

IRWA Project: Part 1 - Procesamiento de texto y análisis exploratorio de los datos

Git URL: https://github.com/raquel-sb/IRWA_2023.git

1. Introducción

Este laboratorio constituye la continuación del proyecto en el que nos enfocaremos durante todo el trimestre. A partir de los avances de la primera práctica, es decir, el pre-procesamiento de nuestros tweets, hemos llevado a cabo la indexación de los documentos del conjunto de datos proporcionado y evaluado el sistema de búsqueda resultante.

Así pues, nuestra meta es determinar la puntuación de cada documento en relación con la consulta, y, por lo tanto, evaluar estos resultados teniendo en cuenta el orden deseado de las puntuaciones mencionadas.

1.1. Mejora Parte 1

Queríamos remarcar que, después del feedback recibido de la Parte 1 del Proyecto, hemos decidido, a la hora de realizar el PreProcessed Tweet, añadir el Hashtag PreProcesado también. En otras palabras, hemos conseguido separar los hashtags por palabras y, obviamente, introducirlos en "PreProcessed_Tweet" con el mismo formato, para así, facilitar al algoritmo la búsqueda de tweets relacionados con la "query" y obtener unos resultados más precisos.

2. Funciones Desarrolladas

2.1. Función `create_index()`

Esta función crea un índice invertido de un Tweet, que es una estructura de datos que permite buscar rápidamente tweets que contengan un término/s dado. La función recibe una lista de líneas, que son un diccionario con el id, el doc_xxx y el texto de cada tweet del dataset. La función devuelve un diccionario con los términos como claves y una lista de tuplas con los documentos donde aparecen estas claves (y las posiciones) como valores. El índice invertido se puede usar para mejorar la eficiencia y la relevancia de las consultas de búsqueda.

2.2. Función `search()`

Esta función recibe una consulta de términos y un índice invertido. Luego, busca los documentos (Tweets) que contienen al menos uno de los términos de la consulta en el índice invertido y devuelve una lista de documentos relevantes. Esto mejora la eficiencia de las búsquedas en un conjunto de Tweets al aprovechar la estructura del índice invertido.

2.3. Función `create_index_tfidf()`

Esta función toma una colección de artículos de Wikipedia y crea un índice invertido. Este índice invertido se implementa como un diccionario de Python, donde las claves son términos y los valores son listas de tuplas que contienen información sobre los documentos y las posiciones en las que aparecen esos términos. El índice invertido se utiliza para mejorar la eficiencia y la relevancia de las búsquedas en el conjunto de artículos de Wikipedia.

2.4. Función `rank_document()`

Esta función toma una lista de términos de consulta, una lista de documentos a clasificar, un índice invertido, las frecuencias inversas de documentos (idf) y las frecuencias de términos (tf). Luego, clasifica los documentos en función de los pesos tf-idf y devuelve una lista de documentos clasificados. Esta función es útil para mejorar la relevancia de los resultados de búsqueda basados en la similitud de coseno entre la consulta y los documentos en un conjunto de datos.

2.5. Función `search_tf_idf()`

Esta función toma una consulta de términos y un índice invertido como entrada. Luego, busca los documentos que contienen al menos uno de los términos de la consulta en el índice invertido y devuelve una lista de documentos clasificados según su relevancia en función de los pesos tf-idf. Esta función es útil para mejorar la eficiencia y la relevancia de las consultas de búsqueda en un conjunto de datos al utilizar el índice invertido y los pesos tf-idf.

2.6. Función `precision_at_k()`

Definición: Precisión es la proporción de documentos relevantes recuperados con respecto al total de documentos recuperados. `Precision@k` se refiere a la precisión en los primeros k resultados.

Interpretación: `Precision@k` evalúa cuántos de los documentos recuperados en los primeros k resultados son relevantes. Es útil para entender qué tan efectivo es el sistema en la parte inicial de la lista de resultados.

2.7. Función `recall_at_k()`

Definición: Recall es la proporción de documentos relevantes recuperados con respecto al total de documentos relevantes en la colección. `Recall@k` se refiere al recall en los primeros k resultados.

Interpretación: `Recall@k` evalúa cuántos de los documentos relevantes se han recuperado en los primeros k resultados. Es útil para entender qué tan completo es el sistema en la parte inicial de la lista de resultados.

2.8. Función `avg_precision_at_k()`

Definición: AvgPrecision@k es el promedio de las precisiones calculadas para cada consulta en los primeros k resultados. Es especialmente útil cuando se tienen múltiples consultas para obtener una medida global del rendimiento del sistema.

Interpretación: AvgPrecision@k te da una idea de la calidad promedio de los resultados en las primeras k posiciones para todas las consultas.

2.9. Función `f1_score_at_k()`

Definición: El F1-Score es la media armónica de precisión y recall. F1-Score@k se refiere al F1-Score calculado en los primeros k resultados.

Interpretación: F1-Score@k proporciona un balance entre precisión y recall en los primeros k resultados, lo que es útil para evaluar sistemas cuando ambas métricas son importantes.

2.10. Función `map_at_k()`

Definición: MAP@k es el promedio de los AvgPrecision@k para todas las consultas.

Interpretación: MAP@k proporciona una medida global del rendimiento del sistema considerando todas las consultas.

2.11. Función `rr_at_k()`

Definición: RR@k es el recíproco del rango en el que se recupera el primer documento relevante. En otras palabras, es la inversa de la posición del primer resultado relevante.

Interpretación: RR@k se centra en la posición del primer documento relevante recuperado. Un RR@k alto indica que los documentos relevantes tienden a aparecer más cerca del comienzo de la lista de resultados.

2.12. Función `mrr_at_k()`

Definición: MRR@k es el promedio de los inversos de los rangos en los que se encuentra el primer documento relevante.

Interpretación: MRR@k se centra en el primer documento relevante recuperado. Un MRR alto indica que los documentos relevantes tienden a aparecer más cerca del inicio de la lista de resultados.

2.13. Función `dcg_at_k()`

Definición: DCG@k mide la calidad de una lista clasificada de resultados de búsqueda. Considera tanto la relevancia como la posición de los documentos en la lista. Se calcula como la suma de las ganancias de cada resultado, descontadas en función de su posición.

Interpretación: DCG@k tiene en cuenta tanto la relevancia de los documentos como sus posiciones. Asigna puntuaciones más altas a los documentos relevantes que aparecen más arriba en la lista. Esta métrica es particularmente útil cuando se califica la relevancia (por ejemplo, con diferentes niveles de relevancia).

2.14. Función `ndcg_at_k()`

Definición: NDCG@k es una métrica que considera la relevancia de los documentos recuperados y la posición en la que aparecen.

Interpretación: NDCG@k proporciona una medida de calidad que toma en cuenta tanto la relevancia como la posición de los documentos en la lista de resultados.

3. Indexing

Primero de todo, después del Pre-Procesamiento de Datos (parte 1 del proyecto), pasamos a la fase de Indexación (parte 2 del proyecto).

3.1. Índice invertido CON información de posición del término

Esto, primeramente, implica construir el índice invertido CON información de posición del término para nuestros datos (utilizando la función `create_index()`). Antes pero, de poder utilizar esta función, tendremos que adaptar nuestros datos al formato de nuestra función, es decir, necesitaremos un input de la siguiente forma:

```
'ID' | 'doc_xxx' | 'PrePreocessed_Tweet'
```

```
1575918081461080065|doc_2|the arm forc liber villag urban territori  
commun region drobysheve lymansk donetsk ukraine russia war ukraine war  
ukraine ukraine will win ukrainian army ukrainecounteroffensive ukraine  
war news slava ukra stand with ukraine
```

Una vez adaptado el formato y llamado a la función, observamos que se tarda, aproximadamente, 3.84 segundos en llevar a cabo la operación.

Seguidamente, llamando a la función `search()` con los parámetros adecuados, podemos probar de insertar cualquier query que deseemos (siempre imprimiendo el top10).

Con esta manera de crear el índice invertido, observaremos que no será lo mismo insertar “annex” que “annexation”, algo parecido ocurría en el laboratorio que llevamos a cabo en clase. Esto sucede porque estamos almacenando en el índice términos derivados.

Ejemplo

```

Insert your query (i.e.: presidents visiting Kyiv):

presidents visiting Kyiv

=====
Sample of 10 results out of 206 for the searched query:
=====

tweet_id = 1575910966206038016 - tweet_title: doc_58
tweet_text: As Europe prepares to defend Ukraine French leaders are donning masks in their traditional
battle flag colors #UkraineRussiaWar #UkraineWillWin #BidenWorstPresidentInHistory https://t.co/EWMecu3DAP

tweet_id = 1575913989195718657 - tweet_title: doc_30
tweet_text: Former Russian Prime Minister Mikhail Kasyanov believes that Russian President Vladimir #Putin
could step down from his position and flee #Russia in a few months.
#Ukraine #UkraineRussiaWar #NATO #Putler https://t.co/VzzStxdkTj

tweet_id = 1575905170952982529 - tweet_title: doc_131
tweet_text: 🚨BREAKING: After Putin's unilateral declaration of the annexation of occupied Ukraine,
President Volodymyr Zelensky has announced that Kyiv has formally requested to join NATO.
https://t.co/aRQwskpZlL

#WARINUKRAINE #UKRAINEWAR #UKRAINERUSSIARWAR #UKRAINE https://t.co/sz0rkTLlrl

tweet_id = 1575908651658641408 - tweet_title: doc_87
tweet_text: 🚨"#Ukraine is ready to talk with Russia, but only with a different Russian president."
Zelensky words after Putin's speech formalizing the annexation, in which he asked Kyiv for negotiations to
end the war
https://t.co/aRQwskH2nL

#WARINUKRAINE #UKRAINEWAR #UKRAINERUSSIARWAR https://t.co/DLwKgztFqG

tweet_id = 1575901742398861312 - tweet_title: doc_156
tweet_text: The United States on Friday announced "severe" new sanctions on Russia in response to what
President Joe Biden called Moscow's "fraudulent" claim to have annexed four Ukrainian regions.
#RUKIGAFMUpdates #UkraineRussiaWar #Russia #Ukraine https://t.co/tdKwUCG4H

tweet_id = 1575897160633430018 - tweet_title: doc_203
tweet_text: STOP :octagonal_sign: sending our tax dollars to the #UkraineRussiaWar. #BidenIsADisgrace
#BidenWorstPresidentInHistory https://t.co/zryQtYgBgC

tweet_id = 1575904025756696582 - tweet_title: doc_143
tweet_text: #UkraineWar #Ukraine #Russia #ukrainerussiawar #Putin #SanktionengegenueUSA #MAGA #俄罗斯 #乌克兰 #中國

Maria Zakharova:

There is no need to speculate. We must admit what the President of the United States spoke about when he
promised to put an end to Nord Stream 2. https://t.co/CJvjLtfbnp

tweet_id = 1575892959572418560 - tweet_title: doc_233
tweet_text: Zelenskyy, President of Ukraine, submits "accelerated" NATO application. This comes after
Russia annexed four Ukrainian territories a few hours ago. Well, shit #annexation #Russia #UkraineWar
#Ukraine #UkraineRussiaWar #Zelensky https://t.co/WUPJRaHNZk

tweet_id = 1575890966489071619 - tweet_title: doc_247
tweet_text: The Russian President, Sir Vladimir Putin announces Russian annexation of four Ukraine
regions.
The move has been condemned by Ukraine and Western countries and represents a major escalation in the 7-
month war #TrendingNow #Russia #Putin #Ukraine #UkraineWar #UkraineRussiaWar https://t.co/DjqCyWmZT5

tweet_id = 1575887266362249216 - tweet_title: doc_281
tweet_text: The President of Russian Federation Vladimir Putin just signed decrees recognizing Kherson and
Zaporozhye regions as independent territories.

#UkraineRussiaWar #Ukraine #UkraineUnderAttack #UkraineWar #UkraineWarCrimes #Ukrainian
#UkraineRussiaConflict #StopPutinNOW #StopRussia https://t.co/cKwgUkdRM9

```

3.2. Índice invertido SIN información de posición del término

A continuación, construimos el índice invertido, esta vez, SIN información de posición del término para nuestros datos (utilizando la función `create_index_tfidf()`). Igual que antes, adaptamos nuestros tweets al formato especificado.

Una vez adaptado el formato y llamado a la función, observamos que, a diferencia que la creación del índice anterior, ahora se tarda, aproximadamente, 304.12 segundos en llevar a cabo la operación. Es evidente que este proceso requiere mucho tiempo, es por eso que, en proyectos futuros, podría resultar beneficioso explorar métodos de cálculo más eficientes.

No obstante, podremos observar en apartados siguientes que obtenemos un resultado más preciso para futuras técnicas de evaluación.

3.3. Queries escogidas

Posteriormente, formulamos cinco queries y mostramos los diez documentos mejor clasificados para cada query. En esta sección, nos centramos en proporcionar detalles esenciales para los diez documentos mejor clasificados, que incluyen:

- tweet_id: ID del tweet.
- tweet_title: doc_xxx.
- score: puntuación del documento.
- tweet_text: texto del tweet.

Señalar que las consultas elegidas son aquellas que consideramos particularmente interesantes y útiles para un lector.

Query 1: Eastern Ukraine separatist groups

→ Input: "Eastern separatists groups"

Esta query se refiere a varias facciones armadas, milicias y entidades políticas que surgieron en las regiones orientales de Ucrania, particularmente en áreas como Donetsk y Luhansk, durante el conflicto que comenzó en 2014. Estos grupos buscaban una mayor autonomía o independencia del gobierno ucraniano y , en algunos casos, incluso abogaron por unirse a Rusia.

Query 2: Humanitarian impact of Russia-Ukraine war

→ Input: "Humanitarian impact"

Esta query prueba si el algoritmo de búsqueda puede proporcionar información sobre las consecuencias humanitarias, como el desplazamiento de civiles o los esfuerzos de ayuda humanitaria.

Query 3: Media coverage of Russia-Ukraine war

→ Input: "Media coverage of war"

Esta query comprueba si el algoritmo de búsqueda puede ofrecer información sobre cómo los medios han cubierto la guerra.

Query 4: Negotiation attempts in Russia-Ukraine war

→ Input: "Negotiations i war"

Esta query tiene como objetivo evaluar si el algoritmo de búsqueda puede proporcionar información sobre algún esfuerzo diplomático o negociación que haya tenido lugar.

Query 5: Russian propaganda and disinformation in the Ukraine conflict

→ Input: Russian propaganda and disinformation

Esta query intenta buscar información referente a la difusión deliberada de información engañosa o falsa por parte de fuentes rusas con el objetivo de influir en la opinión pública, dar forma a narrativas y promover sus intereses geopolíticos en el contexto del conflicto entre Rusia y Ucrania.

3.4. Implementación algoritmo TF-IDF y “ranking based results”

Query 1: Eastern Ukraine separatist groups

```
Insert your query (i.e.: standwithukrain):

Eastern separatists groups

=====
Top 10 results out of 55 for the searched query:

tweet_id = 1575842840768569344 - tweet_title: doc_538 - score: 9.068672454039067
tweet_text: The spokesman of the Eastern group of troops Cherevaty reported that the encirclement of the
Russian group near Lyman in Donetsk region is "at the stage of completion"
#UkraineRussiaWar https://t.co/drAsg9PDes

tweet_id = 1575818569857658880 - tweet_title: doc_812 - score: 7.9992014749313505
tweet_text: First Official APU Report on Lyman: "The operation to encircle the Russian group in the
Estuary is at the completion stage" - Sergey Cherevaty, Eastern Grouping

#UkraineRussiaWar
#OSINT
#Fellas #NAFO

tweet_id = 1575821202064834560 - tweet_title: doc_776 - score: 7.348035242319465
tweet_text: Cherevaty, the spokesperson of the Eastern group of troops, reported the encirclement of the
#Russian group near #Lyman in the #Donetsk region is "at the stage of completion." #Ukraine
#UkraineRussiaWar #UkraineWar

📎 Tpyxa https://t.co/jUcJncxRJ6

tweet_id = 157582023701006017 - tweet_title: doc_790 - score: 7.255889456636643
tweet_text: Update: Addition comments from APU Eastern Grouping

#UkraineRussiaWar
#OSINT
#Fellas #NAFO https://t.co/svKSKcy403

tweet_id = 1575180675002486785 - tweet_title: doc_3778 - score: 6.671130033545804
tweet_text: @nytimes Wherever these 200k draft dodgers have gone, they're many enough to be the future
Russian separatists beloved by Putin #UkraineRussiaWar #RussianArmy #Russians

tweet_id = 1575818037130731520 - tweet_title: doc_820 - score: 6.655307724400341
tweet_text: 🇷🇺 Official comment on the situation in Lyman from the military

💬 "The operation to encircle the Russian group in Lyman is "at the stage of completion", - the
representative of the Eastern group Serhiy Chereviy.
#UkraineWillWin
#UkraineWar
#UkraineRussiaWar
#Russian https://t.co/vuKHUp0foF
```

```
tweet_id = 1575785557896007682 - tweet_title: doc_1183 - score: 5.720494003765528
tweet_text: The head of the Russian-backed separatist administration in east Ukraine's Donetsk region said the Russian stronghold of Lyman was "semi-encircled" by the Ukrainian army and that news from the front was "alarming."

#Russia | #Donetsk | #UkraineRussiaWar
https://t.co/sGp1vw7334

tweet_id = 1575488992929513473 - tweet_title: doc_2382 - score: 4.629911409155417
tweet_text: Ukrainian Forces at the Eastern front in action. https://t.co/ocDykq9rI5 lewat @YouTube #war #ukraïne #russia #ukrainerussiawar #nowar

tweet_id = 1575353426564857857 - tweet_title: doc_2977 - score: 3.6605877681494445
tweet_text: Russian attack hits a school in eastern Ukraine's town of Mykolaivka being used by residents as a shelter.

#UkraineRussiaWar
https://t.co/s5ji7BpJ8h

tweet_id = 1575822314586808320 - tweet_title: doc_745 - score: 3.3648667793298763
tweet_text: Reports of #Russian battle groups and bombers en route to #Lyman are just to cover the retreat, I think. The settlement is gone. #UkraineRussiaWar
```

Query 2: Humanitarian impact of Russia-Ukraine war

```
Insert your query (i.e.: standwithukrain):

Humanitarian impact

=====
Top 10 results out of 30 for the searched query:

tweet_id = 1575722456823738370 - tweet_title: doc_1426 - score: 7.143300400638406
tweet_text: 🇷🇺 Russian missile attack on the humanitarian convoy from Zaporizhzhia to occupied territories. People were heading to rescue their relatives left in russia-controlled areas and provide humanitarian aid.

#UkraineRussiaWar https://t.co/ZlYntJRLiA

tweet_id = 1575684580295970817 - tweet_title: doc_1540 - score: 6.736325307562501
tweet_text: Here's how the war in Ukraine 🇺🇦 is impacting world trade and investment acc to @wef

#SupplyChain #Procurement #economy #UkraineRussiaWar #foodcrisis #logistics https://t.co/NNCRHhTFsW

tweet_id = 1575417601781530626 - tweet_title: doc_2753 - score: 5.339770060872714
tweet_text: India and China opting to make positive impact in Ukraine

https://t.co/Pi8PpSrMg3

#TheIsland #TheIslandnewspaper #TheIslandOnline #features #featurestory #India #China #positiveimpact #Ukraine #UkraineRussiaWar #RussianUkrainianWar https://t.co/e6BQbgAFNK

tweet_id = 1575708323932164098 - tweet_title: doc_1458 - score: 4.995370690082699
tweet_text: 2/2 The rockets destroyed the transport company. During the fire, which was caused by the impact 52 buses burned down, almost a hundred townspeople were injured.
#RussiaIsATerroristState #Ukrainian #UkraineWillWin #UkrainianArmy #ukrainecounteroffensive #Ukraine #UkraineRussiaWar

tweet_id = 1575360480406814721 - tweet_title: doc_2951 - score: 4.935337772238568
tweet_text: The US has seized upon Russia's invasion of #Ukraine to escalate the war with #Russia and impose the cutoff of EU energy trade with Russia that it had long sought. The impact on Europe is devastating.

#Nordstream
#UkraineRussiaWar https://t.co/mhWBxMslgh
```



```
tweet_id = 1575822715973402624 - tweet_title: doc_740 - score: 4.8684451736120495
tweet_text: @EliotHiggins @bellingcat Is there a look into the terrible incident with the humanitarian
convoy in Zaporizhzhia region? Both sides once again blaming each other. #UkraineRussiaWar

tweet_id = 1575397849013096448 - tweet_title: doc_2811 - score: 4.657290573802592
tweet_text: #UkraineRussiaWar #Ukraine #Russia

:oil:#Nordstream 1&2 will forever remain unusable if urgent repairs aren't done. Russia does not have
the corresponding equipment. The cost of infrastructure is 17 billion EUR. This will impact the Russia
economy for a long time, less the EU - RIP https://t.co/65n1vvNsBA

tweet_id = 1575742923068813314 - tweet_title: doc_1388 - score: 4.072519045341019
tweet_text: :projector:Russians shelled the outskirts of #Zaporizhzhia and hit a civilian humanitarian
convoy heading towards the occupied parts. 23 people were killed, a dozen more wounded.
#UkraineRussiaWar https://t.co/365j43jy51

tweet_id = 1575743037996572672 - tweet_title: doc_1386 - score: 3.974166617591281
tweet_text: #NewsAlert | 23 killed after Russian attack on humanitarian convoy in Ukraine: Governor |
reported by news agency AFP

#UkraineRussiaWar

tweet_id = 1575804828562890753 - tweet_title: doc_981 - score: 3.7974965158926763
tweet_text: In #Izium - prized by the Russians as a key tactical position and logistics hub - the world is
witnessing even more appalling scenes of murder and the aftermath of the catastrophic humanitarian
situation.

#UkraineRussiaWar #Kharkiv

https://t.co/a@momMID6c
```

Query 3: Media coverage of Russia-Ukraine war

```
Insert your query (i.e.: standwithukrain):

Media coverage of war

=====
Top 10 results out of 3929 for the searched query:

tweet_id = 1575813822786551808 - tweet_title: doc_883 - score: 4.870877111750377
tweet_text: Ukraine is not better than Russia.
Today they killed 23 people in Rocket attack on a civil convoy.
Although I am against the invasion but I am also against western biased coverage that turns back on
ukraine war crimes.
#UkraineRussiaWar
#poundcrash

tweet_id = 1575242122684293120 - tweet_title: doc_3355 - score: 4.698775773590212
tweet_text: You might think it's conspiracist nonsense to accuse Biden of sabotaging #NordStream2 which
media all blame on Russia, but look in the comments at the video of him threatening to do just that: shows
nothing western media claims on #UkraineRussiaWar can be trusted #NordstreamLeaks https://t.co/lH3kdGaMEI

tweet_id = 1575810496116117504 - tweet_title: doc_926 - score: 4.592619972554513
tweet_text: Coming soon on SGN, live coverage as Putin holds treaty ceremony to officially annex areas of
Ukraine. #UkraineRussiaWar #UkraineWar #Ukraine #Putin

https://t.co/Vi4U6UPcA8

tweet_id = 1575819110251757569 - tweet_title: doc_802 - score: 4.303342768519564
tweet_text: SGN #live coverage of treaty ceremony from Moscow as Russia officially annexes areas of
Ukraine is now live with our interactive chat and English translations #UkraineRussiaWar #UkraineWar
#Ukraine #Putin

https://t.co/Vi4U6UPcA8

tweet_id = 1575816865401954304 - tweet_title: doc_837 - score: 3.821214114252134
tweet_text: #LIVE:
#Putin Annexes Liberated Territories -BREAKING NEWS COVERAGE (Ceremony & Speech)..
https://t.co/Ud4u6RQJSY via @lookner #Kharkiv #Lyman #Kiev #Kyiv #Ukraine #Kherson #Zaporozhye #Odessa
#Mykolayiv #Kharkiv #Russia #UkraineRussiaWar #UkraineWar #NATO #WagnerGriup #LVIV #DPR
```

```
tweet_id = 1575763015894372355 - tweet_title: doc_1271 - score: 3.7580116496514258
tweet_text: Europe blames #Russia for war. This is one more propaganda propagated using media, social
media & inflow of US dollars. Why did you keep expanding @NATO? Why didn't you reach a breakthrough
agreement in 3 decades?

3/
@Europarl_EN
@EU_Commission

#UkraineRussiaWar
#UkraineWar

tweet_id = 1575795770195726336 - tweet_title: doc_1083 - score: 3.6862267502924295
tweet_text: #UkraineRussiaWar #Ukraine #Russia

:globe_with_meridians: Social media
#Dagestan mobilization https://t.co/HI1zjMeK9W

tweet_id = 1575755162936696832 - tweet_title: doc_1331 - score: 3.552013014971761
tweet_text: #UkraineRussiaWar #Ukraine #Russia

:globe_with_meridians: Social media
Reservists before their departure for the front https://t.co/pvwLKSojq5

tweet_id = 1575749242412761089 - tweet_title: doc_1356 - score: 3.3221221614026946
tweet_text: #UkraineRussiaWar #Ukraine #Russia

:globe_with_meridians: Social media
Russians forces reportedly flees from #Lyman https://t.co/q59J0GhM9E

tweet_id = 1575524817431838720 - tweet_title: doc_2257 - score: 3.2224584965606136
tweet_text: I wouldn't be surprised if i see this as a headline in #western propagandists media.
#UkraineRussiaWar #NordStream2 #Nordstream #EnergyCrisis #EU https://t.co/MWL5QwK1xF
```

Query 4: Negotiation attempts in Russia-Ukraine war

```
Insert your query (i.e.: standwithukrain):

Negotiations in war

=====
Top 10 results out of 3928 for the searched query:

tweet_id = 1575324772199710720 - tweet_title: doc_3067 - score: 6.153608314723627
tweet_text: Noam Chomsky: "Most of the world... is calling for #Negotiations now, while the US insists
that priority must be to severely weaken #Russia, hence no negotiations." (1/2)

#Ukraine #UkraineRussiaWar
https://t.co/QB1QLH5FdZ

tweet_id = 1575776384324374528 - tweet_title: doc_1218 - score: 4.730562863510461
tweet_text: #EXCLUSIVE : Dmitry Peskov on the possibility of negotiations with Volodymyr Zelensky: #Kyiv
has left the negotiation track, #Moscow's demands do not change, the Special Military Operation (SVO) will
continue...

#RussianArmy #UkraineRussiaWar #UkraineWar #UkrainianArmy

tweet_id = 1575845160134451200 - tweet_title: doc_515 - score: 4.120218268516906
tweet_text: 3o/9
🇺🇦🇷🇺
negotiating table".
But we already know that negotiations with #Russia will not last more than one round of machine gun fire

#Ukraine #Ukrainian #UkraineKrieg #UkraineWar #UkraineRussiaWar #stopPutin #UkraineCounterOffensive
#Donbass #Kherson #NATO #Europe #USA https://t.co/Eeu5QkuP3u

tweet_id = 1575825005291524098 - tweet_title: doc_699 - score: 3.804847410949541
tweet_text: 🇷🇺 PUTIN: We are ready for negotiations. #KREMLIN #UkraineRussiaWar #PUTIN #Zalwski

tweet_id = 1575871713317105665 - tweet_title: doc_349 - score: 3.6981547825187917
tweet_text: Breaking: Pres. Zelenskyy just announced the UA will not negotiate with RUS while Putin is in
power.

#UkraineRussiaWar
#OSINT
#Fellas #NAFO

tweet_id = 1575653403417628672 - tweet_title: doc_1657 - score: 3.6981547825187917
tweet_text: Ukraine does not stop working on exchanging prisoners with the Russian Federation;
negotiations on the exchange of "all for all" are ongoing 🇺🇦
#Ukraine #UkraineRussiaWar
https://t.co/iQbCElFWmj
```

```
tweet_id = 1575247684621012993 - tweet_title: doc_3305 - score: 3.599360930834501
tweet_text: That awkward moment when your US masters are telling you to negotiate. But the war is going so
well for Ukraine we're told. #UkraineRussiaWar https://t.co/Wec2vFPwYu

tweet_id = 1575304304004562944 - tweet_title: doc_3123 - score: 3.508357036332147
tweet_text: Defcon 2 is coming.
Beware of nuclear war.
Demand peace negotiations.
#UkraineRussiaWar #NordStream2 #Russia #US https://t.co/YUeV3V2V3z

tweet_id = 1575860740556460032 - tweet_title: doc_417 - score: 3.2713842687957917
tweet_text: :flag_ua::flag_ru:Ukraine will not hold any negotiations with Russia while Putin is president,
Zelensky stated (September 30, 2022).

#Ukraine #UkraineWar #UkraineRussiaWar #Zelensky

tweet_id = 1575861001769013248 - tweet_title: doc_415 - score: 3.20234786216413
tweet_text: :flag_ua::flag_ru:Lavrov: we need to take Putin's phrase seriously that the longer Kyiv
refuses to negotiate, the more difficult it will be to agree (September 30, 2022).

#Ukraine #UkraineWar #UkraineRussiaWar #Lavrov #Zelensky
```

Query 5: Russian propaganda and disinformation in the Ukraine conflict

```
Insert your query (i.e.: standwithukrain):

Russian propaganda and disinformation

=====
Top 10 results out of 1509 for the searched query:

tweet_id = 1575413884617383936 - tweet_title: doc_2762 - score: 19.435317936601205
tweet_text: Fake news - A typical British propaganda and disinformation

#Ukraine #UkraineWar #UkraineRussiaWar #Propaganda #Fake #Disinformation https://t.co/LFVEGyLhfK

tweet_id = 1575245412423897105 - tweet_title: doc_3324 - score: 7.953635177621761
tweet_text: @RebelNewsOnline How about a concerted effort of disinformation to the benefit of #Russiangas?
#ONGT #natgas #UkraineRussiaWar https://t.co/OrC5gFvxYo

tweet_id = 1575793780350889984 - tweet_title: doc_1098 - score: 7.240923415464685
tweet_text: #UkraineRussiaWar #Kharkiv #Izium #warehouse #russians #propaganda #referendum russian
warehouse with ammunition, propaganda leaflets found in Izium https://t.co/GNj3TUoysk

tweet_id = 1575228605088468992 - tweet_title: doc_3439 - score: 7.028208537130871
tweet_text: US State Department: The claim that Washington is behind the Nord Stream incident is
unreasonable and part of the Russian disinformation.

#UkraineRussiaWar

tweet_id = 1575642353905573889 - tweet_title: doc_1717 - score: 6.239974890355639
tweet_text: @OrinocoTribune @raymcgovern Your source is a Venezuelan propaganda site? And they have no
hatred for USA? think again, just more friends of #RussiaIsANaziState spewing propaganda filth.

#UkraineRussiaWar
#Ukraine https://t.co/VZ8WIXVY5x

tweet_id = 1575784366957301760 - tweet_title: doc_1187 - score: 4.3032830437981255
tweet_text: #DOPPELGANGER: How Russia-based actors cloned legitimate media outlets from multiple countries
(🇬🇧🇫🇷🇮🇹🇪🇸) to spread #disinformation designed to undermine the support for #Ukraine. #UkraineRussiaWar

@DisinfoEU last investigation: https://t.co/lydkTW0gW6
```

```
tweet_id = 1575533133901930496 - tweet_title: doc_2217 - score: 3.4716025524620107
tweet_text: #American #propaganda and Nord Stream 2 exposed in 4 minutes :point_down::point_down:
https://t.co/Dc03Tcaqw0
#Ukraine #UkraineRussiaWar #NordStream2 #NATO #EU

tweet_id = 1575453048075436047 - tweet_title: doc_2627 - score: 3.3621786042315183
tweet_text: @RichJones89 @AnonOpsSE C'mon, Jones. If you are a defective person, do not think that everyone
around you is the same as you.

Here is a couple of examples of 'Russian' propaganda:
1. https://t.co/W5CEOVNJIL
2. https://t.co/JFak01uTfD
3. https://t.co/h4AHjemP3K

#Ukraine #UkraineWar #UkraineRussiaWar

tweet_id = 1575673820085510144 - tweet_title: doc_1580 - score: 3.3172737497173657
tweet_text: I'll have to read this instalment on the Ukraine- bribery, militias and the dirty war waged in
the name of democracy.

We get good guy - bad guy propaganda but the truth is all the more shocking. #ukrainerrussiawar
#auspol

tweet_id = 1575472762554048512 - tweet_title: doc_2470 - score: 3.3172737497173657
tweet_text: Television news outlets in the Propaganda State have actually started a countdown clock for
Putin's signing of the annexation of occupied territories.

#UkraineRussiaWar
#OSINT
#Fellas #NAFO https://t.co/rqUV8f7c2P
```

4. Evaluation

En la parte final de la práctica realizamos la Evaluación (parte 2). Esta sección se divide en dos partes, ya que evaluamos el sistema utilizando tanto el marco de datos proporcionado para las consultas dadas como nuestras propias consultas personalizadas.

4.1. Queries Proporcionadas

Inicialmente, descargamos el archivo csv que contiene una baseline con tres queries, y los documentos que contienen los tweets originales correspondientes para cada query. En este caso, las queries propuestas son:

Query 1: What is the discussion regarding a tank in Kharkiv? → Input: tank in Kharkiv



Query 2: What discussions are there about the Nord Stream pipeline → Input: Nord Stream pipeline

Query 3: What is being said about the annexation of territories by Russia? → Input: annex territories Russia

Así pues, posteriormente, generamos un dataframe con los documentos resultantes de las queries proporcionadas que contendrá la siguiente información (o columnas):

- tweet_id: ID del tweet.
- query_id: 1, 2 or 3.
- score: puntuación del documento al realizar la query.
- ground_truth: 0 = no-relevante, 1 = relevante.

```
queries_df = queries_df.sort_values(by=['query_id', 'score'], ascending
= [True, False])
queries_df.head()
```

	tweet_id	query_id	score	ground_truth	
0	1575528927245770752	1	4.698085	1	
1	1575448045457707008	1	3.713707	1	
6	1575435463682363392	1	3.266190	1	
10	1575753840233701376	1	2.870063	0	
3	1575834054905462784	1	2.164526	1	

Una vez el dataframe creado, procedemos a evaluar los resultados para cada una de las funciones descritas anteriormente (“evaluation techniques”):

Precision@k

****k = 4****

Query1 = 0.75

Query2 = 1.0

Query3 = 1.0

****k = 8****

Query1 = 0.875

Query2 = 1.0

Query3 = 1.0

****k = 12****

Query1 = 0.75

Query2 = 0.833333334

Query3 = 0.833333334

****k = 16****

Query1 = 0.625

Query2 = 0.625

Query3 = 0.625

****k = 20****

Query1 = 0.5

Query2 = 0.5

Query3 = 0.5

Recall@k

****k = 4****

Query1 = 0.3

Query2 = 0.4

Query3 = 0.4

****k = 8****

Query1 = 0.7

Query2 = 0.8

Query3 = 0.8

****k = 12****

Query1 = 0.9

Query2 = 1.0

Query3 = 1.0

****k = 16 & k = 20****

Query1 = 1.0

Query2 = 1.0

Query3 = 1.0

AvgPrecision@k vs Python predefined function

****k = 20****

Query1 - 0.8684706959706959 vs 0.8708516483516483

Query2 - 1.0 vs 0.9999999999999999

Query3 - 1.0 vs 1.0

F1-Score@k

****k = 4****

Query1 =

0.4285714285714285

Query2 =

0.5714285714285715

Query3 =

0.5714285714285715

****k = 8****

Query1 = 0.777777777

Query2 = 0.888888889

Query3 = 0.888888889

****k = 12****

Query1 = 0.81818182

Query2 = 0.90909091

Query3 = 0.90909091

****k = 16****

Query1 =

0.7692307692307693

Query2 =

0.7692307692307693

Query3 =

0.7692307692307693

****k = 20****

Query1 = 0.666666666

Query2 = 0.666666666

Query3 = 0.666666666

MAP@k

****k = 4**** → 0.3666666666666667

****k = 8**** → 0.7455158730158731

****k = 12**** → 0.930515873015873

****k = 16 & k = 20**** → 0.9561568986568987

MRR@k

****k = 4 & k = 8 & k = 12 & k = 16 & k = 20**** → 1.0

NDCG@k****k = 4****

Query1 = 0.8319

Query2 = 1.0

Query3 = 1.0

****k = 8****

Query1 = 0.8911

Query2 = 1.0

Query3 = 1.0

****k = 12****

Query1 = 0.8984

Query2 = 1.0

Query3 = 1.0

****k = 16 & k = 20****

Query1 = 0.9562

Query2 = 1.0

Query3 = 1.0

Como podemos observar, y hemos mencionado anteriormente, estos datos representan las métricas de evaluación del algoritmo de “search” implementado previamente, en particular, se evalúan los resultados para diferentes valores de k (k = 4, 8, 12, 16, 20) y para las tres consultas definidas más arriba (Query1, Query2, Query3). Cada métrica evalúa aspectos diferentes del algoritmo de recomendación.

Como conclusión general, podemos observar que los resultados sugieren que a medida que k aumenta, el modelo tiende a realizar recomendaciones de mayor calidad, con una mejora en la precisión, el recall, el F1-Score, el MAP@k y el NDCG@k. Además, el MRR@k indica que el modelo es eficaz para encontrar la primera recomendación relevante.

4.2. Nuestras Queries

En el siguiente apartado, seguiremos un marco similar. Sin embargo, esta vez emplearemos:

- Las cinco consultas personalizadas de la sección de indexación.
- Los documentos mejor clasificados, con un valor de k = 20.
- Serviremos como jueces para evaluar la relevancia o no de estos documentos, para así, tal y como hemos realizado arriba, construir un conjunto de datos para su posterior evaluación.

```
new_queries_df = new_queries_df.sort_values(by=['query_id','score'],
ascending = [True, False])
new_queries_df.head()
```

	tweet_id	query_id	score	ground_truth
0	1575842840768569344	1	3.282387	1
1	1575818569857658880	1	3.129791	1
2	1575821202064834560	1	2.879615	1
3	1575820237010006017	1	2.879615	1
4	1575180675002486785	1	2.725980	0

Una vez generado el dataframe, procedemos a evaluar los resultados de cada una de las funciones antes mencionadas:

Precision@k

****k = 4****

Query1 = 1.0

Query2 = 0.5

Query3 = 0.5

Query4 = 0.5

Query5 = 0.75

****k = 8****

Query1 = 0.75

Query2 = 0.625

Query3 = 0.625

Query4 = 0.5

Query5 = 0.625

****k = 12****

Query1 = 0.5

Query2 = 0.6666666666

Query3 = 0.75

Query4 = 0.5833333334

Query5 = 0.5

****k = 16****

Query1 = 0.4375

Query2 = 0.5625

Query3 = 0.625

Query4 = 0.5

Query5 = 0.4375

****k = 20****

Query1 = Query2 = Query3 = Query4 = Query5 = 0.5

Recall@k

****k = 4****

Query1 = 0.4

Query2 = 0.2

Query3 = 0.2

Query4 = 0.2

Query5 = 0.3

****k = 8****

Query1 = 0.6

Query2 = 0.5

Query3 = 0.5

Query4 = 0.4

Query5 = 0.5

****k = 12****

Query1 = 0.6	Query2 = 0.8	Query3 = 0.9
Query4 = 0.7		Query5 = 0.6

****k = 16****

Query1 = 0.7	Query2 = 0.9	Query3 = 1.0
Query4 = 0.8		Query5 = 0.7

****k = 20****

Query1 = Query2 = Query3 = Query4 = Query5 = 1.0

AvgPrecision@k vs Python predefined function

****k = 20****

Query1 -	0.7575271512113617	vs	0.7604511278195488
Query2 -	0.670554298642534	vs	0.6705542986425339
Query3 -	0.6098701298701299	vs	0.6265367965367965
Query4 -	0.5729698028150041	vs	0.5729698028150041
Query5 -	0.6145588235294117	vs	0.6202605779153767

F1-Score@k

****k = 4****

Query1 =	Query2 =	Query3 =
0.5714285714285715	0.28571428571428575	0.28571428571428575
Query4 = 0.28571428571428575	Query5 = 0.4285714285714285	

****k = 8****

Query1 =	Query2 =	Query3 =
0.6666666666666665	0.5555555555555556	0.5555555555555556
Query4 = 0.4444444444444445	Query5 = 0.5555555555555556	

****k = 12****

Query1 =	Query2 =	Query3 =
0.5454545454545454	0.7272727272727272	0.8181818181818182
Query4 = 0.6363636363636365	Query5 = 0.5454545454545454	

****k = 16****

Query1 = 0.5384615384615384 Query2 = 0.6923076923076923 Query3 = 0.7692307692307693

Query4 = 0.6153846153846154 Query5 = 0.5384615384615384

****k = 20****

Query1 = Query2 = Query3 = Query4 = Query5 = 0.6666666666666666

MAP@k

****k = 4**** → 0.195
****k = 8**** → 0.3537380952380952
****k = 12**** → 0.49712193362193363
****k = 16**** → 0.5544424464424464
****k = 20**** → 0.6450960412136884

MRR@k

****k = 4 & k = 8 & k = 12 & k = 16 & k = 20**** → 0.7667

NDCG@k

****k = 4****

Query1 = 1.0 Query2 = 0.5585 Query3 = 0.3633
 Query4 = 0.4144 Query5 = 0.7537

****k = 8****

Query1 = 0.8224 Query2 = 0.6296 Query3 = 0.54974
 Query4 = 0.4565 Query5 = 0.6582

****k = 12****

Query1 = 0.7156 Query2 = 0.7372 Query3 = 0.6835
 Query4 = 0.5885 Query5 = 0.6322

****k = 16****

Query1 = 0.7706 Query2 = 0.795 Query3 = 0.7374
 Query4 = 0.6463 Query5 = 0.6873

****k = 20****

Query1 = 0.9234	Query2 = 0.8478	Query3 = 0.7274
Query4 = 0.75		Query5 = 0.842

Como análisis general, podemos observar que:

- Precision@k:

En general, la precisión varía para diferentes valores de k y consultas.

Remarcar que, la Query1 muestra una precisión más alta en comparación con otras consultas en casi todos los valores de k. Esto implica que, los documentos con más score son los más relevantes, ya que, dentro de la función `precision_at_k()` se ordenan descendientemente.

Finalmente, para k = 20, todas las consultas tienen una precisión de 0.5.

- Recall@k:

En este caso, el recall también varía para diferentes valores de k y consultas.

En general, a medida que k aumenta, el recall tiende a aumentar. Es decir, a medida que aumenta k, aumenta el número de documentos relevantes que se han recuperado con respecto al total de documentos relevantes en la colección.

Finalmente, para k = 20, todas las consultas tienen un recall de 1.0.

- AvgPrecision@k vs Python predefined function:

La comparación de AvgPrecision@k entre el modelo y la función predefinida de Python muestra diferencias muy pequeñas en los valores. Esto indica que el resultado que obtenemos de nuestra función es correcto.

Remarcar que, todos los average precision son mayores que 50% todo mostrando que, nuestro algoritmo es suficientemente bueno.

- F1-Score@k:

El F1-Score es una métrica que combina precisión y recall.

En general, a medida que k aumenta, el F1-Score tiende a aumentar.

- MAP@k:

MAP@k tiende a aumentar a medida que k aumenta, lo que indica una mejora en la calidad de las recomendaciones.

- MRR@k:

MRR@k tiene un valor constante de 0.7667 para todos los valores de k evaluados. Por lo tanto, esto implica que los documentos relevantes tienden a presentarse en los primeros lugares de la lista de resultados.

- NDCG@k:

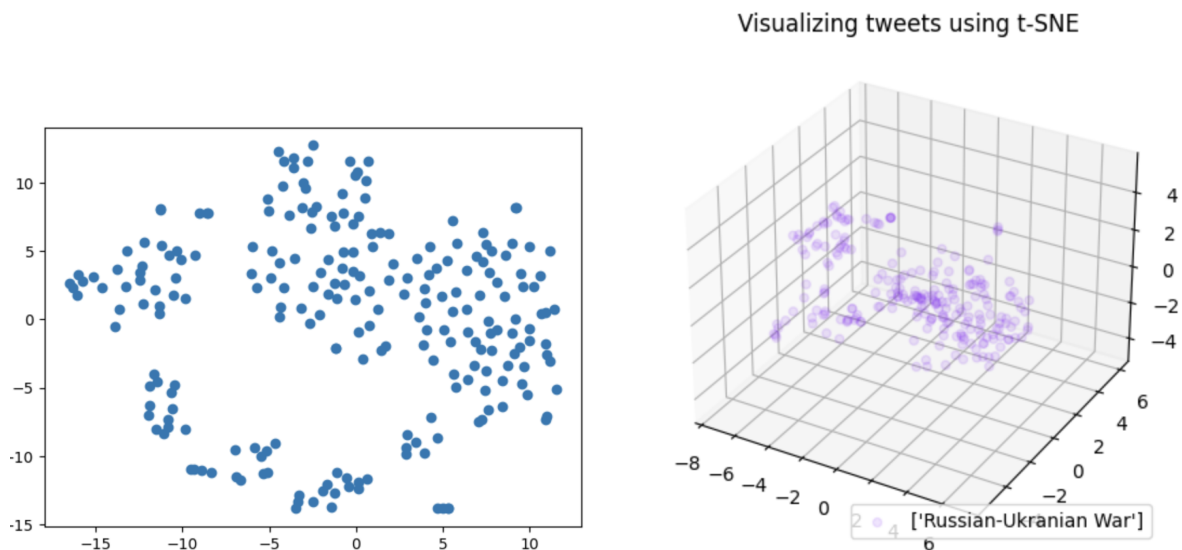
NDCG@k evalúa la calidad de las recomendaciones considerando la relevancia y el orden. En general, a medida que k aumenta, NDCG@k tiende a aumentar, lo que indica que, en general, las recomendaciones son más relevantes y están mejor ordenadas.

En resumen, estos resultados sugieren que, en general, a medida que k aumenta, el modelo tiende a realizar recomendaciones de mayor calidad, con una mejora en la precisión, el recall, el F1-Score, el MAP@k y el NDCG@k. Además, el MRR@k es consistente y muestra que el modelo es eficaz para encontrar la primera recomendación relevante.

4.3. Representación Vectorial

Después de evaluar ambos conjuntos de resultados de búsqueda, se procedió a generar una representación bidimensional de los tweets utilizando la vectorización word2vec mediante T-SNE. Creamos diagramas de dispersión 2D y 3D, y como sabemos por teoría, dado que se emplea “distributed stochastic neighbor embedding”, deberíamos observar que tweets similares están representados por puntos cercanos.

Estos son los resultados:



Efectivamente, la suposición que habíamos hecho justo en el párrafo anterior es correcta. Profundizando en los gráficos obtenidos, por lo que hace a la representación 3D, es evidente que todos los tweets se fusionan en un gran cluster, probablemente debido a su enfoque común sobre la Guerra Ruso-Ucraniana; no obstante, podríamos llegar a decir que se diferencian dos clusters, esto podría ser debido a que la información de los tweets se centra en dos tópicos.

Mientras tanto, en la representación 2D, surgen distintos grupos basados en los subtemas abordados en los tweets (ej.: impacto humanitario, respuesta internacional, guerra mediática e informativa, proceso y negociaciones de paz, etc.).