

Proyecto 1 (P1)

El presente trabajo cuenta con 3 preguntas. Para las tres preguntas, usted deberá trabajar con la colección de datos listada a continuación:

Dataset smogon.csv

- La colección de datos será proporcionada por el profesor. La encontrarán en el campus virtual bajo el nombre smogon.csv
- Suba este archivo .csv a Google Drive y ábralo con Google Sheets.
- Esta colección de datos corresponde a la información disponible en la página <https://www.smogon.com/dex/xy/pokemon/> cada fila representa a un Pokémon.

La columna 'Pokemon' contiene el nombre del pokémon que se está analizando, la columna 'url' contiene la página web de donde se extrajo la información textual. La columna 'texto' contiene la descripción del pokémon y la columna 'moves' contiene información acerca de todos los ataques que aprender dicho pokémon (aquí encontrará tanto el nombre del ataque como la descripción del ataque).

Por ejemplo, en la fila 111, se encuentra el pokémon Chikorita, cuya información fue extraída del siguiente <https://www.smogon.com/dex/xy/pokemon/chikorita/>. Usted puede consultar dicha página web y notar que la lista de ataques empieza en la última sección, titulada «Moves», tal y como se muestra en la siguiente imagen:

Moves

Ancient Power	Rock		Power 60	Accuracy 100%	PP 5	10% chance to raise all stats by 1 (not acc/eva).
Aromatherapy	Grass		Power —	Accuracy —	PP 5	Cures the user's party of all status conditions.
Attract	Normal		Power —	Accuracy 100%	PP 15	A target of the opposite gender gets infatuated.
Body Slam	Normal		Power 85	Accuracy 100%	PP 15	30% chance to paralyze the target.
Bullet Seed	Grass		Power 25	Accuracy 100%	PP 30	Hits 2-5 times in one turn.

Notará que el primer movimiento que aprende Chikorita se llama Ancient Power, el segundo es Aromatherapy, el tercero es Attract, y así sucesivamente.

Cada uno de estos ataques tiene características escritas a su costado en formato textual, por ejemplo, Ancient Power tiene la palabra «Rock», luego tiene la

palabra «Power» y el número «60», luego tiene la palabra Accuracy, y muchas otras palabras.

Toda esta información se encuentra en el archivo separado por comas, aunque se una manera desordenada y confusa.

```
: LCMovesAncient PowerRock Power60Accuracy100%PP510% chance to boost all stats by one  
stage.AromatherapyGrass Power—Accuracy—PP5Cures status on the user's team.AttractNormal Power—  
Accuracy100%PP15Targets of the opposite gender are infatuated and have a 50% chance to do
```

El texto que se encuentra en esta celda no respeta las reglas sintácticas del Español, sin embargo, la computadora puede entender el conocimiento que está expresado en dicho texto.

Pregunta 1:

Agrupamiento mediante TF-IDF (7 ptos).

- Generar la matriz tf-idf utilizando una cantidad de n-gramas elegida por usted (unigramas, bigramas, trigramas, etc.).
- Mostrar el número total de tokens (elementos de su vocabulario) que tiene su matriz tf-idf.
- Imprimir todos los tokens (elementos de su vocabulario)
- Generar un DataFrame con la matriz tf-idf que tenga como cabeceras los elementos de su vocabulario. Imprimir dicha matriz usando la instrucción print().
- Agrupar las filas de su nuevo DataFrame, en base a sus puntuaciones tf-idf (usted puede elegir cuántos clusters desea utilizar).
- Generar un archivo de valores separado por comas (CSV) que contenga su matriz tfidf y el cluster (este csv será usado en la pregunta 3 del proyecto).
- Interpretar los cluster y ponerle un nombre a cada uno. Si no puede ponerle un nombre, escriba un párrafo explicando las características principales de dicho cluster. Es posible que solo algunos clusters tengan interpretación y otros no. Por ejemplo, si usted tiene 18 clusters y solo 10 de ellos pueden ser interpretados, explique el detalle de esos 10. Si no es posible interpretar ninguno de sus clusters, entonces repita el proceso usando otro número de n_clusters.

Pregunta 2:

Agrupamiento mediante un vocabulario controlado (6 ptos):

Usted implementará método para reprocesar la columna «moves» que le permita conseguir un agrupamiento que se ajuste a este dominio específico (es decir, el procesamiento que hará usted, solo servirá para agrupar los personajes de esta colección de datos, pero no tendrá utilidad para agrupar otras colecciones de datos).

Este procesamiento consiste en lo siguiente:

- Dado que el agrupamiento de personajes de esta franquicia es por tipos, usted deberá eliminar todas las palabras que no correspondan un tipo de pokémon.
- Para esto, usted deberá investigar cómo procesar toda la columna «moves» de su dataframe o de su csv, de modo que se eliminen todas las **palabras** o **partes de palabras** que no corresponden a la lista de tipos identificada por usted.
 - **Palabras:** Por ejemplo, si los únicos tipos existentes son “fire” y “water” y usted tiene la cadena “unleashes a scorching stream of water that sizzles with fire” quedaría como “water fire”.
 - **Partes de palabras:** En cambio, para el caso de palabras que contienen un tipo, se mantiene solo el tipo. Por ejemplo, la cadena “Blazing HydroburstWater projects a concentrated burst of superheated water that flashes into steam, enveloping targets in a scalding fog” quedaría como “water water”. Y la cadena “utecfirepython” quedaría como “fire”.
 - Debe tener en cuenta que una cadena puede contener más de un tipo. Así, la cadena “UTECwaterpythonWater” quedaría como water water, y la cadena “waterUTECfirepythonWater” quedaría como “water fire water”.
- Una vez que haya hecho este procesamiento, procese su nueva columna con tfidf y agrupe a los personajes. Para esta pregunta debe usar únicamente unigramas.
- Muestre los tokens.
- Indique cuántos tokens hay en total.
- Agrupe a los personajes e Interprete los clusters obtenidos.

Pregunta 3:

Agrupamiento mediante PCA (7 pts)

- Cargue en un DataFrame el archivo CSV generado en la Pregunta 1 del protecyo.
- Descarte la columna del cluster. Para esto, deberá referenciar el nombre de dicha columna. Imprima el DataFrame usando el comando print().
- Si su DataFrame tiene doble índice, descarte la primera columna. Para esto, deberá referenciar el atributo columns[0] de su DataFrame. Imprima el DataFrame usando el comando print().
- Aplique Análisis de Componentes principales utilizando la cantidad de componentes que usted desee.
- Imprima el número de filas y columnas de su DataFrame original.
- Imprima el número de filas y columnas de su matriz de componentes principales.
- Generar un DataFrame nuevo con la matriz de componentes principales que tenga como cabeceras el número de componente (por ejemplo, PCA1, PCA2, PCA3, etc.). Imprimir dicha matriz usando la instrucción print().

- Agrupar las filas de su nuevo DataFrame, en base a sus puntuaciones PCA (usted puede elegir cuántos clusters desea utilizar).
- Generar un archivo de valores separado por comas (CSV) que contenga su matriz PCA y el cluster.
- Interpretar los cluster y ponerle un nombre a cada uno. Si no puede ponerle un nombre, escriba un párrafo explicando las características principales de dicho cluster. Es posible que solo algunos clusters tengan interpretación y otros no. Por ejemplo, si usted tiene 18 clusters y solo 10 de ellos pueden ser interpretados, explique el detalle de esos 10. Si no es posible interpretar ninguno de sus clusters, entonces repita el proceso usando otro número de n_clusters.
- Compare el agrupamiento de la pregunta 3 con el de la pregunta 1.

Entregables:

Su trabajo deberá contener los siguientes elementos:

- Un zip con el proyecto de Python (debe incluir los archivos .py y .csv usados).
- Un informe en PDF en el que describa los pasos realizados, y se evidencie que está entregando todos los puntos solicitados en los párrafos anteriores. Al final de su documento debe escribir 2 conclusiones por cada miembro del grupo.
 - Cada miembro debe escribir una conclusión escrita con sus propias palabras, y otra conclusión escrita por un asistente de Inteligencia Artificial como chatGPT, Gemini, Claude o Copilot.
 - Después de escribir todas las conclusiones, deben analizar qué recursos ha usado el asistente de Inteligencia Artificial para escribir su conclusión que el los integrantes del curso omitieron. Esto les será de mucha ayuda para que puedan orientar más efectivamente sus conclusiones en proyectos futuros.

En la carátula de dicho informe debe incluir el nombre y código de los integrantes de su grupo y el porcentaje de participación de cada integrante. En la segunda hoja de dicho informe, debe detallar exactamente qué actividades del proyecto hizo cada integrante.

- Una presentación oral.
- Adicional: si usted usó un programa en python u otro lenguaje para preprocesar los textos antes de introducirlos en su proyecto, deberá adjuntar información probatoria de las tareas realizadas (puede ser el código fuente, un documento con capturas de pantalla y la explicación de los pasos, o un video). **No será válido que su trabajo inicie desde un punto ya pre-procesado que no haya sido explicado como parte de su entrega.**

Información obligatoria para el clustering:

Dado que se espera que el agrupamiento realizado en las tres preguntas describa cómo los personajes de esta franquicia se agrupan por tipos (fuego, agua, veneno, etc.), usted deberá especificar a qué tipo hace referencia cada cluster. Para eso, deberá agregar una columna a su dataset agrupado, con el potencial **de cada pokémon**, usando obligatoriamente uno o más de los siguientes enfoques:

- **Opción 1:** Utilizar las columnas Type 1 y Type 2 de Alberto Barradas, disponible en Kaggle, para completar el tipo de cada personaje, de modo que al filtrar cada cluster, se pueda ver al costado los tipos de Alberto Barradas.
- **Opción 2:** Utilizar la columna 'texto' del archivo smogon.csv para extraer el tipo del personaje. Típicamente el tipo o tipos de un pokémon está inmediatamente después de la palabra 'Type' e inmediatamente antes de 'Immune', pero si el personaje no tiene ninguna inmunidad, sería la palabra que aparece inmediatamente antes de 'Strongly resists', pero si el personaje no tiene ninguna doble resistencia, sería la palabra que aparece inmediatamente antes de 'Resists', pero si el personaje no tiene ninguna resistencia, sería la palabra que aparece inmediatamente antes de 'Weak to', pero si el personaje no tiene ninguna debilidad, sería la palabra que aparece inmediatamente antes de 'Very weak to'. Agregue una columna adicional con el tipo o tipos de cada personaje de modo que al filtrar cada cluster, se pueda ver al costado los tipos extraídos del CSV.
- **Opción 3:** Conectarse al JSON disponible en la página de smogon (usted puede presionar Ctrl + U en la página de cualquier personaje del CSV y ver el JSON completo al mover la barra de desplazamiento hacia la derecha). Agregue una columna adicional con el tipo de cada personaje de modo que al filtrar cada cluster, se pueda ver al costado los tipos extraídos del JSON de Smogon.
- **Opción 4:** Contar cuál es el tipo que más se repite en la columna 'moves'. Probablemente ese sea el tipo del personaje. Agregue una columna con el tipo o tipos que más se repite en cada personaje de modo que al filtrar cada cluster, se pueda ver al costado los tipos extraídos del CSV.
- **Opción 5:** Utilice un asistente de inteligencia artificial generativo, para pedirle que le proporcione el tipo de un conjunto de personajes elegidos por usted. Así, usted puede pedirle a chatGPT que agregue una columna adicional con el tipo del personaje. Tenga en cuenta que si le envía toda la base de datos completa, chatGPT, este solo hará bien las primeras filas y luego fantaseará en las siguientes. Es decir, inventará respuestas falsas.
- **Opción 6:** Extraer los tipos del github oficial de Smogon, disponible en <https://github.com/smogon/pokemon-showdown>. Agregue una columna adicional con el tipo o tipos de cada personaje de modo que al filtrar cada cluster, se pueda ver al costado los tipos extraídos del GitHub de Smogon.
- **Opción 7:** otro método elaborado por usted, pero debe ser fácil de comprender y replicar. Además, deberá especificar los pasos de su método para que el profesor pueda replicarlo y obtener el mismo resultado. Agregue una columna adicional

con el tipo de cada personaje de modo que al filtrar cada cluster, se pueda ver al costado los tipos inferidos por usted.

Rúbrica:

Criterio	EXCELENTE	ADECUADO	MÍNIMO	INSUFICIENTE
Desarrollo de software	Diseña y elabora el software para lograr una solución adecuada al problema planteado. El software debe ser ordenado, claro y óptimo. (10 p.)	Diseña y elabora el software para lograr una solución adecuada al problema planteado. El software es solo funcionable. (6 p.)	Diseña el software para lograr una solución adecuada al problema planteado. El software no se concluye adecuadamente. (4 p.)	No logra el diseño ni la implementación correcta del software. (0 p.)
Presentación escrita	El informe contiene las secciones de Antecedentes, Fundamento Teórico, Métodos y Desarrollo y Conclusiones. (5 p.)	El informe contiene las secciones de Antecedentes, Fundamento Teórico, Métodos y Desarrollo, pero no pone énfasis en las conclusiones. (3 p.)	El informe contiene menos de la mitad de las secciones estipuladas, incluyendo conclusiones. (2 p.)	El informe contiene menos de la mitad de las secciones estipuladas, sin incluir conclusiones. (0 p.)
Presentación oral	El alumno presenta el proyecto en forma adecuada y responde a las preguntas del profesor en forma lógica y coherente. (5 p.)	El alumno presenta el proyecto en forma adecuada, pero no responde a todas las preguntas del profesor en forma lógica y coherente. (3 p.)	El alumno no presenta el proyecto en forma adecuada, pero responde a las preguntas del profesor en forma lógica y coherente. (2 p.)	El alumno no presenta el proyecto en forma adecuada ni responde a las preguntas del profesor en forma lógica y coherente. O no se presenta a la presentación oral. (0 p.)