Jorge Altamirano Astorga (175904)
Uriel Miranda Miñón (177508)
Rodrigo Cedeño (176576)

Trabajo Final “Métodos Analíticos”

Introducción

Los sistemas basados en Sistemas de Recomendación y Procesamiento de Lenguaje Natural son cada vez más utilizados debido a las necesidades de las que se encuentra una solución con este tipo de sistema. Hemos podido ver ejemplos recientes como pueden ser los concursos de Netflix para mejorar sus sistemas de recomendación, lo que ha provocado que estos sean aún más populares. Estos sistemas son muy importantes debido a que el contenido actual de internet es demasiado grande, y es muy difícil poder filtrar o encontrar información precisa. Un sistema de recomendación soluciona este problema, al buscar de manera rápida y eficiente entre grandes volúmenes de datos.

Debido a esto, es muy atractivo desarrollar un sistema de recomendación basado en Procesamiento de Lenguaje Natural, con un enfoque distinto al que se puede encontrar publicado. Para este trabajo, fue desarrollado un sistema de recomendación en donde se clasifican distintos tipos de ingredientes para poder recomendar platillos de una región del mundo en específico. Dicho esto, este sistema será sumamente útil para una persona que cuente con determinados ingredientes y desee obtener una recomendación de compra de otros ingredientes para preparar comida de una región específica del mundo.

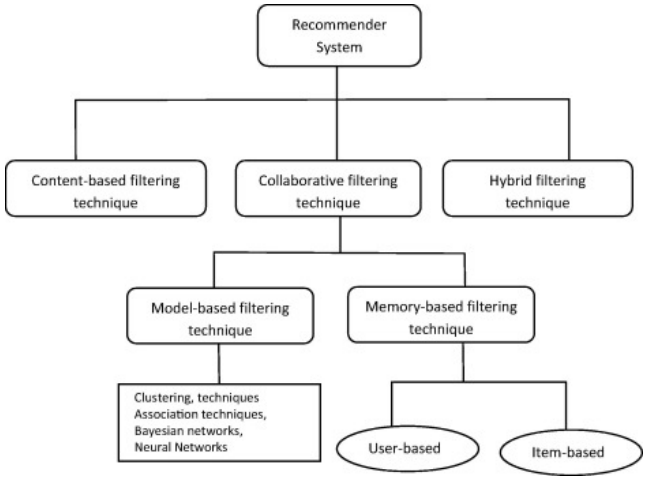
A lo largo de este documento se describirá el proyecto desarrollado, basado en las notas de clase de “Métodos Analíticos” impartida por el Dr. Luis Felipe González Pérez.

Sistemas de recomendación

Existen distintas técnicas de predicción en las que están basados los sistemas de recomendación, dependiendo el uso que se le vaya a dar al mismo, esto puede cambiar. Las tres ramas principales de los sistemas de recomendación son:

1. Content-based filtering
2. Colaborative filtering
3. Hybrid filtering

El siguiente diagrama obtenido del paper “Recommendation systems: Principles, methods and evaluation” nos muestra la estructura de estos métodos:



Para este proyecto se utilizó el método de “content-based filtering” (CBF) , debido a que está enfocado en el análisis de los atributos de los items para generar predicciones. En este tipo de método la recomendación está basada en perfiles de usuario utilizando características extraídas de contenido de usuarios que tienen evaluaciones pasadas. Las técnicas que pueden ser utilizadas para este método realizan recomendaciones a través del aprendizaje el modelo base ya sea con análisis estadístico o utilizando técnicas de aprendizaje de máquina. [1]

Implementación

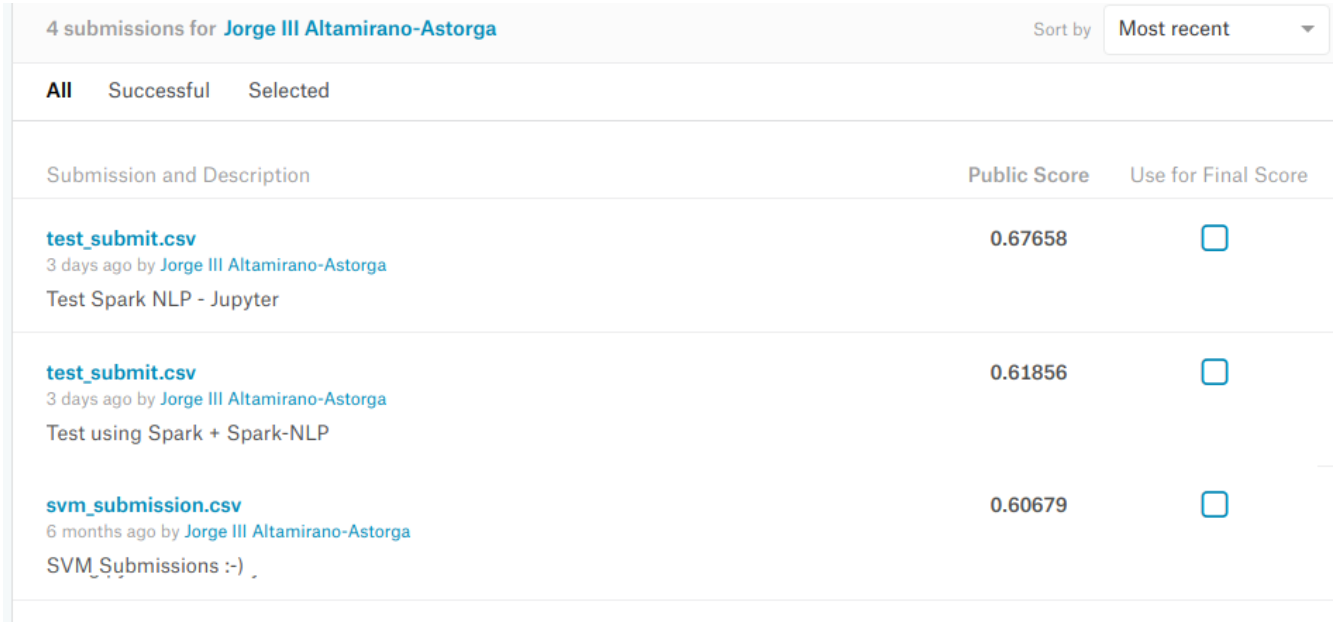
El dataset fue obtenido de la página de Kaggle [2], y los datos provienen de Yummly [3] la cual es un página de internet que es una plataforma de comida dedicada a la extracción de datos y Big Data. Ayudan a las personas a encontrar qué comer basado en preferencias personales recomendando recetas personalizadas para cada persona. Como ya fue mencionado antes, el objetivo de este proyecto fue realizar un sistema, donde al proporcionar ciertos ingredientes, el programa nos arrojará una clasificación de una cocina específica. Por ejemplo, en caso de que nosotros le proporcionemos aceite de oliva, sal, pasta y pimienta, el programa debe clasificar nuestros ingredientes como cocina italiana.

La implementación del proyecto fue realizada en Pyspark utilizando un paquete popular por su escalabilidad (escrito en Scala) y que nos resultó muy ágil: SparkNLP de John Snow Labs; además de utilizar algunas funciones de R para la parte del análisis de datos. Para el paso de tokenización dividimos las n-gramas con ‘espacios’; y posteriormente utilizamos stemming, para poder transformar cada uno de los tokens a las palabras raíz. Es destacable que para Word2Vec, y para Machine Learning, hicimos la tokenización separando por comas y no por espacios: pues los ingredientes muchísimas veces son compuestos, es decir no son formados por una palabra.

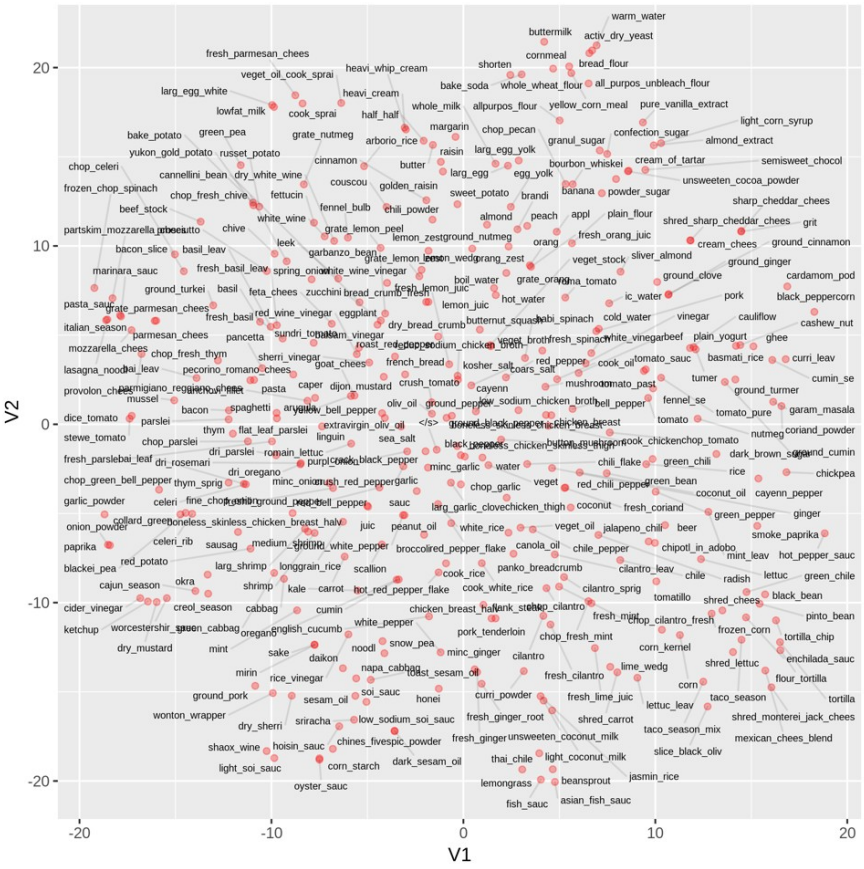
Esta parte es muy importante recalcar ya que no solo se utilizaron técnicas de aprendizaje de máquina, sino que se aplicaron métodos aprendidos en clase para poder hacer la reducción de dimensionalidad, tokenización, algoritmos de lenguaje natural, hashing (TDIDF) y stemming para ya tener consolidar los datos, evitar errores y que nuestro proceso sea mucho más eficiente y efectivo que como se ha hecho anteriormente basándonos en Regular Expresssions, que es una técnica: artesanal, poco escalable y altamente subjetiva.

Después de procesar los datos en nuestro pipeline obtuvimos una mejora significativa en el score que teníamos antes de implementar estos métodos analíticos. El F1 Score con la que medimos precisión y

recall, que teníamos antes era de 0.60 y se logró mejorar a 0.68, lo que significa una mejora del 14%, aunque lo más destacable no es en sí el performance ML, sino la efectividad de los métodos analíticos aplicados: los cuales están disponible de manera relativamente simple.



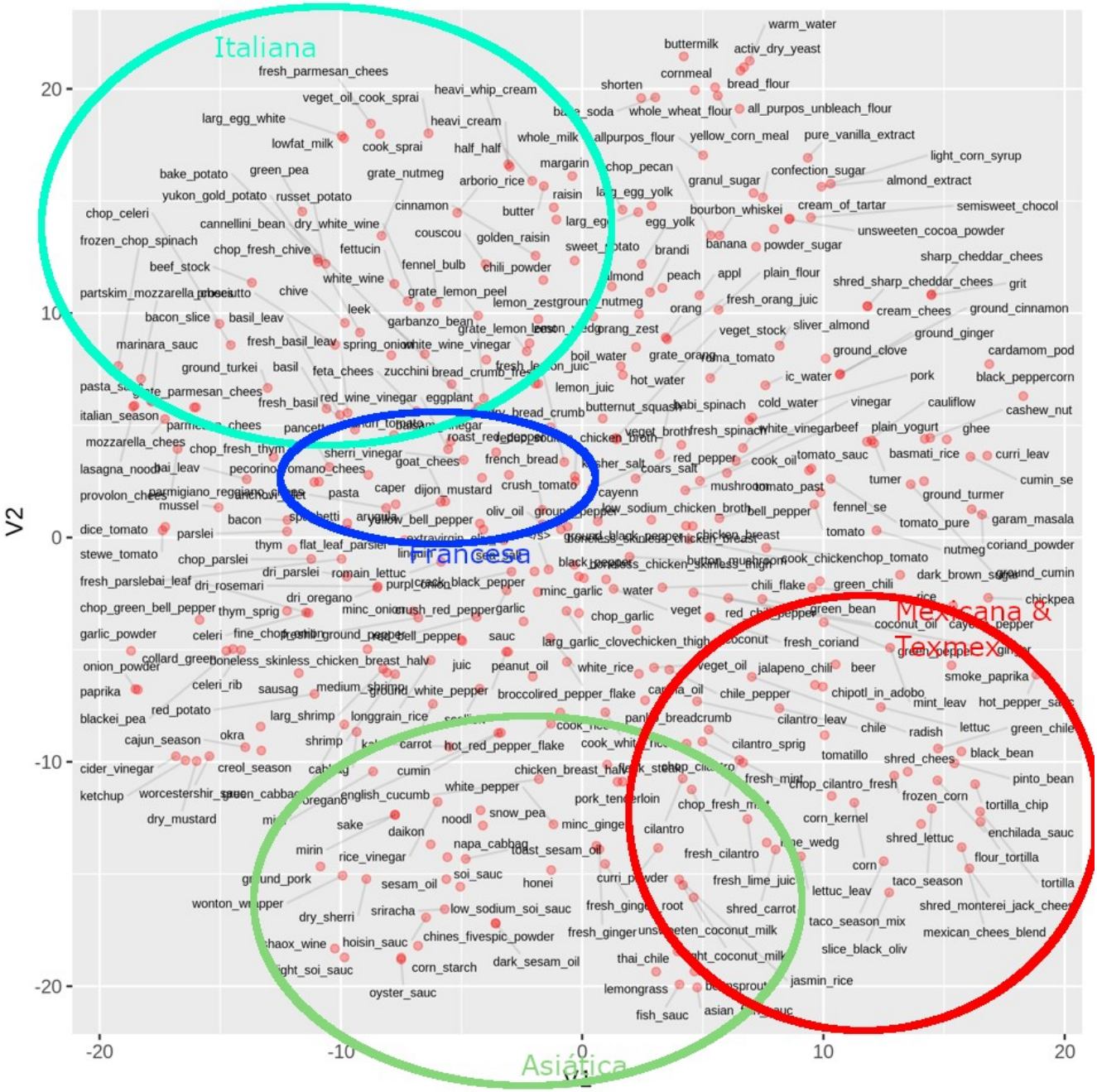
Análisis de los datos



En RStudio se creó la siguiente gráfica para poder visualizar cómo se realizó la distribución de los ingredientes en el espacio. Comenzando con esta gráfica, podemos identificar los diferentes tipos de cocina y los ingredientes que se utilizan en cada una de estas. Un ejemplo es la parte superior-izquierda, donde se encuentran muchos ingredientes de la comida italiana; por otro lado en la parte inferior de la gráfica se pueden encontrar ingredientes de comida asiática; y finalmente podemos observar que la comida mexicana se encuentra muy cercana a la comida asiática debido a que compartimos ingredientes con ese tipo de cocina, ahí podemos encontrar ingredientes como: tortilla, mexican cheese

blend, enchilada sauce, chile. También vemos que la comida americana incluye ingredientes de todo el mundo.

Todos estos resultados muestran mucho del poder de Word2Vec.



Conclusión

Este proyecto ha sido sumamente importante para aprender e implementar los métodos analíticos aprendidos en clase; debido a que no sólo se utilizaron algoritmos de machine learning que pueden llegar a ser sumamente útiles, sino que en esta ocasión se utilizaron algoritmos mucho más eficientes y capaces de procesar información que hacerlo a mano o por fuerza bruta.

Métodos como Word2Vec nos pueden ayudar a analizar datasets de gran escala que en un principio podrían parecer abrumadores, pero en realidad este tipo de métodos nos pueden ayudar al ser herramientas que nos permitan hacer manejable grandes (y no tan grandes) volúmenes de información. Las herramientas tienen el poder de ayudarnos de manera muy importante para poder realizar un análisis, manejo y manipulación apropiados de los datos.

Bibliografía

1. D Jurafsky, J H Martin. (2018). 4: Language Modeling with N-Grams. Speech and Language Processing(250). Stanford, CA: Prentice Hall.
2. https://www.sciencedirect.com/science/article/pii/S1110866515000341
3. https://www.kaggle.com/kaggle/recipe-ingredients-dataset
4. https://www.yummly.com/
5. Notas del Curso Métodos Analíticos, Luis Felipe González, ITAM Primavera 2018
6. https://github.com/JohnSnowLabs/spark-nlp/blob/master/python/example/model-downloader/ModelDownloaderExample.ipynb
7. https://nlp.johnsnowlabs.com/components.html
8. https://nlp.johnsnowlabs.com/notebooks.html
9. https://github.com/JohnSnowLabs/spark-nlp/blob/1.5.0/python/example/vivekn-sentiment/sentiment.ipynb
10. Indix - Lessons from Using Spark to Process Large Amounts of Data – Part I. Retrieved 2018-05-14
11. Spark NLP - Dependencies
12. StackOverflow: Troubleshotting on Spark
13. https://github.com/JohnSnowLabs/spark-nlp/issues/106
14. https://stackoverflow.com/questions/34302314/no-module-name-pyspark-error