Proyecto Final:
Buscador Cinematográfico Basado en Ejemplos

Tratamiento de Datos
Máster en Ing. de Telecomunicación
Curso 2021/2022

November 22, 2021

1 Introducción

En este proyecto, los alumnos harán uso de los conocimientos y técnicas adquiridos durante el curso para resolver una tarea de aprendizaje sobre documentos textuales. Los alumnos trabajarán individualmente o por parejas sobre documentos descargados de un corpus de películas de cine, y las tareas a realizar incluirán necesariamente:

- Procesado y homogeneización de textos
- Modelado de tópicos con el algoritmo LDA
- Medidas de similitud entre películas basadas en sus descripciones

El proyecto tiene una valoración máxima de $2,5$ puntos. Consta de dos partes:

- Proyecto básico: $1,75$ puntos
- Extensión: $0,75$ puntos.

A continuación se indican los requisitos de cada una de las partes:

2 Proyecto básico

El objetivo principal del proyecto básico es diseñar un prototipo elemental del motor de análisis de un buscador de películas similares a una película dada. A

modo de ejemplo, el buscador podría funcionar al estilo de recomendadores publicos como *Tastedive*: el usuario proporciona una película, y el sistema devuelve un ranking o lista ordenada de las películas semánticamente más parecidas a la dada. No se pretende realizar una aplicación de usuario, sino solamente la algoritmia necesaria para determinar las películas más parecidas a una dada.

Para ello, será necesario:

1. Implementar al menos dos funciones que, dada una colección de películas, devuelvan una matriz de similitudes semántica entre ellas: al menos una de las medidas de similitud estará basada en vectores de bolsas de palabras o TFIDF, y otra debe estar basada en la representación de las películas en el espacio de tópicos obtenido por un algoritmo LDA.

2. Implementar una función que, dada una película, devuelva una lista ordenada de las películas más parecidas, de acuerdo con la medida de similitud elegida.

3. Idear e implementar algún procedimiento para evaluar la calidad del buscador.

Los algoritmos de modelos de tópicos tienen como finalidad determinar los temas principales que caracterizan a una colección (corpus) de documentos. Estos algoritmos obtienen una representación de cada documento como vector de tópicos, alternativa a otras representaciones, como las bolsas de palabras. Cabría esperar que el vector de tópicos sea más representativo del contenido temático de una película (en el contexto del corpus al que pertenece) que el vector de palabras o tokens basado en parámetros tf o tf-idf. Si esto es así, una medida de similitud basada en vectores de tópicos podría ser más relevante que una medida entre vectores del espacio de palabras o tokens.

El objetivo principal del proyecto básico es explorar la validez (o no) de esta hipótesis, comparando las prestaciones del buscador cuando se utiliza una medida de similitud basada en bolsas de palabras y cuando se utiliza una medida basada en vectores de tópicos.

2.1 Dataset

La elección de la base de datos para realizar el trabajo es libre. A modo de ejemplo, se proporciona un fichero del *Movielens* dataset. Existen otras alternativas, como el CMU Movie Summary Corpus, que contiene información relativa a decenas de miles películas, incluyendo título y resúmenes, pero la carga y preparación de datos con este dataset puede resultarle más complicada. En cualquier caso, si utilizara un dataset diferente a Movielens, justifique brevemente su elección en la memoria, y asegúrese en todo caso que el dataset contiene un resumen de cada película.

2.2 Medidas de similitud

La similitud entre dos documentos se puede calcular mediante una similitud entre las representaciones vectoriales de los documentos.

Su trabajo experimental debe analizar las siguientes alternativas para llevar a cabo, relativas a la representación de los documentos.

- Cada documento se representa por su vector de frecuencia de términos (Representación "TF") o cualquier medida relacionada, (como la representación "TF-IDF" o cualquier otra representación en el espacio de los tokens.
- Cada documento se representa mediante las proporciones de cada tópico proporcionadas por el algoritmo LDA.

Existen numerosas medidas de similitud entre vectores. En general, las medidas de similitud son opuestas a las medidas de distancia (a mayor distancia, menor similitud) así que puede implementar medidas de similitud a partir de medidas de distancia o de divergencia entre vectores, como la distancia euclídea, la distancia L_1 o la distancia coseno. Cuando los vectores representan probabilidades (como es el caso de las representaciones que resultan del algoritmo LDA) también suelen utilizarse medidas como la divergencia KL, la distancia de Jensen-Shannon o la distancia de Hellinger. Por supuesto, si le resulta conveniente, puede apoyarse en medidas de similitud o de distancia ya implementadas en librerías disponibles de Python.

2.3 Modelado de tópicos

La elección del número de tópicos, o los hiperparámetros del algoritmo LDA, puede afectar a las prestaciones del algoritmo. Explore la influencia de estos parámetros y determine valores que puedan ser más adecuados para este conjunto de datos y este problema.

2.4 Evaluación

Aunque en el trabajo puede incluir alguna evaluación subjetiva de las prestaciones de su algoritmo, basada en ejemplos, debe idear alguna forma de evaluación objetiva. Se sugieren a continuación algunas posibilidades:

- Comparar los resultados del buscador con los resultados de buscadores públicos, como *Tastedive*, sobre un conjunto reducido de películas seleccionadas.
- Comparar los resultados del buscador con preferencias temáticas evaluadas manualmente por un anotador.
- Comparar los resultados del buscador con similitudes inferidas por otros metadatos, como información del género de la película.

La evaluacion del buscador puede realizarse cooperativamente con otros grupos de la asignatura. Tanto el etiquetado como el desarrollo de código para la evaluación del software puede realizarse de forma cooperativa.

3 Extensión

El trabajo de extensión es libre: deberá ampliar el proyecto básico en la dirección que considere más oportuna: automatización de procesos, mejora de prestaciones, ampliación del estudio, etc.

Se sugiere a continuación algunas posibilidades:

- Incluir en el modelo información sobre las puntuaciones u otros metadatos disponibles en el dataset.
- Explorar la visualización del dataset mediante grafos definidos a partir de las medidas de similitud.
- Explorar el potencial de técnicas de NLP como el uso de bigramas, part-of-speech tagging, tesauros, etc, (explotando, por ejemplo, la funcionalidad disponible en la librería `NLTK` de Python).
- Desarrollar un buscador de películas basado en palabras clave a partir del modelo anterior, utilizando técnicas de expansión de queries (usando, por ejemplo, el módulo `wordnet` the python.
- Explorar medidas de similitud entre documentos basadas en word-embeddings, utilizando, por ejemplo, los embeddings proporcionados por Facebook[1])
- Diseñar un interfaz gráfico para una aplicación orientada a usuario.

Tome esta lista como una sugerencia. La extensión del trabajo es completamente libre, siempre que encaje dentro del ámbito de la asignatura. En todo caso, consulte con el profesor si tiene dudas sobre la idoneidad de un trabajo de extensión, o sobre cómo abordarlo.

4 Entrega

Los alumnos deberán proporcionar los siguientes entregables para la evaluación del proyecto final:

1. Memoria descriptiva del trabajo realizado en formato .pdf y una extensión máxima de 11 páginas (excluyendo únicamente portada y referencias).
2. Script de Python con el código implementado debidamente comentado

[1] Véase `https://fasttext.cc/docs/en/english-vectors.html`

Alternativamente a la memoria, los alumnos pueden entregar un notebook de jupyter que resuelva el proyecto y describa la metodología empleada, los experimentos realizados y sus resultados. La extensión máxima del Notebook resuelto y exportado a formato .pdf no podrá exceder las 40 páginas.

En caso de entregar una memoria, no debe incluir en ningún caso el código implementado, pero sí debe constar de cuatro apartados principales:

- Proyecto básico (max. 8 páginas de memoria, 32 págs de versión pdf del notebook)
- Extensión (max. 2 páginas de memoria, 8 págs de versión pdf del notebook)
- Manual de usuario del código (max. 1 página).
- Reconocimiento de autorías. Inexcusablemente, la memoria debe respetar el principio de reconocimiento de autorías. Si ha utilizado fragmentos de código ajenos o cualquier material procedente de fuentes externas, debe especificarlo claramente en la memoria. También debe indicar aquí los grupos con los que haya realizado la parte cooperativa del trabajo, si fuera el caso

5 Evaluación

El proyecto se evaluará de acuerdo con los criterios siguientes:

- Proyecto básico ($1,75$ puntos)
  - Metodología ($0,6$)
  - Calidad de la memoria ($0,75$)
  - Calidad del código ($0,2$)
  - Reproducibilidad de los resultados ($0,2$)
- Extensión ($0,75$ puntos)
  - Originalidad ($0,3$)
  - Calidad del trabajo ($0,45$)

La entrega se realizará vía Aula Global. La fecha límite será el **30 de diciembre**, a las 23,55 horas.