Development of cyberweapons using Artificial Intelligence

Escuela Colombiana de Ingeniería Julio Garavito

Realizado por:
Yohanna Andrea Toro Duran
Daniel Alberto Rosales Castro

Director:
Daniel Orlando Díaz López

Trabajo dirigido

2019-II

Prefacio

Este libro presenta tanto los avances investigativos como prácticos junto con sus respectivos resultados del trabajo dirigido *Development of cyberweapons using Artificial Intelligence*. El libro posee el desarrollo investigativo y funcional de diferentes herramientas basadas en el uso de procesamiento de lenguaje natural para la detección del ciber terrorismo y análisis de información de inteligencia, con el fin de prevenir y contener de actividades terroristas futuras y perfilar adversarios.

Tabla de Contenidos

Lista de figuras

Lista de tablas

Introducción

Definición del proyecto

Objetivos generales

Construcción e investigación de una solución de seguridad ofensiva utilizando modelos de inteligencia artificial con el objetivo de aportar en ámbitos como:

- Reconocimiento del adversario
- Ejecución de campañas de ataque
- Eliminación de rastros o des atribución de un incidente
- Prevención y contención de futuros ataques ciber terroristas
- Monitoreo de contenido en redes sociales con afines terroristas

Objetivos específicos

- Identificar modelos de inteligencia artificial aplicables a ámbitos de la seguridad ofensiva, especialmente para: perfilamiento, explotación, des atribución mantenimiento de accesos entre otros.

- Proponer una solución de seguridad ofensiva basada en modelos de inteligencia artificial para representar un crecimiento en la capacidad operativa y estratégica de agencias de seguridad

- Validar la solución propuesta en escenarios reales para obtener una retroalimentación sobre la funcionalidad e identificar mejoras para proponer trabajos futuros

Desarrollo e Investigación

Módulo de recolección

Este módulo se encarga de Recolectar la información de la API de Twitter y distribuirla por todos los módulos para su correcto procesamiento y análisis.

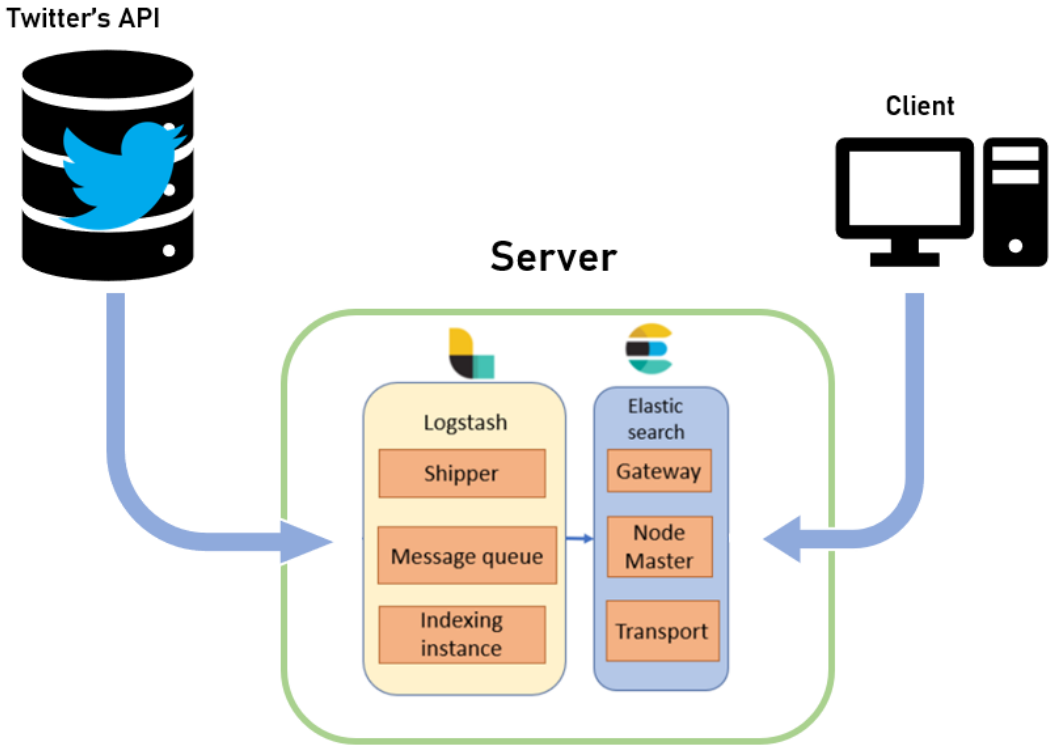

Figura 1: Arquitectura del módulo de recolección

Twitter's API

Twitter posee una API que nos permite recolectar tweets directamente de Twitter en tiempo real, estos vienen de forma no estructurada de todas partes del mundo.
Para las personas que quieran utilizar esta API esta permite:

- Crear, programar y administrar campañas publicitaras en twitter
- Usar y administrar diferentes herramientas proporcionadas por twitter para implementar en páginas web o aplicaciones
- Evaluar y analizar datos para la toma de decisiones

Nosotros aprovechamos la utilidad de recolección y análisis de datos que nos provee, pero a la vez es pertinente hablar sobre las limitaciones que también posee. La Api se maneja por medio de ventanas abiertas (se dice como ventana abierta a una conexión remota activa

a la API) la cual puede ser de dos clases, de Usuario y de Aplicación, es diferente la cantidad de POSTs y GETs que se pueden realizar para cada uno.

Endpoint	Resource family	Requests / window (user auth)	Requests / window (app auth)
GET search/tweets	search	180	450

Figura 2. Tabla de rate-limits de Gets de la API de twitter

Se creó una cuenta como aplicación ya que esta es la que permite más recolección de tweets por minuto. Hay que tener en cuenta que los números que se muestran en la figura 2 es en un tiempo de 15 mins por tanto solo es posible realizar la conexión durante 15 minutos sin interrupciones, una vez hayan pasado esos 15 mins, es necesario cerrar la ventana y realizar otra conexión

Servidor

Para poder recolectar todos los tweets de la aplicación se hace necesario tener un programa que tenga las herramientas necesarias para la recolección y el filtrado de estos. En el servidor se encuentra implementadas e instaladas dos herramientas las cuales permiten realizar la recolección, filtrado y almacenamiento de todos los datos no estructurados que se estén recolectando en tiempo real. Estas dos herramientas son Logstash y Elasticsearch.

1. Logstash:

Es un canal de procesamiento de datos de código abierto del lado del servidor que ingiere datos de una multitud de fuentes simultáneamente, los transforma y luego los envía al lugar en donde estos quieren ser almacenados.

2. Elasticsearch

Es un motor de analítica y análisis distribuido open source para todos los tipos de datos, incluidos textuales, numéricos, geoespaciales, estructurados y desestructurados. Elasticsearch nos permite, la estructuración de los datos en formato JSON para que estos sean correctamente analizados por los 3 modelos a desarrollar en este proyecto. Esta herramienta para realizar filtros de búsqueda dentro de todos los datos recogidos nos deja filtrar por idioma, por país, y por palabras clave que se encuentren dentro del tweet.

Para nuestro caso nosotros configuramos Elasticsearch para que filtre tweets de fuentes que estuvieran en Ingles y que contuvieran las siguientes palabras claves:

'jihad','alqaeda','taliban','islam','Libia','SriLanka','daesh','isis','terrorism','extremism','religion','quran','murder','wahhabi','muslims','younusalgohar','destiny','igbtq','AbuBakraiBaghd

adi','alratv','wahhabism','sufiimammehdigoharshahi','hatecrime','islamicstate' *(para más información sobre las palabras claves ir a la sección apéndice)*
Para la recolección de tweets recolectamos alrededor de 9000 tweets durante una semana, realizando las respectivas conexiones de 15 minutos.

Modelo de predicción de hashtags

Este modelo nos permite asignarle una o más etiquetas a un texto. En este caso usaremos los tweets recolectados para identificar en cada uno de ellos etiquetas relacionados al terrorismo.

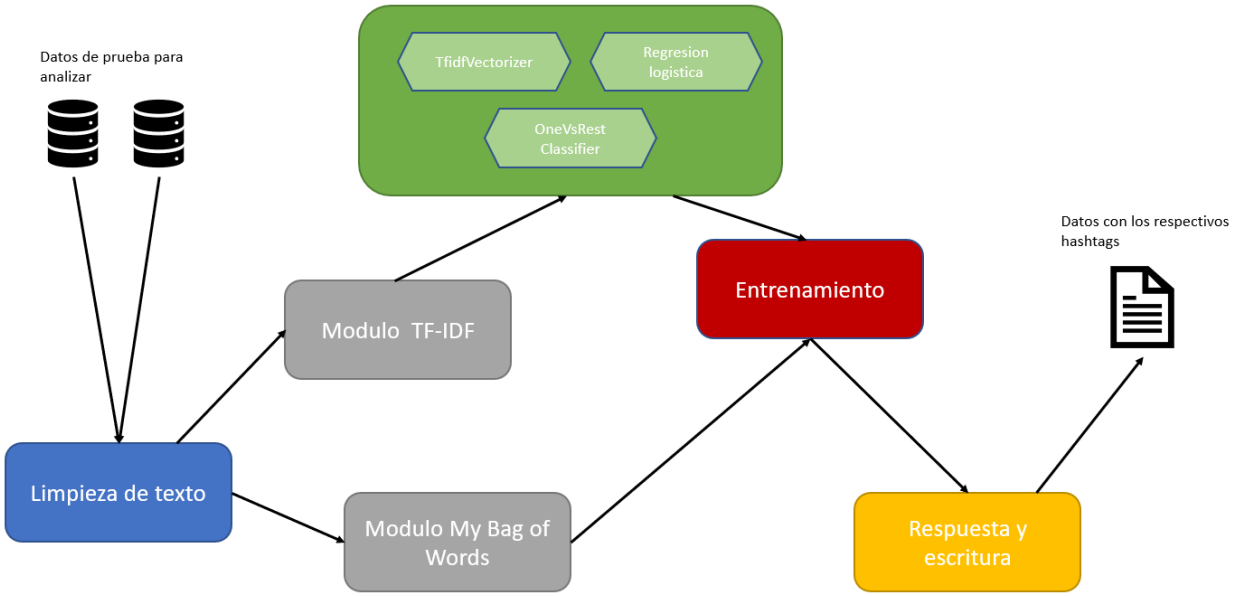

Figura 3. Arquitectura del modelo de predicción de hashtags

Enlace del repositorio(*https://github.com/GrCross/Development-of-cyberweapons-using-Artificial-Intelligence/tree/master/Tags_Prediction*)

Limpieza de texto

Como se está trabajando con datos no estructurados en necesario realizar un procesamiento de texto donde se unifica todo el texto en minúscula, se elimina caracteres especiales y se quitan *stop words* todo esto con el fin de evitar tener tokens extraños y genere algún falso positivo al momento del entrenamiento

ME	IT
MY	ITS
MYSELF	ITSELF
WE	THEY
OUR	THEM
OURS	THEIR
OURSELVES	THEIRS
YOU	THEMSELVES
YOUR	WHAT
YOURS	WHICH
YOURSELF	WHO
YOURSELVES	WHAT
HE	WHERE
HIM	WHOM
HIS	THAT
HIMSELF	THESE
SHE	THOSE
HER	AM
HERS	IS
HERSELF	WERE

Tabla 1. Lista de posibles stop words removidas

Bag of words:

La idea de este módulo es tener las N palabras con el fin de construir un diccionario. Para cada una de las palabras en el diccionario se le asigna un numero de 0 a N y se itera sobre el texto que se quiere vectorizar si alguna de estas palabras está en el diccionario el valor en el vector incrementa.

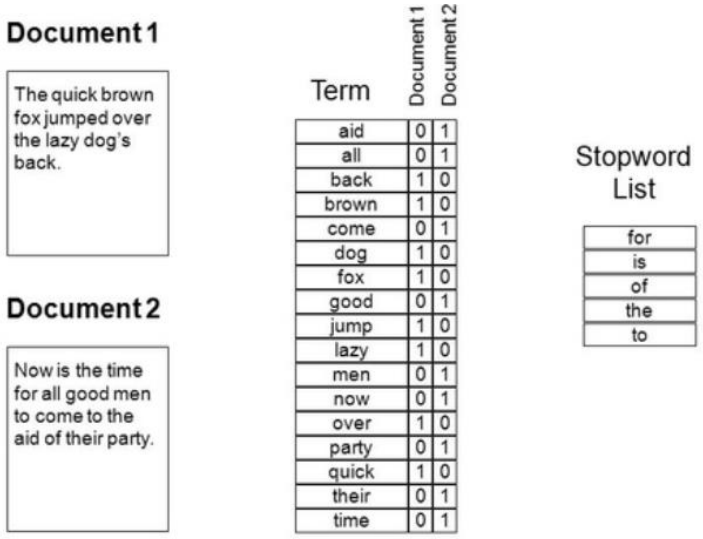

Figura 4. Ejemplo de clasificación con el método de stop words, en donde se tienen 2 documentos, un diccionario de entidades con su respectivo numero de apariciones en los diferentes textos y una lista de Stop words eliminadas

TF-IDF

Este módulo permite determinar la frecuencia de las palabras dentro del corpus, el cual permite tener una alta frecuencia de la palabra en todo el texto y adicional una baja frecuencia en toda la colección de textos que se está analizando lo cual permite realizar un mayor filtro en comparación a *bag of words* ya que se obtienen palabras que no son tan frecuentes en todo el conjunto de datos.

1. **One vs rest classifier:** es una estrategia que consta en asociar un conjunto de ejemplos positivos para una clase determinada y un conjunto de ejemplos negativos que representan todas las demás clases.
2. **Regresión logística**: es un tipo de análisis de regresión que permite determinar el resultado de una variable categórica.

Paquetes necesarios que usa el modelo:

- Numpy - un paquete para la computación científica.
- scikit-learn - una herramienta para el análisis de datos.
- NLTK – paquete para trabajar con lenguaje natural.

Modelo de identificación de entidades

Este módulo usa NER (Named-entity recognition) que sus siglas indican reconocimiento de entidades nombradas. Consiste la extracción de un texto en este caso los tweets entidades como personas, organizaciones, eventos, productos entre otras cosas.

Realizar NER sobre tweets es en particular complicado debido a las características únicas de los tweets. Por ejemplo, los tweets suelen ser cortos, ya que el número de caracteres de un tweet concreto se limita a 140, por lo que la información contextual es limitada. Además, el uso de lenguaje coloquial dificulta los enfoques que pueden existir en la aplicación de modelos enfocados en NER.

La solución parcial para el uso de lenguaje coloquial fue la correcta limpieza de los datos de entrada eliminando símbolos y palabras que no tuvieran un significado explicito.

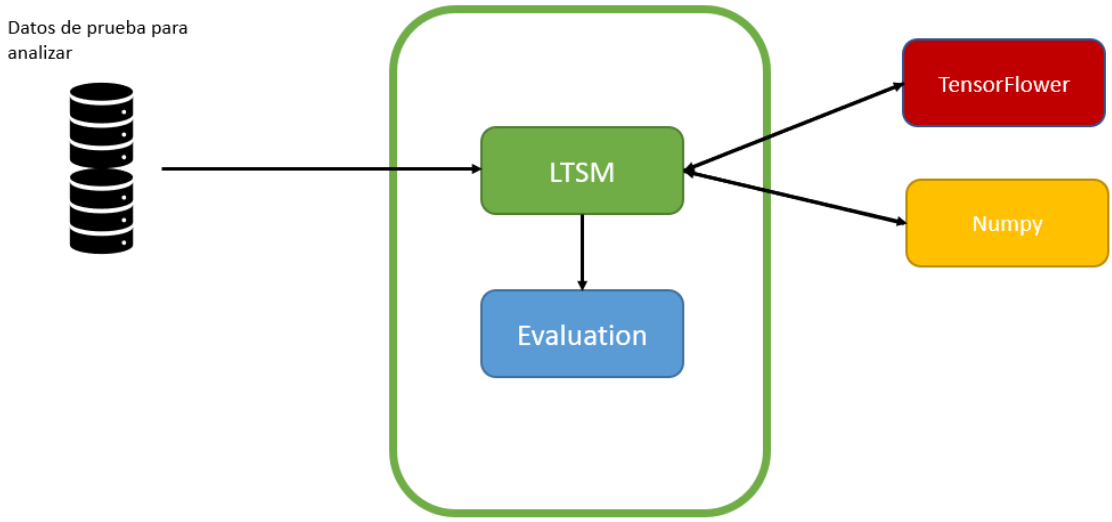

Figura 5. Arquitectura del modelo de identificación de entidates

Enlace del repositorio(*https://github.com/GrCross/Development-of-cyberweapons-using-Artificial-Intelligence/tree/master/NER-LSTM*)

Para determinar y clasificar cada una de las entidades se usó el esquema de marcado *BIO Markup* que posee los siguientes Tags preestablecidos.

PERSON	People including fictional
NORP	Nationalities or religious or political groups
FACILITY	Buildings, airports, highways, bridges, etc.
ORGANIZATION	Companies, agencies, institutions, etc.
GPE	Countries, cities, states
LOCATION	Non-GPE locations, mountain ranges, bodies of water
PRODUCT	Vehicles, weapons, foods, etc. (Not services)
EVENT	Named hurricanes, battles, wars, sports events, etc.
WORK OF ART	Titles of books, songs, etc.
LAW	Named documents made into laws
LANGUAGE	Any named language
DATE	Absolute or relative dates or periods
TIME	Times smaller than a day
PERCENT	Percentage (including “%”)
MONEY	Monetary values, including unit
QUANTITY	Measurements, as of weight or distance
ORDINAL	“first”, “second”
CARDINAL	Numerals that do not fall under another type

Tabla 2. lista de tags disponibles y sus descripciones

Bernhard	B-PER
Riemann	I-PER
Carl	B-PER
Friedrich	I-PER
Gauss	I-PER
and	O
Leonhard	B-PER
Euler	I-PER

Tabla 3. Ejemplo de cómo puede ser usado el BIO Markup

Bidirectional LSTMs

Consiste en la construcción de un bloque de redes neuronales el cual realiza un procesamiento sobre el texto para analizar los diferentes tokens de interés. El Long short-term memory se encarga de procesar el texto de izquierda a derecha mientras que el bidireccional realiza dicho análisis en la dirección opuesta.

Este modulo es entrenado mediante una *función de perdida cruzada de entropía* la cual es la medida de la diferencia entre dos distribuciones de probabilidad para una determinada variable aleatoria o conjunto de eventos. Cuando hablamos de entropía esta es el numero de bits requeridos que se requieren para transmitir un evento aleatorio seleccionado de una distribución de probabilidad.

$$H(p, q) = -\sum_{x} p(x) \log q(x).$$

Figura 6. Definición formal de la entropía, en donde p y q son dos variables aleatorias discretas

Una de las cosas que se necesitó por la demora en el entrenamiento y construcción de las redes neuronales usando funciones de perdida cruzada de entropía es la optimización de la función de perdida. Para esto se utilizo el algoritmo de optimización ADAM (adaptive moment estimation) el cual es un algoritmo de optimización que actualiza los pesos de las conexiones de la red de forma iterativa en base a los datos de entrenamiento.

Evaluation:

Este módulo se encarga de evaluar las diferentes entidades con métrica F1. La métrica F1 es una medida de precisión de una prueba, donde esta posee dos variables importantes una de precisión y otra de recuperación. La de precisión se encarga de determinar todas las entidades que son verdaderas devueltas por el clasificador y la variable de recuperación corresponde a todas las entidades ya sean positivas o negativas que tiene el recolector. Esta métrica realiza una clasificación binaria donde 1 corresponde a una recuperación y precisión correcta y 0 ocurre cuando dichos valores son incorrectos.

La variable de precisión se calcula dividiendo los resultados positivos correctos entre el numero de todos los resultados positivos generados por el clasificador mientras que la variable de recuperación se calcula dividiendo el numero de resultados correctos entre el número de muestras que debían haber sido identificadas como positivas.

1. Paquetes necesarios para el desarrollo del modelo:
   • Numpy un paquete para la computación científica.
   • Tensorflow una plataforma de código abierto para el aprendizaje automático.

Módulo de determinación de proximidad

El modelo de determinación de proximidad nos permite la comparación de diferentes textos o contenidos para poder determinar qué tan similares son estos, en nuestro caso nosotros utilizaremos los contenidos de los tweets recolectados para identificar que otros tweets recolectados son los más similares entre sí.

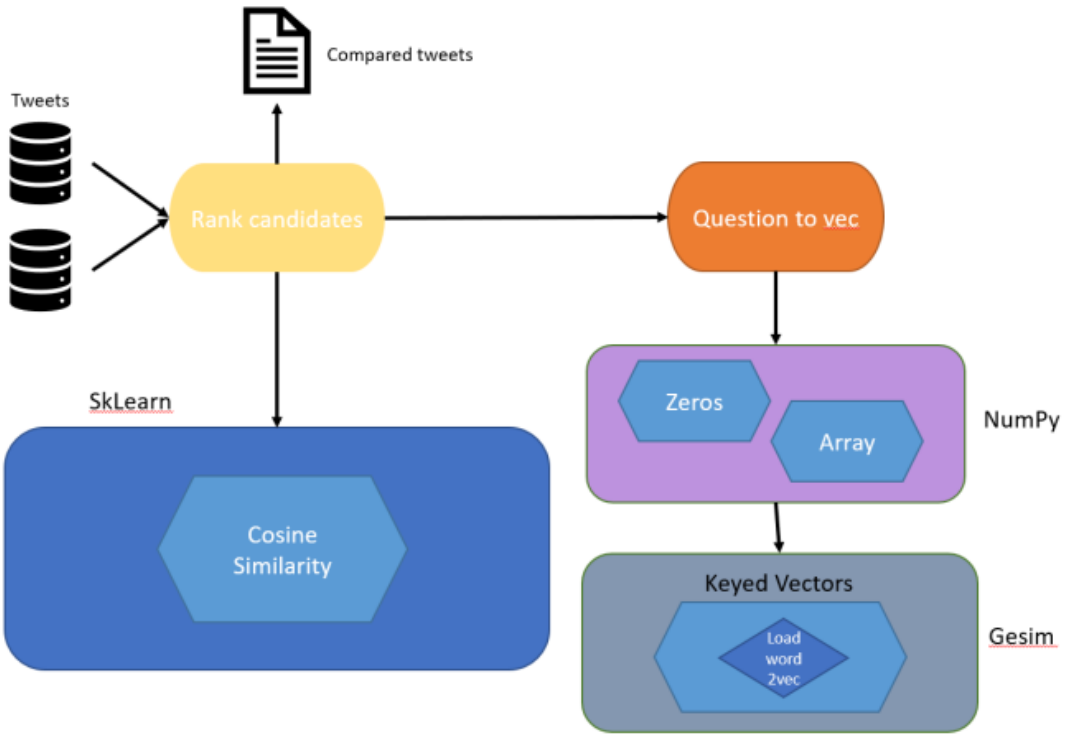

Figura 7. arquitectura del modelo de proximidad

Enlace del repositorio(*https://github.com/GrCross/Development-of-cyberweapons-using-Artificial-Intelligence/tree/master/Vector%20Space%20Models%20of%20Semantics*)

En este modelo se encuentran varios componentes para el análisis de proximidad teniendo en cuenta la semejanza del coseno que nos permite determinar qué tan parecido es un texto de otro.

Rank candidates

En esta parte es donde se reciben los tweets recolectados y el que nos da la salida de los tweets más parecidos entre sí. Aquí es donde con la ayuda de las diferentes herramientas de vectorización de texto y la comparación mediante Similitud del coseno determinar lo ya antes mencionado y ranquear a sí los tweets más similares.

Question to vec

Aquí se vectorizan los tweets para el correcto análisis futuro mediante métodos cuantitativos con la ayuda de diferentes librerías de Python como lo son Numpy y Gesim:

1. Numpy:

Numpy es una librería que nos proporciona una mayor flexibilidad al trabajar con vectores y matrices, está centrado en la computación científica usando Python. Además, lo podemos usar como un contenedor multidimensional de datos genéricos y así facilitar el análisis de los datos.

2. Gesim

Gensim es una biblioteca de código abierto para el modelado de temas no supervisados y el procesamiento del lenguaje natural, que utiliza modelos estadísticos modernos de maching learning.
Esta herramienta nos permite la estructuración y formación de diferentes fuentes de datos que se utilizaran para el entrenamiento del modelo. Estas fuentes las sacaremos en este caso de Google de una librería que posee embeddings especiales para el entrenamiento de este módulo[1].

Cosine Similarity

La evaluación del coseno nos permite mediante métodos cuantitativos saber que tan similar son dos palabras diferentes.

[1] https://code.google.com/archive/p/word2vec/

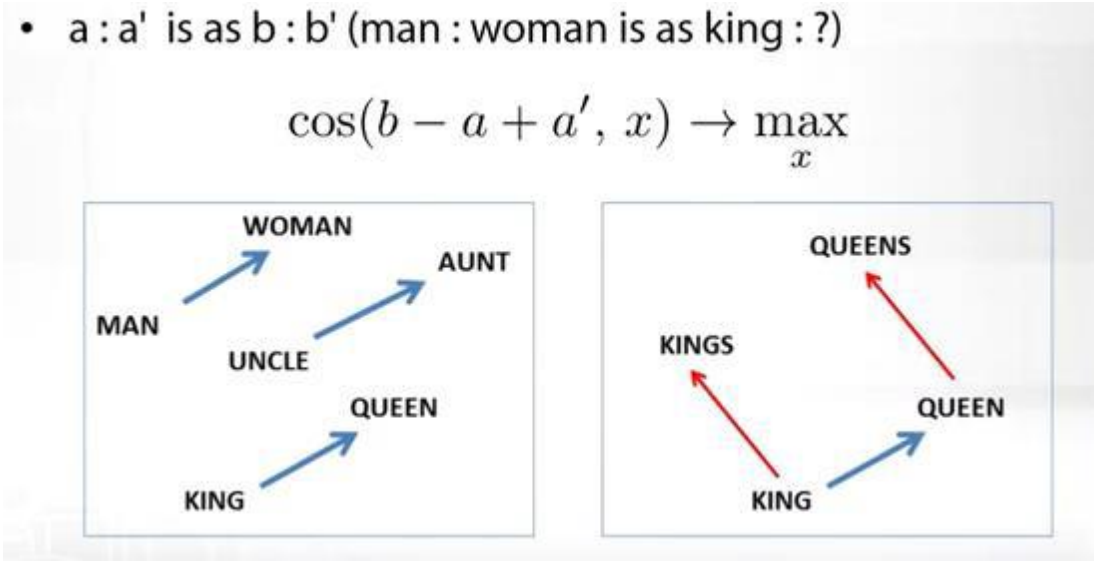

Figura 8. Ejemplo semejanza del coseno

En la anterior imagen se debe tener en cuenta que se están manejando vectores. Se tiene que al vectorizar Man el vector obtenido es similar al vector de Woman, a la vez el vector de Uncle y Aunt. Por tanto si tenemos una palabra como King, si aplicamos las operaciones aritméticas que se muestran en la imagen podemos decir: (King-man)+ Woman y el resultado será un vector similar al vector de Queen. Esto es porque al vector que define a King como hombre le estamos restando esa parte y agregando una representación femenina, la cual correspondería a la de Queen.

Para este trabajo estamos usando SKlearn que es una librería de Python la cual contiene la similitud del coseno implementada.

Diseminación y repositorio de resultados

Para saber sobre el código fuente de este proyecto, obtener los data sets que se recolectaron y probar los resultados obtenidos, estos se encuentran en el siguiente repositorio: https://github.com/GrCross/Development-of-cyberweapons-using-Artificial-Intelligence.git

Conclusiones

- Por medio del modelo de predicción de hashtags se pueden identificar contenidos relacionados con terrorismo con el fin de agilizar el proceso de análisis de información y prevención del ciberdelito, gracias a que las redes sociales hoy en día constituyen uno de los medios más frecuentes y con mayor facilidad para la distribución de información pública dirigida a una población especifica con el fin, ya sea de conseguir apoyo de la comunidad o incluso por lo contrario generar terror dentro de esta.

- El modelo de identificación de entidades permite notar de una manera rápida las organización o personas que están siendo mencionadas por parte de un conjunto de fuentes de información consideradas de interés.

- Gracias al modelo de identificación de entidades tuvimos la necesidad de crear un conjunto de datos de entrenamiento relacionados con ciberterrorismo para que al momento de entrenar el modelo se obtenga una mayor precisión y así evitar en gran mayoría los falsos positivos.

- Por medio del modelo de identificación de proximidad se es posible descubrir nuevas fuentes de información que inicialmente no se habrían considerado de interés por parte de un analista de inteligencia militar.

- Se es necesario un conocimiento previo en librerías de ciencias datos como los son pandas, numpy, sklearn entre otros para obtener mejores resultados, ya que estas librerías proveen muchas herramientas realmente útiles para el procesamiento de lenguaje natural.

- Para lograr mejores resultados en cada uno de los modelos se hace necesario una limpieza exhaustiva a los datos ya que esta ayudara a un mejor análisis con datos relevantes.

- Es necesario recalcar que este trabajo dirigido tiene muchas cosas en las que mejorar, como por ejemplo la precisión en los resultados obtenidos, como también la limpieza de los datos para reconocer el lenguaje coloquial que se habla en las diferentes redes sociales. Deseamos que en un futuro este trabajo sea ampliado o sea base para otros proyectos ya que la información e investigación que se desarrolló en el presente proyecto pude ser un insumo importante para temas relacionados con el procesamiento de lenguaje natural y la prevención de ciberdelitos.

Agradecimientos

Se agradece cordialmente al doctor y director de este trabajo dirigido Daniel Orlando Díaz López por sus constantes retroalimentaciones en el transcurso del proyecto y su aporte en conocimientos como también fuentes de información.

Lista de referencias

Timothy Baldwin, Young-Bum Kim, Marie Catherine de Marneffe, Alan Ritter, Bo Han, and Wei Xu. 2015. Shared tasks of the 2015 workshop on noisy user-generated text: Twitter lexical normalization and named entity recognition. 2015

Nut Limsopatham and Nigel Collier. Bidirectional LSTM for Named Entity Recognition in Twitter Messages, 2016-WNUT2016

Jason P.C. Chiu and Eric Nichols. Named Entity Recognition with Bidirectional LSTM-CNNs

Maria Heath, Orthography in Social Media: Pragmatic and Prosodic Interpretations of Caps Lock 2018

Anna Potapenko, Andrei Zimovnov Natural Language Processing week 1, week 2, week 3 Coursera.

Christopher Olah, Understanding LSTM Networks. August 27, 2015

Jason Brownlee, A Gentle Introduction to Cross-Entropy for Machine Learning, October 21, 2019

Jason Brownlee, Gentle Introduction to the Adam Optimization Algorithm for Deep Learning, July 3, 2017

Koo Ping Shung, Accuracy, Precision, Recall or F1?, march 15, 2018

Apéndice

Jihad: En el Occidente, “jihad” se traduce generalmente como “guerra santa”.

Alqaeda: Organización paramilitar radical dentro del islam político.

Islam: religión abrahámica monoteísta basada en el Corán

Daesh: Se trata de una transliteración del acrónimo árabe formada por las mismas palabras que componen ISIS (Estado Islámico de Irak y Siria, por sus siglas en inglés), es decir, 'Al-dawla al-islâmiyya fi l-'Irâq wa l-shâm'

Isis: siglas en inglés para “Estado Islámico de Irak y el Levante”

Wahhabi: forma estricta y conservadora del islamismo, es hoy en día es la religión oficial de Arabia Saudita.

Younusalgohar: AlGohar es un firme opositor del wahabismo, que considera una amenaza para el islam y el mundo entero.

AbuBakraiBaghdadi: anterior líder de ISIS

Alratv: es un canal de televisión islámico que dice estar dedicado al bienestar espiritual de la humanidad.

Sufiimammehdigoharshahi: líder espiritual y fundador del movimiento espiritual RAGS que ahora es conocido como Messiah Foundation International