the method is based on the representation of texts with the TF-IDF (Term frequency - Inverse document frequency) model and the cosine distance function, likewise, a preprocessing is carried out to normalize the corpus and improve results. The proposed methodology clearly shows that similar articles can be found thanks to a comparison of previously labeled data.

Resumen

Se sabe que cada vez, los artículos científicos son más abiertos y tienen menos restricciones gracias a internet, por lo tanto, el volumen de estos aumenta muy rápido y los investigadores necesitan de herramientas que ayuden a automatizar procesos repetitivos, disminuyendo el tiempo que necesitan para hacer un trabajo académico o profesional. En este artículo, se busca proporcionar a investigadores, un método rápido y eficiente para obtener artículos similares a los que se tienen en consideración para cierto proyecto o línea de investigación. Para cumplir con el objetivo, el método se basa en la representación de textos con el modelo TF-IDF (del inglés Term frequency – Inverse Document Frequency) y la función de Distancia Coseno, así mismo, se realiza un preprocesamiento para normalizar el corpus y mejorar los resultados. La metodología propuesta, muestra claramente que se puede encontrar artículos similares gracias a una comparación de datos previamente etiquetados.

I. Introducción

Año con año la cantidad de artículos científicos aumenta, debido a la gran facilidad de compartir información, por eso muchas veces los expertos deben estar actualizándose con lo más reciente de su campo de estudio, para lo cual deben de revisar decenas de artículos publicados año con año o incluso artículos viejos que han sentado las bases de su campo de estudio, para esto es necesario que se revisen estos artículos, sin embargo cuando se

tienen muchos, para una sola persona o un solo grupo de investigación, es imposible revisar completamente decenas de estos en poco tiempo para poder publicar avances. Por eso la automatización de la revisión de artículos, o análisis de estos, tiene una gran importancia en el mundo académico y profesional. Entre los métodos más usados para poder procesar y analizar grandes cantidades de textos es usar técnicas de Procesamiento de Lenguaje Natural (PLN), ya que con estas técnicas la extracción de información de grandes cantidades de texto es mucho más rápida y eficiente. [4]

Entre estas técnicas están la clasificación de textos de acuerdo con ciertos tópicos, resumen de texto automático, extracción del tópico de textos, etc., y para realizar este tipo de métodos, se necesitan métricas con las cuales guiar los métodos o algoritmos. Entre estos, desde hace algunas décadas, la similitud de textos juega un papel muy importante. [3]

Cuando las personas expertas en algún campo buscan información acerca de cierto tema que quieren investigar, se encuentran con el problema de elegir que artículo leer, debido a la gran masa de textos que hay, por eso este artículo busca resolver o ayudar a estas personas a elegir de entre cierta cantidad de artículos, los artículos que más les convenga leer o revisar, esto con base a la similitud de un artículo del tema que busca o está desarrollando.

II. Estado de la técnica

Existen muchos trabajos donde se ha estudiado ampliamente las similitudes entre textos, lo cual se puede ver en las encuestas (Surveys) del estado de la técnica [3], algunas personas están enfocadas en la teoría y otros investigan aplicaciones de estas técnicas en el mundo real [1,2,6] entre esas aplicaciones se pueden encontrar desde noticias o artículos en internet hasta documentos oficiales de gobiernos.

Se han desarrollado muchas funciones para obtener similitud entre textos, una vez que ya se tiene el texto normalizado y en forma vectorial, entre las cuales están la distancia euclidiana, distancia coseno, distancia Manhattan, etc. La distancia coseno a pesar de ser una medida clásica, por su sencillez de obtener y comprender, sigue siendo una métrica para trabajos actuales, debido a los buenos resultados que entrega. [5,6]

Los modelos de representación de cadenas de texto más usados son los basados en bolsa de palabras como el modelo TF-IDF, pero también existen otros como word2vec que han sido utilizados en los últimos años, este modelo fue creado en 2013 por Google basado en aprendizaje profundo o “Deep Learning”, para computar y generar representaciones vectoriales de las palabras que capturan similitudes contextuales y semánticas. [6,9]

Las aplicaciones de estos métodos ayudan a reducir considerablemente el tiempo que una persona gasta en leer o procesar manualmente grandes cantidades de texto, Singh, R., y Singh, S. [1] miden la similitud de noticias de varias fuentes que tratan de un mismo suceso para identificar los artículos principales. Alanoca, H. A., Chire, J., y Oblitas, J. [2] usan estas técnicas para encontrar diferencias entre los planes del gobierno de los candidatos a presidencia de Perú y los objetivos de las metas de desarrollo sustentable de la ONU. Por su parte, Novotná, T. [6] analiza decisiones judiciales relacionadas con un asunto legal similar, lo cual los abogados hacen normalmente en la toma de decisiones manualmente, el caso se estudia con el Tribunal Supremo de la República Checa.

En cuanto al procesamiento de artículos científicos, dada la relevancia para los académicos, de obtener cada vez mejor información y más rápida, algunos expertos procesan este tipo de textos con técnicas de PLN como Reconocimiento de Entidad con Nombre y Extracción de Relaciones de Entidad, (Named Entity Recognition y Entity Relationship Extraction), como Gao, X., Tan, R., & Li, G. [4] proponen un framework general para tareas de extracción de información.

Otros además de hacer análisis con ayuda de los métodos de similitud entre textos, realizan

clustering o agrupación de artículos basándose en las similitudes; Rinartha, K., & Kartika, L. G. S. [7] agrupan artículos científicos de acuerdo con su contenido obteniendo resultados con un alto porcentaje de clasificaciones correctas.

Masumeh Islami Nasab [8] propuso un método para calcular la similitud semántica entre artículos, usando WordNet para encontrar asociaciones semánticas de palabras, esta es una base de datos del idioma inglés que agrupa palabras en inglés en conjuntos de sinónimos, proporcionando definiciones cortas y generales, almacenando relaciones semánticas entre los conjuntos de sinónimos.

III. Marco teórico

En el campo de estudio del Procesamiento de Lenguaje Natural, se trabaja con un corpus que es un conjunto de textos compilados con un fin específico que idealmente es representativo de los fenómenos que se pretenden estudiar y balanceado con relación a la lengua.

Muchas veces se requiere hacer un análisis complejo de grandes cantidades de texto o de algún corpus específico, para lo cual obtener un valor de que tanto se parecen ciertos artículos con otros es indispensable a la hora de aplicar alguna técnica de PLN. Esto es posible usando varias métricas y agrupando documentos similares. Para poder medir la similitud entre textos, en las últimas tres décadas [3] se han propuesto varias técnicas de similitud, entre las cuales han ido cambiando dos cosas: el método para obtener distancia entre palabras, sentencias y documentos, y el método para representar estas estructuras de texto como números para un procesamiento fácil en la computadora, entre estos métodos están los basados en cadenas de texto, en corpus, etc. [3]

En este trabajo la similitud se obtendrá con base al corpus que en este caso es un conjunto de resúmenes de “artículos académicos”, el método para representar el documento es usando el modelo de bolsa de palabras (TF-IDF. En cuanto a la función de distancia entre dos documentos se utilizó la distancia coseno.

Distancia coseno:

En lugar de medir la distancia entre dos puntos, se transforma al problema de ángulos correspondientes a los dos puntos en el espacio de vectores. La similitud es calculada midiendo el coseno del ángulo entre dos vectores.

$$Sim(Doc\ A, Doc\ B) = cos\theta = \frac{\overrightarrow{Doc\ A}\overrightarrow{Doc\ B}}{||Doc\ A|| \cdot ||Doc\ B||}$$

La representación del texto como vectores de números es importante en la mayoría de los trabajos de PLN y los métodos basados en el corpus toma en cuenta todo el vocabulario del corpus, por lo cual, a la hora de hacer análisis de similitudes entre textos, se quiere diferenciar entre documentos que si se parezcan con los que no, con base a todos los demás.

En los estudios más recientes este tipo de métodos se han utilizado de tres formas: con el modelo de la bolsa de palabras, representaciones de distribuciones y métodos de factorización de matrices, siendo el más utilizado el primero, la idea del modelo de bolsa de palabras es representar el documento como una combinación de una serie de palabras y a su vez los modelos basados en esta representación son BOW (Bolsa de palabras, Bag-of-Words), TD-IDF, LSA, etc.

BOW:

Representa el texto contando el numero de palabras que aparece en un documento y luego usa este conteo para medir la similitud.

TF-IDF:

Se basa en que una palabra aparece con mayor frecuencia en un documento y hay un gran número de documentos del corpus que contienen esta palabra. Sin embargo, a pesar de que las palabras aparecen frecuentemente, no tienen un significado especial para el documento en sí.

$$tf\_idf(w, d, D) = tf(w, d)\ x\ idf(w, D)$$

$$tf(w, d) = Freq(w, d)$$

$$idf(w, D) = log\frac{|D|}{N(w)}$$

Donde Freq(w,d) indica que tan seguido una palabra es usada en el documento, |D| indica el número del documento y N(w) es el número de palabras distintas (vocabulario) que aparecen en el documento.

IV. Configuración del experimento

Pregunta de investigación:

¿Cómo encontrar artículos similares a cierto artículo para una posible revisión de literatura usando distancia coseno entre documentos, para reducir el tiempo que un investigador tiene que tomarse manualmente a buscar artículos relacionados con su trabajo?

Los principales pasos de la metodología propuesta para responder a la pregunta de investigación se describen abajo:

1. Obtención de corpus

El corpus se obtuvo de Kaggle Datasets, sitio donde personas de todo el mundo comparten datos abiertos para que los demás puedan hacer análisis de inteligencia artificial, machine learning o ciencia de datos principalmente. [10]

Descripción del corpus:

El corpus es un conjunto de resúmenes de artículos de investigación, se utilizó originalmente para realizar predicciones de las etiquetas, hay un conjunto de entrenamiento y otro de prueba, pero para este trabajo sólo se utilizó el conjunto de entrenamiento, ya que no se hace algún entrenamiento de modelo de ML (Machine Learning) o de DL (Deep Learning). Estos documentos provienen de los siguientes 4 temas:

- Ciencias de la Computación
- Matemáticas
- Física
- Estadísticas

Sin embargo, pueden tener más de una sola etiqueta.

2. Preprocesamiento

Se realiza un normalizado de los documentos del corpus, con las siguientes operaciones o técnicas (Ver archivo utils.py y Figura 1):

- Extracción de caracteres acentuados
- Expansión de contracciones
- Conversión a minúsculas
- Eliminación de líneas extras
- Lematización del texto
- Eliminación de caracteres especiales
- Eliminación de espacios extra
- Eliminación de “Stop Words”

```
print("Comparación antes vs despues del preprocesamiento:
print("Texto antes de ser normalizado:\n")
print(corpus[1][:200]+"\n\n")
print("Texto después de ser normalizado:\n")
print(norm_corpus[1][:200])
```

```
Comparación antes vs despues del preprocesamiento:

Texto antes de ser normalizado:

we propose the framework considering optimal $t$-matchi
ngs excluding a prescribed $t$-factors inside bipartite
graphs. a proposed framework was the generalization of
a nonbipartite matching problem an


Texto después de ser normalizado:

propose framework considering optimal matchings excludi
ng prescribed factors inside bipartite graph proposed f
ramework wa generalization nonbipartite matching proble
m includes several problem triangle
```

Figura 1 Comparación antes vs después de normalización del corpus.

3. Representación de los documentos

Se utiliza el modelo TF-IDF, previamente explicado en el presente trabajo.

4. Obtención de similitud

Se obtiene una matriz de N x N donde N es el número de documentos y cada valor corresponde a la similitud coseno entre pares de documentos.

5. Obtención de artículos similares

Se ordenan los índices de los documentos, obteniendo los “n” documentos más similares a cada uno del corpus.

Se espera proveer a académicos y profesionales de un método sencillo y rápido de escoger artículos relevantes de acuerdo con la similitud de textos.

V. Análisis de resultados

La metodología propuesta se implementó usando Python 3. Para el experimento se obtuvo los documentos similares para cada uno, en 5 rangos (De los documentos ordenados de mayor a menor de acuerdo con su similitud con el documento en cuestión). Luego se obtuvo el promedio de coincidencias de etiquetas o de los tópicos de los documentos en ese rango con el documento en cuestión, abajo se describe el criterio de las coincidencias:

Tomando en cuenta que se tiene un artículo A del cual se ha ordenado los demás documentos con base a su similitud con A, sea B un artículo de estos, existe coincidencia del artículo B con A, sí alguna de las etiquetas de B coincide con las etiquetas de A, de esta manera se sabe por los datos previamente etiquetados que en al menos un tema, su contenido coincide. Si el documento B coincide con A en más de una etiqueta, sólo se indica una coincidencia.

Con base en esta métrica propuesta de coincidencias se obtiene este número entre el número total de coincidencias posibles (El caso en que todos los documentos del rango coinciden en al menos un tema con el artículo en cuestión), este valor puede observarse en la tercera columna de la Tabla 1.

Tabla 1. Resultados

Rango de documentos similares (1 es el mayor el cual equivale al mismo documento)	Promedio de coincidencias de etiquetas de todos los documentos respecto a los demás	Promedio (Valor anterior) / Total posible de coincidencias (En este caso se probó en rangos de 20 documentos)
1 - 21	16.31	0.815
1001 - 1021	11.76	0.588
2001 - 2021	10.29	0.515
3001 - 3021	9.42	0.471
4001 - 4021	8.75	0.438

De a cuerdo a la Tabla 1 se puede observar una tendencia en la segunda y tercera columna de que entre mayor sea el valor de similitud de los documentos (Del 1 al 21 son los de mayor similitud y del 4001 al 4021 son los de menor similitud de los rangos tomados a consideración) con el documento en cuestión, se tendrá una mayor coincidencia con el tópico de este.

Para una mejor visualización del resultado se muestran los siguientes ejemplos de lo obtenido, donde se puede observar en el ejemplo 1, que los temas del documento en cuestión son los mismos que el documento obtenido más parecido con el método (Ver Figura 3 y 5); en el ejemplo 2, se observa que los temas del documento más parecido coinciden en dos de tres (Ver Figura 7 y 9):

```
In [50]: # Visualización de resultados
         # Documento A
         norm_corpus[10]

Out[50]: 'study problem extracting selective connector consideri
         ng given set query vertex q \\subseteq v inside graph g
         v e selective connector wa subgraph g exhibit cohesiven
         ess property contains query vertex doe necessarily conn
         ect relaxing connectedness requirement allows connector
         detect multiple community tolerant outlier achieve intr
         oducing new measure network inefficiency instantiating
         search considering selective connector problem finding
         minimum inefficiency subgraph show minimum inefficiency
         subgraph problem wa nphard devise efficient algorithm a
         pproximate mean several case study inside variety appli
         cation domain human brain cancer food network show mini
         mum inefficiency subgraph produce highquality solution
         exhibiting desired behavior selective connector'
```

Figura 2 Documento en cuestión (Ejemplo 1).

```
utils.get_topics_of_doc(temas,10)

['Computer Science', 'Data Structures and Algorithms']
```

Figura 3 Temas del documento anterior (Ejemplo 1).

```
# Documento más parecido a A:
norm_corpus[matrix_idx[10][0]]
```

```
'study nphard problem motivated energyefficiently maint
aining connectivity symmetric wireless sensor communica
tion network given edgeweighted n vertex graph find con
nected spanning subgraph minimum cost cost wa determine
d letting vertex pay expensive edge incident inside sub
graph provide algorithm work inside polynomial time one
find set obligatory edge yield spanning subgraph \\log
n connected component also provide lineartime algorithm
reduces input graph consists tree together g additional
edge equivalent graph g vertex based obtain polynomialt
ime algorithm considering g\\in \\log n negative side s
how \\log n approximating difference optimal solution c
ost natural lower bound wa nphard presumably exact algo
rithm running inside ^ n time inside f \\cdot n^ time c
onsidering computable function f'
```

Figura 4 Documento con mayor similitud al documento anterior (Ejemplo 1).

```
In [55]: utils.get_topics_of_doc(temas,matrix_idx[10][0])
Out[55]: ['Computer Science', 'Data Structures and Algorithms']
```

Figura 5 Temas del documento con mayor similitud al documento anterior (Ejemplo 1).

```
# Visualización de resultados
# Documento A, Ejemplo 2
norm_corpus[7000]
```

```
'continuously rotating halfwave plate crhwp wa promising tool i
mprove sensitivity large angular scale inside cosmic microwave
background cmb polarization measurement crhwp single detector m
easure three stokes parameter q u thereby avoiding set systemat
ic error introduced mismatch inside property orthogonal detecto
r pair focus implementation crhwps inside large aperture telesc
ope e primary mirror wa larger current maximum halfwave plate d
iameter \\sim crhwp placed primary mirror focal plane inside co
nfiguration one need address intensity polarization \\rightarro
w p leakage optic becomes source f noise also cause differentia
l gain systematics arise cmb temperature fluctuation inside pap
er present performance crhwp installed inside polarbear experim
ent employ gregorian telescope primary illumination pattern crh
wp wa placed near prime focus primary secondary mirror find \\r
ightarrow p leakage wa larger expectation physical property pri
mary mirror resulting inside f knee mhz excess leakage could du
e imperfection inside detector system e detector nonlinearity i
nside responsivity timeconstant demonstrate however subtracting
leakage correlated intensity signal f noise knee frequency wa r
educed mhz \\ell \\sim considering scan strategy wa promising p
robe primordial bmode signal also discus method considering noi
se subtraction inside future project precise temperature contro
l instrumental component leakage reduction play key role'
```

Figura 6 Documento en cuestión (Ejemplo 2).

```
utils.get_topics_of_doc(temas,7000)
```

```
['Physics',
 'Cosmology and Nongalactic Astrophysics',
 'Instrumentation and Methods for Astrophysics']
```

Figura 7 Temas del documento anterior (Ejemplo 2).

```
# Documento más parecido a A, Ejemplo 2:
norm_corpus[matrix_idx[7000][0]]
```

```
'future cosmic microwave background cmb satellite mission aim u
se b mode polarization measure tensortoscalar ratio r sensitivi
ty ^ achieving goal require sufficient detector array sensitivi
ty also unprecedented control systematic error inherent cmb pol
arization measurement since polarization measurement derive dif
ference observation different time different sensor detector re
sponse mismatch introduce leakage intensity polarization thus l
ead spurious b mode signal expected primordial b mode polarizat
ion signal wa dwarfed known unpolarized intensity signal leakag
e could contribute substantially final error budget considering
measuring r help simulation approximate magnitude angular spect
rum spurious b mode signal resulting bandpass mismatch differen
t detector wa assumed detector calibrated considering example h
elp cmb dipole sensitivity primordial cmb signal ha perfectly m
atched consequently mismatch inside frequency bandpass shape de
tector introduces difference inside relative calibration galact
ic emission component simulate help range scanning pattern cons
idered considering future satellite mission find spurious contr
ibution r reionization bump large angular scale \\ell wa \\appr
ox ^ assuming large detector array percent sky masked show ampl
itude leakage depends angular coverage per pixel result scan pa
ttern'
```

Figura 8 Documento con mayor similitud al documento anterior (Ejemplo 2).

```
utils.get_topics_of_doc(temas,matrix_idx[7000][0])
```

```
['Physics', 'Cosmology and Nongalactic Astrophysics']
```

Figura 9 Temas del documento con mayor similitud al documento anterior (Ejemplo 2).

VI. Conclusiones

El uso del modelo TF-IDF junto con la función de similitud coseno, muestran resultados favorables conforme a la hipótesis de la investigación. Se logró implementar un método que encuentre extractos (Resúmenes) de artículos similares a cierto artículo con el cual el investigador esté trabajando, lo cual puede ahorrar tiempo en encontrar por lo menos artículos que sean de temas parecidos, ya que en cuestión de minutos es posible obtener los artículos más similares. Sin embargo, estos resultados podrían ser mejorados al considerar modelos más complejos de representación de textos y debería ser probado con un corpus que utilice todo el texto del artículo para poder generalizar más el método.

VII. Trabajo futuro

Entre los posibles trabajos próximos a experimentar, se espera realizar una comparación entre distintos modelos de representación de textos, y/o funciones de similitud diferentes. Sería interesante también, programar una herramienta que permita obtener directamente artículos de internet y obtener su similitud con el trabajo que se esté considerando, reduciendo así aún más el tiempo que los investigadores se toman en la elección de literatura.

Referencias

[1] Singh, R., & Singh, S. (2021). Text Similarity Measures in News Articles by Vector Space Model Using NLP. *Journal of The Institution of Engineers (India): Series B*, *102*(2), 329-338.

[2] Alanoca, H. A., Chire, J., & Oblitas, J. (2021). Comparative analysis of the government plans of the Peruvian presidential candidates, SDO (UN) and State Policies of the National Agreement based on NLP. *arXiv preprint arXiv:2104.01765*.

[3] Wang, J., & Dong, Y. (2020). Measurement of text similarity: a survey. *Information*, *11*(9), 421.

[4] Gao, X., Tan, R., & Li, G. (2020, March). Research on text mining of material science based on natural language processing. In *IOP Conference Series: Materials Science and Engineering* (Vol. 768, No. 7, p. 072094). IOP Publishing.

[5] Arunesh, Dr. P K Arunesh. (2016). TEXT MINING: TEXT SIMILARITY MEASURE FOR NEWS ARTICLES BASED ON STRING BASED APPROACH. 10.5281/zenodo.57373.

[6] Novotná, T. (2020). Document Similarity of Czech Supreme Court Decisions. *Masaryk University Journal of Law and Technology*, *14*(1), 105-122.

[7] Rinartha, K., & Kartika, L. G. S. (2019, August). Scientific article clustering using string similarity concept. In *2019 1st International Conference on Cybernetics and Intelligent System (ICORIS)* (Vol. 1, pp. 13-17). IEEE.

[8] Nasab, M. I., & Javidan, R. (2015). A new approach for finding semantic similar scientific articles. *Journal of Advanced Computer Science & Technology*, *4*(1), 53.

[9] Sarkar, D. (2016). *Text Analytics with python*. New York, NY, USA:: Apress.

[10] https://www.kaggle.com/abisheksudarshan/topic-modeling-for-research-articles