UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE ESTUDIOS ESTADÍSTICOS
Máster en Minería de Datos e Inteligencia de Negocios

TRABAJO FIN DE MÁSTER (TFM)

Aplicación de técnicas de Big Data Science para la gestión de crisis.
Application of Big Data Science techniques for crisis management.

PRESENTADO POR

Caio Fernandes Moreno

DIRECTOR

Dr. Ramón Alberto Carrasco Gonzalez

Madrid, 2016

Tabla de contenido

1. AGRADECIMIENTOS 5
2. RESUMEN 6
3. INTRODUCCIÓN 7
4. CRONOGRAMA 9
5. OBJETIVOS 10
6. JUSTIFICACIÓN 10
7. METODOLOGÍA 11
8. PRELIMINARES 11
9. REVISIÓN DE BIBLIOGRAFÍA 16
10. ARQUITECTURA CONCEPTUAL PROPUESTA 27
11. CONFIGURACIONES PRÁCTICAS DE LA ARQUITECTURA. APLICACIÓN A UN CASO REAL 29
11.1 PRIMERA VERSIÓN 29
11.2 SEGUNDA VERSIÓN 43
12. MATERIAL Y MÉTODOS 56
13. CONCLUSIONES Y TRABAJO FUTURO 57
14. REFERENCIAS 59

Tabla de figuras

FIGURA 1 - EDW 12
FIGURA 2 – CUADRANTE GARTNER 14
FIGURA 3 – EJEMPLO DE EXPRESIÓN DE SENTIMIENTO 17
FIGURA 4 – EJEMPLO DE EXPRESIÓN DE SENTIMIENTO 18
FIGURA 5 – ARQUITECTURA CONCEPTUAL 29
FIGURA 6 – BÚSQUEDA EN TWITTER 30
FIGURA 7 – "MAP OF A TWITTER STATUS OBJECT" 31
FIGURA 8 - HISTOGRAMA DE LA PUNTUACIÓN (SCORE) 40
FIGURA 9 - NUBE DE PALABRAS 41
FIGURA 10 - BASE DE DATOS 42
FIGURA 11 - CUADRO DE MANDO CREADO CON CTOOLS/PENTAHO 42
FIGURA 12 - CUADRO DE MANDO CREADO CON CTOOLS/PENTAHO 43
FIGURA 13 - PROPUESTA DE ARQUITECTURA PRÁCTICA 45
FIGURA 14 - CUADRO DE MANDO LEYENDO LOS DATOS DE MONGODB 47
FIGURA 15 - CÓDIGO HDFS + APACHE FLUME + TWITTER 49
FIGURA 16 - DATOS DE TWITTER ALMACENADOS EN HDFS CON APACHE FLUME 49
FIGURA 17: TRABAJO (JOB) PARA LEER LOS DATOS DE TWITTER Y GOOGLE SEARCH Y GRABAR EN UNA BASE DE DATOS POSTGRESQL Y EN EL DATA LAKE (HDFS Y HIVE) 50
FIGURA 18 - DATOS DE TWITTER COLECTADOS POR LA HERRAMIENTA PDI 51
FIGURA 19 - DATOS DE TWITTER Y GOOGLE SEARCH EN EL DATA LAKE COLECTADOS CON PDI 51
FIGURA 20 - CONSULTA A LOS DATOS DE TWITTER ALMACENADOS EN APACHE HDFS/HIVE 52
FIGURA 21 - CALCULADORA AMAZON 54
FIGURA 22 - CUADRO DE MANDO CON ANÁLISIS DE SENTIMIENTO Y RELEVANCIA CREADO CON PENTAHO 55
FIGURA 23 - NUBE DE PALABRAS 56

1. Agradecimientos

Agradezco a mi Padre Celestial, a mi familia, a la Universidad Complutense de Madrid, a todos los profesores y funcionarios de la Universidad, a mi director de TFM (trabajo fin de máster) Dr. Ramón Carrasco y a mis compañeros de estudios.

Especialmente a mi mujer y mis dos hijas.

Sólo, yo no sería capaz ni de empezar.

2. Resumen

A pesar de la existencia de una multitud de investigaciones sobre el análisis de sentimiento, existen pocos trabajos que traten el tema de su implantación práctica y real y su integración con la inteligencia de negocio y big data de tal forma que dichos análisis de sentimiento estén incorporados en una arquitectura (que soporte todo el proceso desde la obtención de datos hasta su explotación con las herramientas de BI) aplicada a la gestión de la crisis.

Se busca, por medio de este trabajo, investigar cómo se pueden unir los mundos de análisis (de sentimiento y crisis) y de la tecnología (todo lo relacionado con la inteligencia de negocios, minería de datos y *Big Data*), y crear una solución de Inteligencia de Negocios que comprenda la minería de datos y el análisis de sentimiento (basados en grandes volúmenes de datos), y que ayude a empresas y/o gobiernos con la gestión de crisis.

El autor se ha puesto a estudiar formas de trabajar con grandes volúmenes de datos, lo que se conoce actualmente como *Big Data Science*, o la ciencia de los datos aplicada a grandes volúmenes de datos (*Big Data*), y unir esta tecnología con el análisis de sentimiento relacionado a una situación real (en este trabajo la situación elegida fue la del proceso de *impechment* de la presidenta de Brasil, Dilma Rousseff). En esta unión se han utilizado técnicas de inteligencia de negocios para la creación de cuadros de mandos, rutinas de ETC (Extracción, Transformación y Carga) de los datos así como también técnicas de minería de textos y análisis de sentimiento.

El trabajo ha sido desarrollado en distintas partes y con distintas fuentes de datos (*datasets)* debido a las distintas pruebas de tecnología a lo largo del proyecto.

Uno de los *datasets* más importantes del proyecto son los tweets recogidos entre los meses de diciembre de 2015 y enero de 2016. Los mensajes recogidos contenían la palabra "Dilma" en el mensaje. Todos los *twittees* fueron recogidos con la *API de Streaming* del *Twitter*. Es muy importante entender que lo que se publica en la red social Twitter no se puede manipular y representa la opinión de la persona o entidad que publica el mensaje. Por esto se puede decir que hacer el proceso de minería de datos con los datos del Twitter puede ser muy eficiente y verídico.

En 3 de diciembre de 2015 se aceptó la petición de apertura del proceso del *impechment* del presidente de Brasil, Dilma Rousseff. La petición fue aceptada por el presidente de la Cámara de los Diputados, el diputado Sr. Eduardo Cunha (PMDB-RJ), y de este modo se creó una expectativa sobre el sentimiento de la población y el futuro de Brasil.

También se ha recogido datos de las búsquedas en Google referentes a la palabra Dilma; basado en estos datos, el objetivo es llegar a un análisis global de sentimiento (no solo basado en los twittees recogidos).

Utilizando apenas dos fuentes (Twitter y búsquedas de Google) han sido extraídos muchísimos datos, pero hay muchas otras fuentes donde es posible obtener informaciones con respecto de las opiniones de las personas acerca de un tema en particular. Así, una herramienta que pueda recoger, extraer y almacenar tantos datos e ilustrar las informaciones de una manera eficaz que ayude y soporte una toma de decisión, contribuye para la gestión de crisis.

3. Introducción

Los sitios web de microblogging se han desarrollado hasta el punto de convertirse en una valiosa fuente de variada clase de información, debido a su capacidad de recoger lo que la gente publica en los mensajes en tiempo real acerca de sus opiniones sobre una diversidad de temas: cuestiones debatidas en la actualidad, se queja y expresa el sentimiento positivo, negativo o neutral en cuanto a determinada materia, etc.

Hay varios factores que pueden desencadenar una crisis, como por ejemplo: grandes cambios en los valores de índice de la bolsa y en las tasas de cambio, los desastres naturales. Dichas crisis se ven reflejadas en los sentimientos que esos factores producen en las personas. Es posible decir que a la suma de sentimientos generales (expresados a través de cualquier medio), cuando es conocida, se puede proceder a la categorización de una cierta situación. La caída de la Bastilla en Francia, en 1789, es un ejemplo de cómo la suma de los sentimientos de cierta cantidad de personas puede categorizar una situación: muchos franceses tenían un sentimiento negativo en relación a la monarquía, lo que causó una situación a la que se puede categorizar como crisis (y el posterior cambio de la forma de gobierno de Francia). (National Geographic, 2012)

Una definición de crisis, y que viene muy bien en al ámbito de análisis de sentimientos, es: “cambio profundo y de consecuencias importantes en un proceso o una situación, o en la manera en que estos son apreciados.” (RAE, 2016). Basado en esta definición se puede concluir que Francia sufrió una crisis.

Con esa definición de crisis se puede definir que una gestión de crisis es la suma de medidas tomadas para enfrentar, adaptar y hasta superar un profundo cambio sufrido dentro de una realidad gubernamental, social o empresarial.

Es cierto decir que en la época de la caída de la Bastilla no se tenía la tecnología existente hoy, las propias acciones radicales de la población fueron los datos que demostraron el sentimiento de la mayoría de la nación y que la gestión de la crisis resultó en el cambio del sistema de gobierno de Francia.

En aquella época tener o no el conocimiento previo de la suma de los sentimientos de la población quizás no importara al Rey, en cambio, para los gobiernos democráticos actuales, se cree que, dicho conocimiento tiene gran importancia en las toma de decisiones de los gobernantes.

Pero no solamente los gobiernos están sujetos a enfrentar crisis, no solamente a los gobernantes políticos les es importante poseer información estratégica, para las empresas y los empresarios también es importante. Un ejemplo de lo crucial que es para las empresas poseer información y obtener determinados conocimientos es el caso de la empresa norte americana JetBlue. Esta empresa se enorgullece de su reputación excepcional de servicio al cliente. Sin embargo, cuando la costa este de los Estados Unidos fue golpeada con una tormenta de hielo mortal en 2007, JetBlue se vio obligada a cancelar muchos vuelos a medida que sus operaciones se fueron derrumbando. Las consecuencias de la catástrofe fueron enormes. Debido a que JetBlue tiene una estricta política de no cancelar vuelos, los pasajeros enfurecidos se quedaron atrapados en los aeropuertos durante casi una semana. Llevaron a Internet a su indignación y la empresa fue arrastrada a una tormenta caótica. El director ejecutivo David Neeleman actuó rápidamente para anular el alboroto. Negándose a culpar al clima por el desastre, Neeleman escribió una carta pública de disculpa, redactó una declaración de derechos de los clientes y estableció un plan para compensar a los clientes afectados. Apareció en televisión en vivo, en YouTube y otros programas y medios para ofrecer una disculpa sincera en nombre de JetBlue. (Julia Hanna, 2008).

En el caso de la empresa norteamericana un desastre natural la hizo pasar por una crisis, pero el director ejecutivo, probablemente basándose en los datos a los que él tenía acceso (y también en los comentarios que las personas escribían – posiblemente con sentimientos negativos) junto con todo el conocimiento de negocios que tenía, él tomó una decisión que seguramente ayudó la empresa a recuperar la confianza de sus clientes, y también a salir de la crisis.

Desde 1789 hasta 2007, y de 2007 hasta hoy, la cantidad de informaciones y datos disponibles a los gestores de los más distintos departamentos y sectores ha crecido mucho. Así, ahora tenemos una gran dificultad: la existencia de demasiados datos e informaciones (los cuales recoger, extraer, almacenar, interpretar y analizar puede asistir en la toma de decisiones y en la gestión de crisis).

Además, la construcción de la tecnología para detectar y resumir dichas informaciones, trabajando con grandes cantidades de datos, es un desafío muy grande. Tan grande es el desafío que no existen muchos trabajos que propongan arquitecturas completas (que incluyen desde la origen de los datos hasta la visualización de la información de manera sensiblemente comprensible) para la gestión efectiva de crisis. Buscando encontrar una manera de superar dicho desafío, este trabajo se enfoca en encontrar esa arquitectura.

En este trabajo, utilizando el microblog popular llamado Twitter (que es la plataforma más importante existente actualmente en condiciones de recoger los sentimientos de la gente y sus opiniones y proporcionarlas de forma gratuita y abierta) será desarrollada una herramienta sobre la base de sus tweets donde se construirán modelos de clasificación de sentimiento en positivo o negativo.

Esta herramienta puede ser utilizada por el Gobierno, o por los ciudadanos, para entender mejor a la población, para entender lo que la gente de la muestra recogida piensa (y también sus líderes) en las redes sociales, identificar las personas más activas en el uso de las herramientas sociales para expresar su opinión y también identificar a los más influyentes.

La solución presentada es una solución de Big Data Science y puede ser implementada en un entorno de producción con muchos datos, lo único necesario es invertir dinero en la infraestructura de servidores necesarios para el almacenamiento de muchos datos y procesamiento de los datos (es importante tomar nota de que en el curso de este trabajo resultó bastante costoso la creación de una infraestructura de Big Data capaz de almacenar y procesar todos los datos recogidos, luego la herramienta ha sido creada y pensada para soportar muchos datos, pero para este trabajo nos limitaremos a trabajar con pocos datos, los suficientes para hacer todas las pruebas).

Es muy importante decir que en ningún momento se busca, en este trabajo, encontrar una solución ideal y capaz de tener la responsabilidad de juzgar el sentimiento. Mucho se habla de sistemas capaces de tomar decisiones de forma autónoma, pero en este trabajo no se busca en ningún momento transferir esta importante responsabilidad de los humanos hacia las máquinas, la capacidad de decidir. En su charla en TED, Zeynep Tufekci afirmó que no podemos transferir nuestras responsabilidades hacia las máquinas. (Zeynep Tufekci, 2016).

Teniendo las informaciones clave es mucho más fácil gestionar las crisis en el principio en lugar de dejar que pase mucho tiempo. Este trabajo busca crear una herramienta capaz de apoyar el trabajo de personas humanas en la gestión de la crisis dentro de las organizaciones. Esta compuesto por:

- la introducción: explica y ejemplifica lo que es una crisis y cuál es el objetivo principal de este trabajo.
- objetivos: discrimina y detalla los objetivos principales y secundarios de este trabajo.

- justificación: explica el motivo de utilizar la herramienta creada y para qué fue creada – qué funciones tiene.
- metodología: aclara cómo fue creada la herramienta y cómo funciona.
- preliminares: presenta las definiciones de conceptos importantes y necesarios para el entendimiento de la parte teórica y práctica de este trabajo.
- revisión de la bibliografía: expone los conceptos definidos y explicados por grandes y respetables autores de la esfera de la tecnología y ciencia.
- arquitectura propuesta: propone una arquitectura para la solución presentada.
- configuraciones prácticas de la arquitectura y aplicación a un caso real: muestra las configuraciones de la arquitectura de la herramienta y se desmuestra cómo se puede utilizar esa herramienta en un caso real.
- conclusiones y trabajo futuro: describe las conclusiones a las que se han llegado y considera cuáles son las futuras aplicaciones problabes de esta herramienta; y
- referencias.

4. Cronograma

Abajo se puede ver el cronograma inicial del TFM.

		Mes 1				Mes 2				Mes 3				Mes 4				Mes 5			
Tareas	**% de conclusión**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**
Estudiar la API de Twitter y Google Search para entender como extraer datos en tiempo real filtrando por palabras claves deseadas	100%	X	X																		
Extraer los datos de Twitter y Google Search grabar en un archivo json	100%	X	X	X	X	X	X	X	X	X	X	X	X								
Hacer el parser del archivo JSON y grabar en una base de datos MySQLy PostgreSQL	1000%	X	X	X	X	X	X	X	X	X	X	X	X								
Crear visualizaciones con los datos originales	100%	X	X	X	X	X	X	X	X	X	X	X	X								
Encontrar un Lexicon en portugués	100%	X	X	X	X	X	X														
Crear un script en Python para separar el Sentilex 1.0 e 2.0 en dos archivos (pos.txt y neg.txt)	100%							X													
Crear una función en R para limpiar los datos sucios de los mensajes de twitter y Google Search	100%								X												
Crear una función en R para contar la frecuencia de palabras negativas y positivas presentes en lo mensaje y definir el sentimiento	100%									X	X										
Crear una función en R para hacer la nube de palabras	100%											X									
Crear un cuadro de mando para visualizar los indicadores	100%					X	X	X	X	X	X	X									
Ejecutar la carga y procesamiento final con todos los datos colectados y generar el análisis del estudio	100%					X	X	X	X	X	X	X									
Probar las distintas tecnologías de Big Data y integrar con el trabajo	100%													X	X	X	X	X	X	X	X
Crear nuevos cuadros de mandos para la mejor visualización y integración con el mundo Big Data	100%													X	X	X	X	X	X	X	X

El cronograma final se ha cambiado mucho debido al gran deseo del autor de probar distintas tecnologías, herramientas y técnicas; eso ha dificultado muchísimo la conclusión del trabajo, por haber consumido mucho tiempo en pruebas y prácticas. Como consecuencia ha sobrado poco tiempo para formular, analizar y elegir una única solución mejor para el problema de crear una herramienta de toma de decisión para analizar tweets y datos de Google Search.

Se puede decir que se ha llegado a una versión 0.1 Alpha que se considera como la versión final de la herramienta.

5. Objetivos

Es posible detallar el objetivo de este trabajo respondiendo la pregunta ¿qué?

- ¿Qué?

Objetivo principal del trabajo: definir una arquitectura conceptual de análisis de sentimiento global que tenga en cuenta diversas fuentes de datos (como *twitter y google*) y que genere conocimiento que se pueda integrar en las herramientas *OLAP* de la organización para ayudar en la gestión crisis.

Objetivos secundarios: probar diversas configuraciones prácticas de dicha arquitectura con las principales herramientas existentes actualmente incluyendo las principales de *Big Data* y obtener conclusiones sobre las más adecuadas.

6. Justificación

- ¿Por qué?

Debido a que los datos extraídos de Twitter son información pública, porque la persona que los publicó expresó su opinión de su propia y legítima voluntad en una red social, esta información es muy valiosa y puede ser extraída y utilizada para fines positivos. Esta información no debe utilizarse para fines malos, de ninguna manera este control puede ser un instrumento de coacción o intimidación de la población.

Así, uno de los usos positivos, propuesto en este trabajo, es el empleo de la herramienta en cuestión con el propósito de conseguir, a través de la muestra recogida, entender el sentimiento u opinión de la muestra recogida en ese momento (que fue un hecho histórico del año 2015 a 2016: el proceso de la solicitud de *impechment* del Presidente de Brasil) y enriquecer dichos datos con los sentimientos extraídos de las búsquedas hechas en Google.

- ¿Para qué?

El objetivo de la investigación es aprender y aplicar las técnicas Big Data Science, Big Data, Business Intelligence, análisis de sentimiento y procesamiento de lenguaje natural con la finalidad de entender el sentimiento de las personas en relación a Dilma, ex-presidente de Brasil.

7. Metodología

- ¿Cómo?

Mediante la realización de cuatro acciones principales: extracción y filtro de los datos colectados, preparación de los datos, predicción del sentimiento utilizando un algoritmo de clasificación o la técnica de contar palabras (frecuencia) que sean negativas y positivas y después, la categorización de los datos en positivo y negativo, y la visualización de las siguientes informaciones:

- ¿Quiénes son las personas con más seguidores con la mayor cantidad de mensajes negativos?
- ¿Quiénes son las personas con más seguidores con la mayor cantidad de mensajes positivos?
- ¿Quiénes son las diez personas con el mayor número de seguidores con una mayor cantidad de mensajes negativos?
- ¿Quiénes son las diez personas con el mayor número de seguidores con una mayor cantidad de mensajes positivos?
- ¿Cuáles son los mensajes más relevantes? (La medida de relevancia se hará a través de la cantidad de *retweets* hechos. Es decir, cuantas más menciones, o *retweets*, más relevante el mensaje).

8. Preliminares

Actualmente se estudian los distintos asuntos de forma separada debido a la gran complejidad de cada tema. Lo que se propone en este trabajo es una solución robusta de Big Data Science, donde se juntan las técnicas de Business Intelligence, Big Data y Data Science.

Para un mejor entendimiento se explicarán algunos conceptos importantes de forma muy resumida para el entendimiento de Big Data Science.

Big Data Science

Mucho se habla de Big Data Science como la unión de Big Data y Data Science. La ciencia de los datos (Data Science) ha sido por muchos años estudiada por matemáticos, estadísticos y los profesionales de inteligencia de negocio, pero con la llegada de la era de los muchos datos *(Big Data Era),* principalmente debido a la llegada de las redes sociales, estamos viviendo un momento donde se unen distintas áreas de conocimiento para llegar a ser capaces de extraer valor de los datos.

Existen muchos conceptos que se pueden asociar a lo que llamamos aquí como Big Data Science, son:

- Big Data
- Data Science

- Artificial Intelligence
- Business Intelligence
- Machine Learning
- Data Mining
- Predictive Analytics
- Data Lake
- Enterprise Data Warehouse (EDW)
- ETC (Extracción Transformación y Carga) o en ingles *ETL (Extraction Transform and Load).*

Inteligencia de Negocio (Business Intelligence (BI))

La inteligencia empresarial (BI) es un término general que incluye las aplicaciones, la infraestructura, las herramientas y las prácticas más eficaces que permiten el acceso y el análisis de la información para mejorar y optimizar las decisiones y el rendimiento. (Gartner, 2016).

EDW (Enterprise Data Warehouse)

De acuerdo con Gartner (2016), un almacén de datos es una arquitectura de almacenamiento diseñada para almacenar datos extraídos de sistemas de transacción, almacenes de datos operativos y fuentes externas. A continuación, el almacén combina esos datos de forma agregada y resumida, adecuada para el análisis de datos y la generación de informes a nivel de toda la empresa para necesidades empresariales predefinidas.

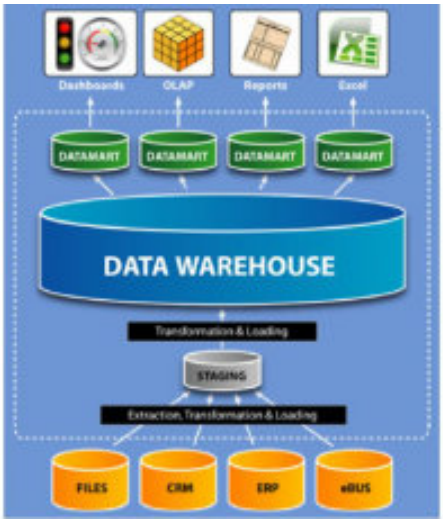


Figura 1 - EDW

Los cinco componentes de un almacén de datos son:

1. Fuentes de datos de producción
2. Extracción y conversión de datos
3. El sistema de gestión de base de datos del almacén de datos
4. Administración de los almacenes de datos
5. Herramientas de inteligencia empresarial (BI)

Un almacén de datos contiene datos organizados en áreas sujetas a abstracción con versiones variantes en el tiempo de los mismos registros, con un nivel apropiado de grano o detalle de datos para que sea útil a través de dos o más tipos diferentes de análisis, a menudo desplegados con tendencias a la tercera forma normal. Un *data mart* contiene datos similarmente variantes en el tiempo y orientados al sujeto, pero con relaciones que implican el uso dimensional de datos en los cuales los hechos son distintamente separados de los datos de dimensión, haciéndolos más apropiados para categorías individuales de análisis. (Gartner, 2016)

Lago de Datos (Data Lake)

Un lago de datos es una colección de instancias de almacenamiento de varios activos de datos adicionales a las fuentes de datos de origen. Estos activos se almacenan en una copia casi exacta, o incluso exacta, del formato de origen. El propósito de un lago de datos es presentar una visión no refinada de los datos sólo a los analistas más calificados, para ayudarles a explorar sus técnicas de refinamiento y análisis de datos independientemente de los compromisos del sistema de registro que puedan existir en una analítica tradicional Almacén de datos (como un *data mart* o *data warehouse*). (Gartner, 2016)

Grandes Datos (Big Data)

Los grandes datos son activos de información de gran volumen, alta velocidad y / o alta variedad que exigen formas innovadoras de procesamiento de información rentables que permiten una mejor comprensión, toma de decisiones y automatización de procesos. (Gartner, 2016)

Científico de datos / Data Scientist

La función de científico de datos es crítica para las organizaciones que buscan extraer conocimiento de los activos de información para iniciativas de "grandes datos" y requiere una amplia combinación de habilidades que se pueden cumplir mejor como un equipo, por ejemplo: La colaboración y el trabajo en equipo son necesarios para trabajar con negocios Partes interesadas para entender las cuestiones empresariales. Las habilidades analíticas y de modelado de decisiones son necesarias para descubrir relaciones dentro de los datos y detectar patrones. Se requieren habilidades de

gestión de datos para construir el conjunto de datos relevante utilizado para el análisis. (Gartner, 2016)

En 2012, Thomas H. Davenport y D.J. Patil escribieron en Harvard Business Review que el trabajo de científico de los datos será el trabajo más sexy del siglo XXI. (Thomas H. Davenport y D.J. Patil, 2012)

El informe que publicó "McKinsey" en junio de 2011 estimó que para el mundo de grandes datos en el que vivimos, se espera que la demanda por talento experto en análisis de datos podría alcanzar de los 440,000 a 490,000 puestos de trabajo para el 2018. (Steven Overly, 2013)

En un artículo de Bloomberg se afirma que los sueldos de los científicos de datos ya están por encima de 200.000 dólares al año. (Rodrigo Orihuela, Dina Bass, 2015).

Predictive analytics / Analítica Predictiva

La analítica predictiva describe cualquier método de minería de datos con cuatro atributos:

1. Un énfasis en la predicción (en lugar de descripción, clasificación o agrupación)

2. Análisis rápido medido en horas o días (en lugar de los meses de minería de datos tradicionales)

3. Hacer hincapié en la pertinencia comercial de las ideas resultantes (sin análisis de torre de marfil / *ivory tower analyses*)

4. Cada vez, con más frecuencia, un énfasis en la facilidad de uso, que las herramientas sean accesibles a los usuarios de negocios.

(Gartner, 2016).

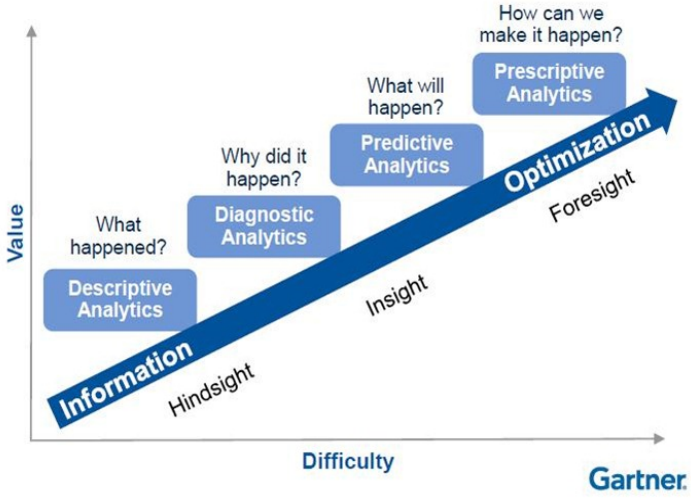


Figura 2 – Cuadrante Gartner

Minería de Datos (Data Mining)

"La minería de datos es la extracción de información implícita, previamente desconocida y potencialmente útil de los datos. La idea es construir programas de computadora que tamizan las bases de datos automáticamente, buscando regularidades o patrones. Los patrones fuertes, si se encuentran, probablemente se generalizarán para hacer predicciones precisas sobre los datos futuros. ... El aprendizaje automático proporciona la base técnica para la minería de datos. Se utiliza para extraer información de los datos en bruto en las bases de datos ... "

En el capítulo 1 del libro, los autores Ian Witten and Eibe Frank escriben:

"La minería de datos se define como el proceso de descubrir patrones en los datos. El proceso debe ser automático o (más habitualmente) semiautomático. Los patrones descubiertos deben ser significativos en que deben conducir a alguna ventaja, por lo general económica. Los datos están invariablemente presentes en cantidades sustanciales ". (Witten, I. H., & Frank, E. 2005)

Aprendizaje automático (Machine Learning)

Los algoritmos avanzados de aprendizaje automático están compuestos de muchas tecnologías (como el aprendizaje profundo, las redes neuronales y el procesamiento del lenguaje natural), utilizadas en el aprendizaje no supervisado y supervisado, que funcionan guiadas por las lecciones de la información existente. (Gartner, 2016)

Análisis prescriptivo (Prescriptive Analytics)

La analítica prescriptiva es una forma de análisis avanzado que examina los datos o el contenido para responder a la pregunta "¿Qué se debe hacer?" O "¿Qué podemos hacer para que _____ suceda?", Y se caracteriza por técnicas como análisis de gráficos, simulación, complejos procesamientos de eventos, redes neuronales, motores de recomendación, heurística y aprendizaje automático. (Gartner, 2016)

Análisis en tiempo real (Real-time analytics)

La analítica en tiempo real es la disciplina que aplica la lógica y las matemáticas a los datos para proporcionar ideas para tomar mejores decisiones rápidamente. Para algunos casos de uso, el tiempo real simplemente significa que el análisis se completa en unos segundos o minutos después de la llegada de nuevos datos. La analítica en tiempo real bajo demanda espera a que los usuarios o sistemas soliciten una consulta y luego entregan los resultados analíticos. El análisis continuo en tiempo real es más proactivo y alerta a los usuarios o activa las respuestas a medida que ocurren los eventos. (Gartner, 2016)

9. Revisión de bibliografía

En esta parte del trabajo se revisará la bibliografía existente sobre el tema de análisis de sentimiento y las técnicas existentes actualmente. Se ha utilizado el trabajo "*A Social Media Sentiment Analysis Model to Support Marketing Intelligence in Kenya*" hecho por Kiptanui Dennis Too, donde el autor realiza un extenso trabajo de revisión bibliográfica de los principales conceptos e investigadores de los últimos años en el área de análisis de sentimiento.

Se ha realizado una adaptación y traducción del inglés al español con el objetivo de ayudar el entendimiento de los retos y técnicas.

Algunas palabras se han dejado también en inglés para facilitar el estudio de otros investigadores que puedan desear utilizar este trabajo como base para trabajos futuros (muchas veces los términos solo en español resultan difíciles de entender, y cuando son utilizados juntos con el término anglosajón se facilita muchísimo la investigación).

El análisis de sentimiento se conoce como la aplicación de *machine learning* y estadística para determinar el sentimiento en un texto, se habla también de este tema como análisis de opinión.

Análisis de Sentimiento o análisis de opinión.

Desde siempre el hombre ha valorado la opinión de un servicio, producto, persona y empresa antes de su propia toma de decisión. En la vida cotidiana existe la costumbre de preguntar qué restaurante u hotel se recomienda, se contratan muchas personas por medio de recomendaciones o referencias, y muchas otras situaciones donde la opinión de los demás influye en la toma de cierta decisión por una persona.

Se valora una recomendación en base a dos criterios principales:

a) Relevancia
b) Opinión / Sentimiento

A) Relevancia

La relevancia es muy importante porque algo puede tener mayor importancia si la persona es de gran relevancia; un ejemplo práctico es cuando dos personas recomiendan el mismo producto, pero una de las personas causa más impacto debido a su mayor relevancia. Otro ejemplo práctico es cuando un periódico publica un artículo sobre algo, la opinión de un periódico tiene más relevancia muchas veces que la opinión de una persona común.

B) Opinión / Sentimiento

El sentimiento es una clasificación hecha por medio de un algoritmo, donde se clasifica un texto como positivo o negativo, en algunas veces se introduce también el sentimiento neutro.

Análisis de opiniones no es algo nuevo

Desde siempre se pide la opinión de una madre o un padre para decir si se debe o no casarse con una persona, o si se debe comprar un coche, además de muchas otras situaciones similares. Desde siempre el ser humano pide consejo a otros, eso es parte de la naturaleza humana: pedir opinión de otras personas que consideramos expertos en esos temas.

Lo que ha cambiado mucho en los últimos años ha sido el surgimiento de las redes sociales para intercambiar nuestras opiniones. Antes, nuestras opiniones tenían una propagación limitada, pero con la llegada de las redes sociales y el Internet, nuestras opiniones o sentimientos pueden llegar hasta cualquier parte del mundo y contribuir con a que el mundo siga cambiando.

Debido a este cambio, las empresas y gobiernos se ven obligados a tener una estrategia de gestión de crisis, para garantizar que algo que puede empezar en un mensaje de twitter o cualquier red social no se transforme en una crisis incapaz de ser gestionada, sin tener que invertir muchísimo tiempo y recursos.

Existen muchas herramientas y fuentes de datos disponibles donde se puede obtener la opinión de las personas.

Una simple búsqueda en twitter sobre una palabra hace posible tener una idea general de lo que se habla de un tema, de una organización, persona u otro tema cualquiera. En la imagen abajo se puede ver que buscando la palabra #dilma es posible obtener mensajes positivos o negativos sobre el tema.

Figura 3 – Ejemplo de expresión de sentimiento

Figura 4 – Ejemplo de expresión de sentimiento

Las opiniones son centrales para casi todos los aspectos relacionados con los humanos y aún son una de las claves para influenciar nuestro comportamiento. El análisis de sentimiento es el estudio computacional de las opiniones, sentimientos y emociones expresadas en un texto. El origen del análisis de sentimiento tiene su raíz en las disciplinas de la Psicología, Sociología y Antropología. (Rambocas et al, 2013)

Se habla de análisis de sentimiento con distintos términos como minería de opinión, extracción de la evaluación y también como análisis de subjetividad. (Kadam et al, 2013)

El análisis de sentimientos busca asociar automáticamente un fragmento de texto con un puntaje (*score*) de sentimiento. (Kumar et al, 2012)

Pang et al (2002) clasificaron los documentos por polaridades positivas o negativas. Dirigieron su estudio en las revisiones de la película. Compararon tres técnicas diferentes de aprendizaje de máquinas: Naïve Bayes, Support Vector Machine y Maximum Entropy. Posteriormente, añadieron la dimensionalidad del análisis mediante puntajes de revisión de calificación. Descubriendo que el mejor clasificador entre los tres era la máquina del vector del apoyo. También se dieron cuenta que el acercamiento de aprendizaje de la máquina dependía grandemente del dominio. Esto es porque las palabras pueden representar diversas emociones en diversos dominios.

Hu y Liu (2004) estudiaron el análisis del sentimiento de los comentarios de los clientes sobre los productos. Propusieron un marco para analizar y comparar las opiniones de los consumidores sobre los productos competidores. El prototipo de opinión Observer usó el descubrimiento de reglas supervisadas para extraer características y sus correspondientes pros y contras. Realizan el estudio en tres etapas. Identificaron la característica primero, después, para cada característica, calcularon las polaridades antes de finalmente resumir esas revisiones.

Go et al. (2009) utilizaron el aprendizaje a distancia para adquirir datos de sentimientos. Teniendo en cuenta tweets que terminan con emoticonos positivos como J: - * y emoticonos negativos como L "": - / como positivo y negativo respectivamente. Comparando los modelos SVM, NB y MaxEnt. El modelo SVM superó al resto. También consideraron que el espacio de la característica y el unigram funcionaron mejor.

Pak y Paroubek (2010) utilizaron emoticonos para identificar tweets y entrenar a un clasificador. Recopilaron datos utilizando un paradigma similar de aprendizaje a distancia, pero clasificados en términos subjetivos u objetivos. Informaron que tanto POS como bigrams ayudan. Ambos estudios se basan en modelos n-gram.

Qi, Guo y Hinrich (2013) utilizaron la máquina del vector de ayuda (SVM) con una gama grande de características, de características de la posición, de características estilísticas, de emoticons, de nombre de dominio, de legibilidad y de otras estadísticas para clasificar tweets. Presentaron la selección de todas las características descritas usando la información mutua y la validación cruzada de diez veces. Descubrieron que las estadísticas simples de los tweets tales como conteo de la palabra o de la legibilidad pueden asistir en el análisis del sentimiento del gorjeo.

Turney et al (2002), mientras estudiaba nuevamente las revisiones de los clientes, introdujo un algoritmo no supervisado como recomendado o no recomendado. Los artículos clasificados se basan en frases fijas y sintácticas utilizadas para expresar opiniones. Utilizaron frases en lugar de palabras e implementaron el algoritmo basado principalmente en información mutua puntual y recuperación de información para calcular la orientación semántica de la revisión dada. El estudio se realizó en tres etapas. 1) Comenzó por extraer primero el léxico phrasal de las revisiones. 2) Luego se identificó la polaridad de cada frase. 3) Con respecto a la polaridad media de una frase, una revisión fue polarizada.

Mullen, Tony y Nigel Collier (2004) introdujeron el análisis de concepto utilizando máquinas de vector de soporte (SVM) para reunir diversas fuentes de información. Introdujeron modelos que usan características y combinan los modelos de unigram. Utilizaron el método de orientación semántica con PMI. Después de experimentar con el dominio de revisión de películas, descubrieron que las SVMs híbridas que combinan SVM de SVM con estilo unigram con aquellas basadas en medidas de favorabilidad reales obtuvieron un rendimiento superior.

Mohammad et al (2013) estudiaron cómo construir-estado-del-arte en el análisis del sentimiento de los tweets. Incorporan la máquina del vector de ayuda que tenía un F-cuenta de 69.02 en el nivel del mensaje y de 88.93 en el nivel del término. Implementaron una variedad de características basadas en la forma superficial y las categorías léxicas. La mayor parte de la ganancia en el rendimiento fue liderada por las características del léxico del sentimiento junto con las características del n-gramo.

Subjetividad y objetividad

No todas las piezas del texto suelen contener opiniones útiles. La clasificación de subjetividad / objetividad viene a separar oraciones subjetivas de oraciones objetivas. Pang et al (2002) estudiaron el análisis del sentimiento determinando si una oración es subjetiva u objetiva. Las oraciones subjetivas suelen expresar opiniones, mientras que las oraciones objetivas sólo indican hechos.

Niveles de análisis del sentimiento

Al realizar el análisis del sentimiento, la puntuación del análisis del sentimiento general se puede calcular de diferentes niveles:

Análisis de sentimiento a nivel documental (Document level sentiment analysis)

Esencialmente opera sobre un texto opuesto en la unidad de un documento. En esta clasificación de nivel de documento, se considera una sola revisión sobre un solo tema (Varghese et al, 2013). Esto se convierte en un desafío cuando se trata de declaraciones comparativas. Todas las frases en el documento pueden no ser relevantes o expresar ninguna opinión, por lo tanto, la clasificación objetivo / subjetivo es absolutamente imperativa.

Análisis de sentimiento a nivel de sentencia (Sentence level sentiment analysis)

El análisis del sentimiento de nivel de frase evalúa la polaridad de cada oración. La clasificación de la subjetividad a nivel de oraciones es útil porque la mayoría de los documentos contienen una mezcla de oraciones subjetivas y objetivas (Wiebe et al, 2005). En el caso de oraciones simples, una sola oración tiene una opinión única sobre una entidad. Las oraciones complejas no son deseables para el análisis del sentimiento de nivel de oración (Varghese et al, 2013).

Análisis del sentimiento de nivel de frase (Phrase level sentiment analysis)

La clasificación de sentimientos de nivel de frase es un enfoque más meticuloso para la minería de opinión. A veces se denomina análisis del sentimiento de nivel de aspecto. El contexto de la opinión depende de la redacción. Las palabras que parecen muy próximas entre sí se consideran en una frase (Varghese et al, 2013). Turney et al (2004) usaron frases en vez de palabras y aplicaron punto de información mutua en inglés PMI (Pointwise mutual information) entre palabras para etiquetar las polaridades.

Revisión de las Técnicas de Análisis de Sentimiento (Review of Sentiment Analysis Techniques)

Existen básicamente cuatro categorías principales para realizar el análisis del sentimiento:

I. Keyword spotting

II. Lexical affinity

III. Statistical methods

IV. Concept-level techniques

I. Keyword spotting / Palabra clave spotting

Es el enfoque más naïve (naïve approach). Clasifica el texto basado en la presencia de palabras de afecto. Estas palabras se suelen dar algunos valores sentimentales en una determinada anotación lingüística de esquemas.

Por lo tanto, la presencia de las palabras clave probablemente indicó la orientación de la polaridad sentimental. No es robusto a la negación y depende de características superficiales (Cambria, Erik et al, 2013).

II. Afinidad lexical (Lexical affinity)

Este enfoque es mucho más potente que la palabra clave spotting (keyword spotting). Además de detectar palabras de afecto, asigna a palabras arbitrarias una afinidad probable a emociones particulares (Cambria, Erik et al, 2013). Este enfoque no es robusto a la negación y a las oraciones con otro significado.

III. Métodos de estadística (Statistical methods)

Este método utiliza el enfoque de aprendizaje automático. Pang et al (2002) observaron que las técnicas de aprendizaje automático (machine learning techniques) (Naïve Bayes, Support Vector Machines y Maximum Entropy) superaron a las líneas de base producidas por humanos (human-produced baselines). Algunos estudios anteriores con algoritmos de aprendizaje automático en las revisiones de películas. Básicamente, al alimentar a un algoritmo de aprendizaje automático un gran corpus de formación de textos afectivamente anotados, el sistema podría no sólo aprender la valencia afectiva (affective valence) de palabras clave de afecto, sino también otras palabras clave arbitrarias (arbitrary keywords) (Cambria, Erik et al, 2013).

IV. Técnicas basadas en el concepto (Concept-based techniques)

Estos métodos emplean ontologías web o redes semánticas, ya que depende en gran medida de las bases de conocimiento. Superior a las técnicas puramente sintácticas, la técnica basada en concepto puede detectar expresiones de múltiples palabras (Cambria, Erik et al, 2013). Tienen la capacidad de analizar expresiones de varias palabras relacionadas con conceptos que expresamente transmiten emoción.

Aprendizaje de máquinas en el análisis de sentimientos (Machine Learning in Sentiment Analysis)

El método de aprendizaje automático (ML) aplicable al análisis del sentimiento pertenece principalmente a la clasificación supervisada. Básicamente incluye dos

conjuntos de datos: un entrenamiento y un conjunto de pruebas. El conjunto de entrenamiento es utilizado por un clasificador automático para conocer las características de diferenciación de los documentos, y un conjunto de pruebas se utiliza para validar el rendimiento del clasificador automático. Las técnicas de aprendizaje automático como Naïve Bayes (NB), máxima entropía (MaxEnt) y máquinas vectoriales de soporte (SVM) han logrado un gran éxito en la categorización de textos. Los otros métodos de aprendizaje de máquinas más conocidos en el área de procesamiento de lenguaje natural son K-Nearest neighborhood, ID3, C5, clasificador de centróides, clasificador de winnow y el modelo de N-gram (G.Vinodhini y Chandrasekaran, 2012).

La máquina de vectores de soporte (support vector machine o SVM) es un algoritmo de aprendizaje de última generación que ha demostrado ser eficaz en las tareas de categorización de texto y robusto en el espacio de grandes características (Saif Mohammad, 2013). Este enfoque fue propuesto por Vapnik (1995). Pang et al (2004) aplicaron SVM, MaxEnt y NB para clasificar las revisiones de películas como positivas o negativas. Descubrieron que el SVM funcionaba mejor que los clasificadores Naïve Bayes y Maximum Entropy. Ye et al (2009) su experimento de aplicación de support vector machine (SVM), Naïve Bayes y el modelo N-gram a los exámenes de destino muestran que el SVM supera a los demás. Yun-qing xia et al (2007) al proponer un marco de colocación unificado (UCF) se dio cuenta de que el clasificador SVM asigna etiquetas correctas a la mayoría de las opiniones verdaderas.

La situación para la precisión del análisis del sentimiento es similar a la precisión en la extracción de opinión. Se redujo en 0.172 cuando no se usó el clasificador SVM. Esto justifica la importancia del clasificador SVM en su método de minería de opinión. Songho Tan (2008) presenta un estudio empírico de la categorización del sentimiento en documentos chinos. Además de investigar los diversos métodos de selección de características, también comparó varios métodos de aprendizaje incluyendo clasificador de centroides, KNN, NB, clasificador de winnow y SVM en el corpus de sentimiento chino. Vio que SVM realizó mejor clasificación de sentimientos que el resto. Rui Xia et al (2011) realizaron un estudio comparativo de la eficacia de la técnica de conjunto para la clasificación de sentimientos. Emplearon tres algoritmos de clasificación de texto bien conocidos, a saber, Bayes naïve, MaxEnt y SVM como clasificadores base para cada uno de los conjuntos de características y también concluyeron que SVM funcionó aún mejor. Se han desarrollado múltiples variantes de SVM en las que se utiliza SVM multi-clase para la clasificación de sentimientos (Kaiquan Xu, 2011). En la mayoría de los estudios comparativos se encuentra que SVM supera a otros métodos de aprendizaje automático en la clasificación de sentimientos (Vinodhini y Chandrasekaran, 2012).

Naïve Bayes es un algoritmo de clasificación simple pero eficaz (Vinodhini y Chandrasekaran, 2012). Es una familia de clasificadores probabilísticos basada en la aplicación del teorema de Bayes con suposiciones sustanciales entre características. Melville et al (2009) utilizaron NB para presentar un marco unificado en el que utilizaron información léxica de fondo en términos de asociaciones de palabras. Rui Xia (2011) aplicó el algoritmo NB en conjuntos de características al hacer un estudio comparativo de la eficacia de la técnica de conjunto para la clasificación de

sentimientos. Este algoritmo es ampliamente utilizado para la clasificación de documentos (Songbo Tan, 2008). Su reconocimiento de la independencia de la palabra lo hace eficiente.

La clasificación de entropía máxima (MaxEnt) es un método de clasificación por aprendizaje automático que generaliza la regresión logística a problemas multiclase (generalizes logistic regression to multiclass problems). Los modelos MaxEnt son modelos basados en características (feature-based models) (A Go et al, 2009). La idea detrás de los modelos MaxEnt es que uno debe preferir los modelos más uniformes que satisfacen una restricción dada (Nigam et al, 2009). Dada una variable independiente, el modelo predice resultados posibles para una variable dependiente categóricamente distribuida. Pang et al (2002) informan que, de estudios previos, MaxEnt superó a Naïve Bayes algunas veces, pero no siempre. Esto se debe a que, a diferencia de Naïve Bayes, MaxEnt no hace suposiciones sobre las relaciones entre las características, por lo tanto, podría potencialmente desempeñarse mejor cuando los supuestos de independencia condicional no se cumplen. El experimento y el análisis de McDonald y Ryan (2009) dieron un apoyo significativo al método de peso de mezcla para el entrenamiento de modelos de entropía máxima condicional a gran escala con regularización de L2.

La idea detrás del algoritmo de clasificación centróide es simple y directa (Sangbo Tan, 2008). El clasificador de Rocchio es el enfoque del clasificador centróide que se aplica a la clasificación de texto utilizando *term frequency-inverse document frequency* (tf-idf). Erik et al (2013) aplicaron algoritmo de centroide para realizar análisis de sentimiento de dominio abierto. Ellos combinaron la mayor taxonomía existente del conocimiento común con la red semántica basada en el lenguaje natural del conocimiento de sentido común y luego aplicaron el escalamiento multidimensional en la base de conocimiento resultante.

Otros algoritmos de aprendizaje de máquina también se han utilizado, pero no ampliamente. K-Nearest Neighbor (KNN) es un típico ejemplo de clasificador que no construye una representación explícita, declarativa de la categoría, pero se basa en las etiquetas de categoría adjuntas a los documentos de formación similar al documento de prueba (Vinodhini y Chandrasekaran, 2012). Winnow es un método conducido erróneamente con esquema multiplicativo. Glance, Natalie et al (2005), mientras que derivar la inteligencia de marketing de la discusión en línea empíricamente ha encontrado en winnow un algoritmo de clasificación de documentos muy eficaz. Rudy Prabowo (2009) combina la clasificación basada en reglas (RBC), el aprendizaje supervisado y el aprendizaje automático en un nuevo método combinado y prueba el método en revisiones de películas. Los resultados muestran una mayor eficacia de la clasificación en términos de F1 mediana y macro.

Métodos de conjunto (Ensemble methods)

Los métodos de conjunto son algoritmos de aprendizaje que construyen un conjunto de clasificadores y luego clasifican nuevos puntos de datos tomando un voto (ponderado) de las predicciones (Dietterich, 2000). Es bastante obvio que los grupos

de personas a menudo pueden tomar mejores decisiones que los individuos. Lo mismo se cree en el aprendizaje automático. La idea detrás de los mecanismos del conjunto es explotar las características de varios aprendices independientes combinándolos para lograr un mejor desempeño que el mejor clasificador base (Fersini et al, 2014). El objetivo principal de ensamble es maximizar la precisión y diversidad individuales. En esencia, se esfuerza por lograr un rendimiento combinando de las opiniones de múltiples alumnos.

Se deben cumplir dos condiciones necesarias para lograr un buen conjunto: exactitud y predicción de la diversidad. (Fersini et al, 2014)

Razones del método de conjunto (Reasons for ensemble method)

Las tres razones fundamentales para considerar un enfoque de método de conjunto según Dietterich (2000) son:

1. Estadística - Al construir un conjunto el algoritmo puede promediar sus votos y reducir el riesgo de elegir el clasificador incorrecto.

2. Computacional - Un conjunto construido ejecutando la búsqueda local desde muchos puntos de partida diferentes puede proporcionar una mejor aproximación a la verdadera función desconocida que cualquiera de los clasificadores individuales.

3. Representación Cuando la función verdadera f no puede ser representada por ninguna de las hipótesis en H (las sumas ponderadas de las hipótesis tomadas de H podrían expandir el espacio).

Léxico de sentimientos (Sentiment Lexicon)

Los léxicos del sentimiento (Sentiment lexicons) son listas de palabras con asociaciones a sentimientos positivos y negativos (Saif Mohammad, 2013). La lista contiene palabras que se usan en el entrenamiento de los clasificadores de sentimiento.

Los idiomas que se han estudiado en su mayoría son el inglés y el chino (Vinodhini y Chandrasekaran, 2012). Ejemplos de léxicos de sentimientos son:

Sentiment140 lexicón un proyecto de la Universidad de Stanford (Go et al, 2009) es un conjunto de alrededor de 1,6 millones de tweets con emoticones positivos y negativos. Al aplicar el enfoque de aprendizaje automático, los tweets se clasifican como positivos o negativos basados en emoticones.

Respuesta a preguntas multi-perspectiva (*Multi-Perspective Question Answering*) (MPQA) Wiebe et al (2005) es un léxico con expresiones subjetivas.

SentiWordNet Lexicon Esuli et al (2006). Proporciona una extensión para WordNet que asocia las sintaxis con un valor emocional.

Bing et al (2004), mientras estudiaba la minería y la sumarizacion / resumen de los comentarios de los clientes construí el léxico Bing Liu. Es útil ya que incluye errores ortográficos, variantes morfológicas, jerga y marca social-media.

LWIC (Linguistic Inquiry y Word Count) evalúa las emociones en un texto empleando palabras pre-clasificadas en un diccionario.

Kouloumpis et al (2011) descubrieron que mientras analizaba el dominio de microblogging, aunque las características de parte de la voz (POS) no fueran particularmente útiles para el análisis de sentimientos, el léxico sentimental era en cierta medida útil.

Para este trabajo se buscaba clasificar el sentimiento en el idioma portugués brasileño, por eso se ha utilizar el léxico portugués de Portugal llamado Sentilex PT.

Sentilex PT

El SentiLex-PT es léxico de sentimiento para el portugués, que consta de 7.014 consignas y 82,347 formas conjugadas. En concreto, el léxico describe:

- 4,779 (16,863), adjetivos
- 1.081 (1.280) nombres
- 489 (29.504) y los verbos
- 666 (34.700) expresiones idiomáticas

Las entradas de léxico corresponden a predicadores humanos, es decir predicadores que se construyen con los nombres humanos, procedentes de diferentes recursos públicos (léxicos y corpus).

La sensación de atributos descritos en cada entrada son los siguientes: el objetivo de la sensación, la polaridad del predicador y el método de asignación de polaridad.

La polaridad de la información asociada entradas es en la mayoría de los casos asignados de forma manual. Algunas entradas de adjetivos, sin embargo, se ordenan automáticamente por una herramienta (JALC) desarrollado por el equipo del proyecto para este fin. Las formas conjugadas de los verbos y expresiones idiomáticas, así como los respectivos atributos morfológicos fueron extraídos de forma semiautomática de LABEL-LEX-sw, un léxico de palabras sencillas a disposición de los portugueses, que fue desarrollado por Ranchhod et al. (1999), en la etiqueta http://label.ist.utl.pt/pt/].

Extracción de características (Feature Extraction)

Para el análisis del sentimiento, la extracción de características es una de las tareas más complejas, ya que requiere el uso de procesamiento de lenguaje natural para identificar automáticamente las características de las opiniones en análisis. También refleja en gran medida la eficiencia de la tarea de análisis del sentimiento general.

Retos en el análisis del sentimiento (Challenges in sentiment analysis)

El análisis del sentimiento es un campo de investigación bastante nuevo que sigue ganando terreno. Algunos desafíos observados son:

Extracción de entidad nombrada (Named Entity Extraction)

Las entidades nombradas son frases nominales definidas que se refieren a tipos específicos de individuos, como marcas, productos, organizaciones, etc. El objetivo de la extracción de entidades nombradas es identificar todas las menciones textuales de las entidades nombradas en una pieza de texto (Pak et al, 2010).

Falta de ortografía y ortografía creativa (Misspelling and creative spelling)

Los usuarios de redes sociales usualmente no tienen un formato estandarizado para expresar sus opiniones. Los usuarios a menudo deletrean mal las palabras o usan algunas dimensiones artísticas para escribir sus palabras. Estas palabras suelen crear problemas al tratar de analizarlos.

Argot y lenguaje informal (Slang and informal language)

Muchas generaciones y grupos de personas usan diferentes lenguajes informales con respecto a varios contextos. En Brasil, la diferencia en la diversidad y por un concepto de reconocimiento, muchas personas, especialmente los jóvenes, utilizan un lenguaje informal. Estas formas de escribir difieren de una región a otra, visto que Brasil es un país de dimensiones continentales con mucha diversidad.

Extracción de información (Information Extraction)

Como la información de las fuentes de datos viene de muchas formas y tamaños, la complejidad del lenguaje natural puede dificultar el acceso a la información en el texto de opinión (Pak et al., 2010). A diferencia de los seres humanos, las máquinas no pueden identificar la información relevante tan fácilmente como lo hacen los humanos. Las herramientas de PNL todavía están tratando de construir una representación de propósito general del significado del texto sin restricciones.

Resolución de co-referencia (Co-reference Resolution)

Suele ocurrir en el aspecto de análisis de sentimiento y nivel de entidad. Cuando no se descubren las palabras co-referentes, el análisis del sentimiento efectivo no puede llevarse a cabo (Pak et al, 2010). Los dos tipos diferentes de opiniones son la opinión directa donde las emociones se expresan explícitamente con un objetivo, mientras que la opinión comparativa compara varios objetivos. Los textos comparativos pueden contener co-referencias que deben ser efectivamente resueltas para corregir los resultados.

Extracción de Relación (Relation Extraction)

Esta es un área importante de investigación en el PNL (Procesamiento de Lenguajes naturales) o en ingles NLP (Natural Language Processing). La extracción de la relación es la tarea de encontrar la relación sintáctica entre palabras en una oración (Pak et al, 2010). La semántica de una oración se realiza conociendo las dependencias de las palabras.

Dependencia del dominio (Domain Dependency)

Muchos clasificadores entrenados para clasificar las polaridades de opinión en un dominio específico y en el que realizarán su tarea relativamente bien, pueden producir resultados más bien pobres si se aplican en un dominio diferente. Woller, Martin et al (2013) estudiaron cómo realizar el análisis del sentimiento de independencia de dominio para las revisiones de películas. Esencialmente esto sigue siendo un reto no resuelto en el análisis del sentimiento.

10. Arquitectura Conceptual propuesta

El proyecto ha sido planeado en 4 capas, donde se busca recoger, almacenar, utilizar técnicas de análisis de sentimiento y visualizar los datos.

1) Colectar los datos de distintas fuentes de datos	2) Almacenar los datos en una base de datos relacional (EDW) y en un entorno Big Data (Data Lake)	3) Análisis de sentimiento	4) Visualizar los datos en una herramienta de Inteligencia de Negocio

Las herramientas deben ser de código abierto, se busca crear una arquitectura capaz de escalar para gestionar muchísimos volúmenes de datos. También se pide que sea una solución de bajo coste para ser implantada en pequeñas, medianas y grandes empresas.

1) Colectar los datos de distintas fuentes de datos

El sistema tiene que ser capaz de integrarse con muchísimas fuentes de datos como Twitter, Google Search, Instagram, Youtube, Facebook y Blogs.

En este trabajo se ha permitido la implementación solo de 1 o 2 fuentes de datos por cuestión de tiempo.

2) Almacenamiento de los datos

Se busca un sistema capaz de almacenar los datos con una base de datos relacional (EDW), NoSQL o con un entorno Big Data (Data Lake).

3) Análisis de sentimiento

Una parte importante es crear un subsistema capaz del análisis de sentimiento. Los datos pueden ser procesados en modo *batch* o tiempo real.

4) Visualización de los datos

La visualización de los datos tiene que ser por medio de una herramienta de inteligencia de negocio. Existen muchas ventajas en utilizar una herramienta de BI como pueden ser: seguridad de los acceso a los datos, facilidad en el mantenimiento de los *KPI's (Key Performance Indicator)* y escalabilidad.

La herramienta de BI debe ser una herramienta web donde los usuarios puedan acceder por medio de un usuario y una clave de acceso, desde una estación de trabajo.

En la imagen de abajo se puede ver la arquitectura conceptual del sistema, donde se leen los datos de distintas fuentes de datos representados en el dibujo como fuente 1, fuente 2, fuente 3 y fuente 4. Se debe ser capaz de añadir más fuentes de datos en el futuro, el enfoque inicial del proyecto es colectar datos de Twitter y Google Search, pero en el futuro se busca añadir otras fuentes de datos para enriquecer el modelo global de análisis de sentimiento.

El proceso de recolección, almacenamiento y procesamiento de los datos debe ser hecho de forma automática, si posible utilizando una herramienta de ETL de mercado. La periodicidad de la lectura de los datos y todo el proceso de ETC (extracción, transformación y carga) de los datos debe ser parametrizado por el usuario final, es decir, debe ser algo sencillo para cambiar.

El subsistema de minería de textos y análisis de sentimiento puede ser ejecutado en tiempo real o en modo batch con un retraso de 1 día o más, pero no más de una semana. Se permite utilizar cualquier herramienta o entorno para esta tarea mientras sea de código abierto y libre su utilización, se permite utilizar R, Python o Weka.

El usuario final necesita una interface sencilla para las visualizaciones, se debe crear uno o más cuadros de mando integral para dar al usuario una visión del negocio.

Se busca una herramienta capaz de ayudar el usuario a gestionar la crisis y ser capaz de tomar decisiones basadas en los datos provenientes del nuevo sistema. Se valora también la utilización de una herramienta OLAP, capaz de permitir al usuario final crear sus propios análisis basado en los datos (los datos pueden estar agregados o no). Se desea la capacidad de analizar grandes volúmenes de datos con una herramienta OLAP.

Figura 5 – Arquitectura Conceptual

11. Configuraciones prácticas de la arquitectura. Aplicación a un caso real

La aplicación práctica en un proyecto real ha sido desarrollada en dos versiones: la primera, una versión que llamaremos Data Science y una segunda versión, que llamaremos Big Data Science (donde se han probado distintas tecnologías de Big Data).

11.1 Primera versión

En esta versión se ha dado enfoque en Data Science y se ha utilizado solamente los datos de Twitter. El trabajo ha consistido en crear una herramienta capaz de analizar los mensajes de twitter.

Todas las informaciones han sido extraídas de Twitter a través de un programa desarrollado en Python, R y Java.

La plataforma de Twitter permite realizar consultas a sus datos como se puede ver en la figura abajo.

Búsqueda: Tweets – From Everyone – Near you.

Con el enlace abajo se puede hacer una búsqueda en Twitter. https://twitter.com/search?f=tweets&vertical=news&q=%23bigdata&near=me&src=typd

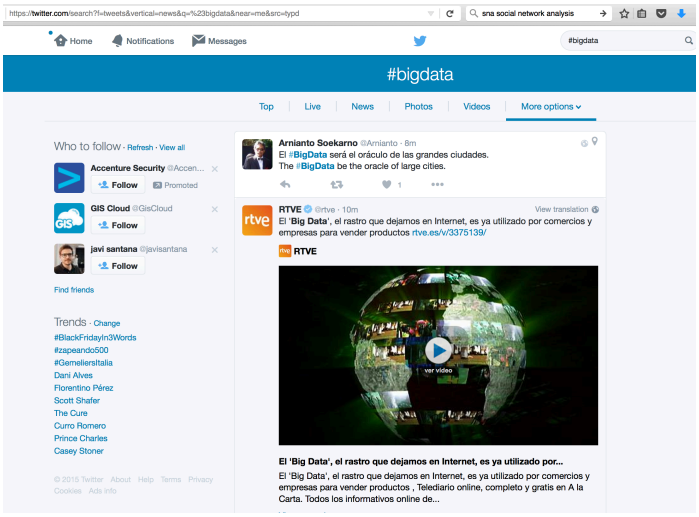


Figura 6 – Búsqueda en Twitter

Un mensaje de Twitter contiene 140 caracteres, pero para cada mensaje de Twitter generada se almacena muchos metadatos que pueden ser utilizados para minería de datos.

Abajo se encuentra la imagen llamada “Map of a Twitter Status Object” creada por Raffi Krikorian el 18 de abril de 2010 donde se puede ver la cantidad de información rica para minería de datos.

Como se puede ver, no solo el texto es generado, también informaciones como: identificador único del usuario, sus informaciones de perfil (nombre, nombre de tela, descripción, localización, página web), su país, muchas veces su localización exacta desde donde ha publicado (esto solo cuando la persona permite enviar su latitud y longitud de forma consciente), cantidad de seguidores, idioma del mensaje y hora de su publicación.

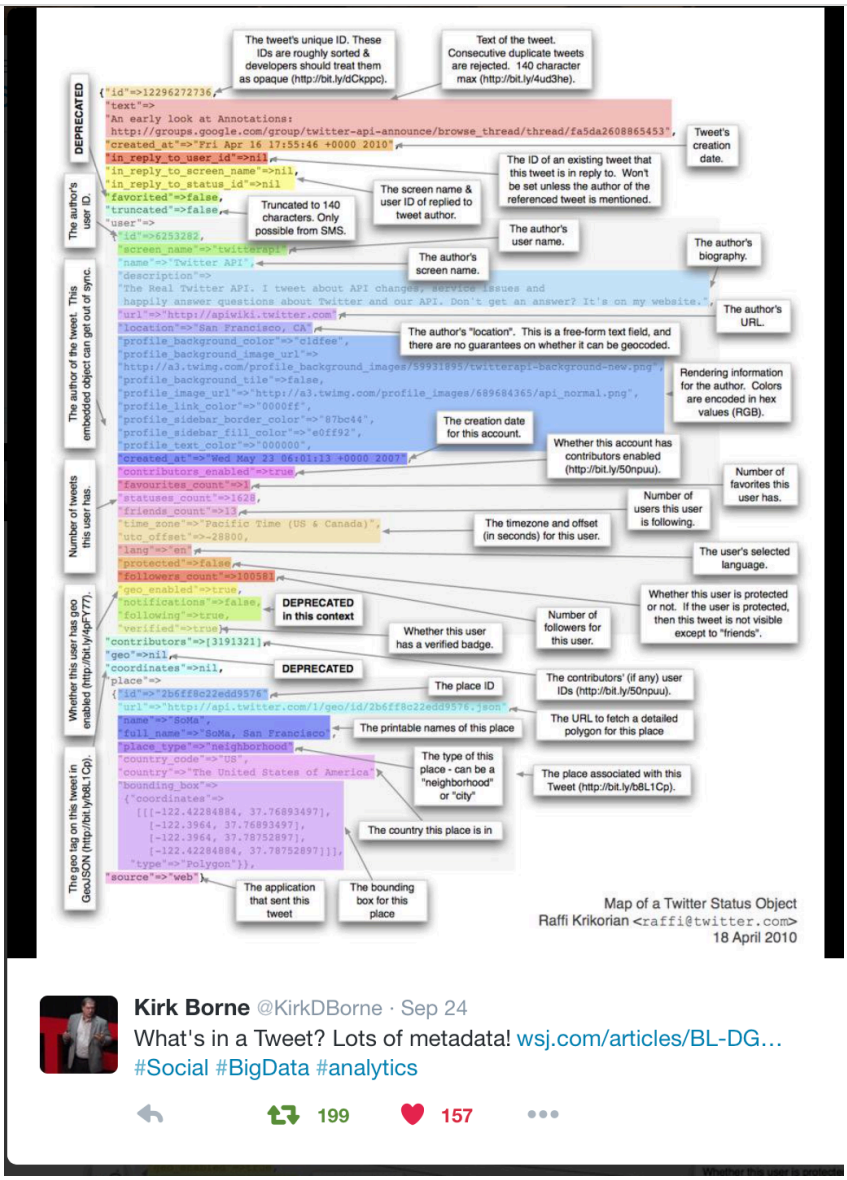


Figura 7 – "Map of a Twitter Status Object"

Extracción de los datos de Twitter.

El primer paso del trabajo ha sido crear una forma de extraer los datos de Twitter.

Todos los datos extraídos han sido publicados en https://github.com/caiomsouza/TwitterRawData/releases/tag/DITRD-v1.0.0 y pueden ser utilizados de forma libre.

El principio de esta etapa puede parecer una tarea simple, pero ha sido una fase donde he invertido mucho tiempo e incluso no he encontrado la solución que acepto como la ideal para mis necesidades, pero por razones de tiempo he tenido que darme por satisfecho con una de las herramientas y he avanzado con el proyecto (en la

versión Big Data Science del proyecto se ha encontrado otras formas mejores que se explicarán después).

Actualmente existen ya desarrolladas y disponibles de forma gratuita y abierta algunas APIs en *Python, Java, R* y otros lenguajes donde el primer paso es elegir cuál API se desea utilizar, después aprender el funcionamiento de la API y entonces extraer los datos.

Como conozco Java, Python y R he probado distintas formas de extraer datos de Twitter con Java, Python, R, inclusivo utilizando Apache Spark con Java lo que me ha dado resultado en diversas herramientas de extracción de datos creadas, pero ambas haciendo lo mismo.

Las distintas APIs de Twitter suelen tener la misma capacidad de extraer datos y no suelen tener muchas diferencias, por lo que no quiero recomendar ninguna en particular porque no he visto muchas diferencias en las que he probado.

Me ha gustado las pruebas que he hecho con R, con Python y con Java utilizando también Apache Spark.

Algo muy importante sobre la API de Twitter es la obligatoriedad de crear una cuenta en Twitter y después un aplicativo capaz de utilizar la API de Twitter.

Con este aplicativo se puede extraer datos de Twitter, pero hay un límite de mensajes que se pueden extraer y todo eso se debe tener en consideración a la hora de crear su solución de extracción de datos.

En este trabajo he utilizado la API de Streaming de datos de twitter donde es permitido captar los mensajes que se están generando en este momento.

Se puede captar los mensajes en tiempo real o hacer una consulta en la base de datos de Twitter para una determinada palabra clave, pero esta segunda opción tiene muchas limitaciones.

Para más información de cómo crear una cuenta en Twitter, un aplicativo que permita utilizar la API de Twitter y para conocer mejor la API de Stream y la de consultas se recomienda entrar en www.twitter.com y leer la documentación de Twitter.

Abajo podemos ver el código mencionado para extraer los datos de Twitter utilizando R:

r_sentiment_analysis_sentilex-pt01.R

```
setwd("~/git/caiomsouza/src/r-script")

# http://thinktostart.com/sentiment-analysis-on-twitter/
# https://jeffreybreen.wordpress.com/2011/07/04/twitter-text-mining-r-slides/
#http://www.r-bloggers.com/mining-twitter-for-consumer-attitudes-towards-hotels/

#library(devtools)
#install_github("twitteR", username="geoffjentry")
#install.packages("ROAuth")
```

```
#install.packages("RCurl")
#install.packages("bitops")
#install.packages("digest")
#install.packages("rjson")
library(twitteR)
library(plyr)
library(ROAuth)
library(bitops)
library(digest)
library(rjson)


#twitter user @caiomsouza
api_key <- " Cambiar_Para_Sus_Datos "
api_secret <- " Cambiar_Para_Sus_Datos "
access_token <- " Cambiar_Para_Sus_Datos "
access_token_secret <- "Cambiar_Para_Sus_Datos"

setup_twitter_oauth(api_key,api_secret,access_token,access_token_secret)
```

En esta parte se pide escoger entre dos opciones, 1 o 2. Ambas opciones me han funcionado sin ningún problema.

```
> setup_twitter_oauth(api_key,api_secret,access_token,access_token_secret)
[1] "Using direct authentication"
Use a local file to cache OAuth access credentials between R sessions?
1: Yes
2: No

Selection:
```

El próximo paso es llamar la función ***searchTwitter*** con los dos parámetros: palabra clave y cantidad de mensajes a extraer.

```
tweets = searchTwitter("dilma", n=2000)
```

El código R que se ve abajo es para contar la cantidad de twitter colectado.

```
length(tweets)
```

Resultado:

```
> length(tweets)
[1] 2000
```

Si se desea se puede mirar los mensajes que han sido colectados y se encuentran en tweets.

```
Tweets
```

Resultado:

```
[[1669]]
[1] "rralves: E o que a governANTA fez? Retirou o menino de circulação para ser esquecido pela imprensa e pela justiça! Não... https://t.co/yoLMNwBymL"

[[1670]]
```

[1] "zaanganeles: RT @o_antagonista: Dilma em alerta: \"Ninguém sabe o que Otávio Azevedo pode falar\" https://t.co/gaIJu7xDyP https://t.co/qZIjSWnP4L"

[[1671]]
[1] "diegorrr_: oq a Dilma tá fazendo com o Twitter? Alguém da um jeito nessa mulher urgente!!"

[[1672]]
[1] "joalmagalhaes: Viva Juiz Moro\nDELAÇÃO DA ANDRADE GUTIERREZ – Estão em pânico Dilma, Lula, Lulinha e o PMDB do Rio https://t.co/8SPVpeAUhI vía @veja"

[[1673]]
[1] "heauxn0ire: Presidenta deve tá fritando por lá.\nDilma aproveita folga de Carnaval para pedalar em Porto Alegre https://t.co/6U3zaAvbC5"

Existe mucho material disponible en internet sobre otras formas de extraer twittees, pero para mis objetivos he me he quedado con la opción mostrada anteriormente.

En mi caso, no voy trabajar con una gran cantidad de mensajes en esta primera versión, la idea es, de forma manual, utilizando la herramienta R Studio, ejecutar el código de la extracción y recoger 2000 twittees para hacer un pequeño estudio del sentimiento general de las personas que están hablando de determinada palabra clave.

He probado con otras palabras claves de mi interés personal y estas palabras no han llegado a tener ni 2000 mensajes. Para pequeñas y medianas empresas esta solución es suficiente, porque ellas no llegan a tener mucha gente hablando sobre sus productos, marca, etc.

En el caso de la palabra clave “dilma” se pueden encontrar muchísimos twittees que son creados cada día.

Durante algunos días he recogido más de 1 millón de mensajes con la palabra clave “dilma”, esto no lo he hecho con R y sí con una herramienta hecha en Python y después con una herramienta de extraer datos hecha en Java y con Apache Spark.

En el caso de la palabra “dilma” es recomendable pensar en una infraestructura de Big Data utilizando Apache Kaftka, Apache Spark y Apache Hive para la tarea de ingestión y extracción de datos y almacenaje los datos.

Para un mejor entendimiento de esta solución recomendable es necesario explicar algo sobre los componentes mencionados arriba.

Apache Spark es el motor (engine) más rápido actualmente para el procesamiento de datos en grandes volúmenes.
http://spark.apache.org/

Apache Kaftka es un sistema de mensajería abierto y muy utilizado. Ha sido originalmente desarrollado por LinkedIn.
http://kafka.apache.org/

Apache Hive es una infraestructura para almacenar datos (data warehouse) construida sobre *Hadoop* para proporcionar la suma de datos, consultas y análisis de bases de datos muy grandes en almacenamiento distribuido. Ha sido inicialmente desarrollada por *Facebook*, pero actualmente es utilizada y desarrollada por otras empresas como *Netflix*. Amazon mantiene un fork del proyecto *Apache Hive* que ha sido incluida en su producto *Amazon Elastic MapReduce* en *Amazon Web Services*.

https://hive.apache.org/
http://infolab.stanford.edu/~ragho/hive-icde2010.pdf

Preparación de los datos extraídos de los mensajes de Twitter

El código de abajo en R es necesario para separar solo el texto de cada mensaje.

```
Tweets.text = laply(tweets,function(t)t$getText())
Tweets.text
```

Se ha utilizado la siguiente función para limpiar los datos.

```
# Función para limpiar los datos
clean.text <- function(some_txt)
{
  some_txt = gsub("&amp", "", some_txt)
  some_txt = gsub("(RT|via)((?:\b\\W*@\\w+)+)", "", some_txt)
  some_txt = gsub("@\\w+", "", some_txt)
  some_txt = gsub("[[:punct:]]", "", some_txt)
  some_txt = gsub("[[:digit:]]", "", some_txt)
  some_txt = gsub("http\\w+", "", some_txt)
  some_txt = gsub("[ t]{2,}", "", some_txt)
  some_txt = gsub("^\\s+|\\s+$", "", some_txt)

  # define "tolower error handling" function
  try.tolower = function(x)
  {
    y = NA
    try_error = tryCatch(tolower(x), error=function(e) e)
    if (!inherits(try_error, "error"))
      y = tolower(x)
    return(y)
  }
  some_txt = sapply(some_txt, try.tolower)
  some_txt = some_txt[some_txt != ""]
  names(some_txt) = NULL
  return(some_txt)
}
```

```
clean_text = clean.text(Tweets.text)
```

Cargar en memoria el Lexicón (Diccionario de Palabras positivas y Negativas).

Una parte muy importante del análisis de sentimiento / opinión es encontrar o crear un lexicón con palabras que sean positivas y negativas.

En este trabajo es posible utilizar 3 lexicón, siendo uno en inglés y dos en portugués.

No ha sido posible encontrar un lexicón para el portugués de Brasil y por esta razón se ha decidido usar el lexicón llamado SentiLex versión 1 y 2 del portugués de Portugal. Las tres opciones de lexicón utilizadas en este trabajo son muy conocidas por la comunidad científica y muy utilizados para trabajos de análisis de sentimiento.

Para este trabajo ha sido necesario crear otro programa en Python capaz de leer el SentiLex y preparar los diccionarios en el formato solicitado por R. Los códigos están disponibles en github.com/caiomsouza.

Después de ser preparados los diccionarios la próxima etapa en R es:

```
#3. Cargar en memoria el Lexicon (Diccionario de Palabras positivas y Negativas).

# en = Wordbank en ingles | pt01 = Wordbank en portugues con SentiLex-PT01 | pt02 = Wordbank en portugues con SentiLex-PT02
version <- "pt02";

if (version == "en") {

  # Wordbanks from https://github.com/mjhea0/twitter-sentiment-analysis/tree/master/wordbanks
  pos = scan('wordbanks/positive-words.txt', what='character', comment.char=';')
  neg = scan('wordbanks/negative-words.txt', what='character', comment.char=';')
  head(pos)
  head(neg)

} else if(version == "pt01"){

  # SentiLex-PT01
  pos = scan('/Users/caiomsouza/git/Bitbucket/u-tad/final-project/src/r-script/SentiLex-PT01/pos-pt01.txt', what='character', comment.char=';')
  neg = scan('/Users/caiomsouza/git/Bitbucket/u-tad/final-project/src/r-script/SentiLex-PT01/neg-pt01.txt', what='character', comment.char=';')
  head(pos,20)
  head(neg,20)


} else if (version == "pt02") {

  # SentiLex-PT02
  pos = scan('/Users/caiomsouza/git/Bitbucket/u-tad/final-project/src/r-script/SentiLex-PT02/pos.txt', what='character', comment.char=';')
  neg = scan('/Users/caiomsouza/git/Bitbucket/u-tad/final-project/src/r-script/SentiLex-PT02/neg.txt', what='character', comment.char=';')
  head(pos,20)
  head(neg,20)

}
```

Análisis de sentimiento

Para determinar el sentimiento de un tweet es posible utilizar dos técnicas distintas: i) método basado en lexicón, y ii) método basado en aprendizaje de máquina (*machine learning*). El método basado en aprendizaje supervisado utiliza un clasificador y un corpus. El método basado en lexicón es un aprendizaje no supervisado o semántico donde se utiliza un diccionario de palabras positivas y negativas.

Se conoce, por medio de estudios anteriores de otros investigadores de la comunidad académica mundial, que utilizar un clasificador basado en SVM y Naive Bayes supera el rendimiento del método basado en lexicón, pero la ventaja del método basado en lexicón es no tener que crear corpus para cada dominio, ahorrando tiempo y creando un método más genérico para distintos dominios.

Es posible mejorar muchísimo el rendimiento con un método ensamblado donde se utiliza una puntuación de sentimiento basado en el método de lexicón como una variable para el método de aprendizaje de máquina con SVM o Naive Bayes.

Actualmente se pueden tener resultados positivos con los dos métodos. La precisión obtenida con clasificadores multidominio (método basado en lexicón) son de 70 - 75% y la precisión con clasificadores específicos a partir del 80% (método basado en aprendizaje de máquina y un corpus).

El principal reto actualmente en la investigación del análisis de sentimiento es tratar la subjetividad. Esto es un gran reto también para las personas, y para las máquinas aun más grande.

Este trabajo busca encontrar un método automático de clasificar el sentimiento de los tweets con la máxima precisión posible, pues con la gran cantidad de tweets para determinados asuntos es imposible hacer este trabajo manualmente. También es importante aclarar que el enfoque es hacerlo para el idioma portugués.

La solución encontrada ha sido aplicar la función score.sentiment para extraer el sentimiento del texto.

Esta función es genérica y se puede aplicar para cualquier texto en cualquier idioma (lo que hace que sea muy interesante debido a la flexibilidad encontrada).

Pero, es muy importante aclarar que no es la mejor solución existente en el mundo y, además, tampoco he podido comparar los resultados de este algoritmo con otros existentes ahora mismo.

En esta primera versión no ha sido posible hacer una investigación profunda sobre el estado del arte de análisis de sentimiento en el mundo y tampoco el estado del arte de análisis con mensajes en el idioma portugués-brasileño.

Muchas veces, las empresas o instituciones públicas no hacen ningún tipo de análisis de lo que se habla en el Twitter, o muchas veces lo hacen de forma muy manual.

Este trabajo lo que busca es encontrar una forma de automatizar este proceso y dejarlo lo más genérico posible, aunque haciendo esto tenga que sacrificar la calidad de la predicción del sentimiento.

Para proyectos donde el objetivo es tener una mejor predicción, lo recomendable es crear un Corpus específico para el dominio de estudio y aplicar un algoritmo de aprendizaje supervisado donde es posible conseguir mejores resultados, pero aun así deja la solución muy personalizada y poco genérica.

La función que se ve abajo, lo que hace, es separar en palabras, coger cada palabra y mirar en el Lexicón si es una palabra positiva o negativa.

Se cuenta la cantidad de palabras positivas y la cantidad de palabras negativas y se hace una cuenta simple para intentar predecir el sentimiento del texto.

La fórmula final en R ha quedado:

score = sum(pos.matches) - sum(neg.matches)

Donde la puntuación (score) es disminuir la cantidad de la suma de palabras positivas encontradas en el texto con la suma de palabras negativas encontradas en el texto.

Los valores son negativos, positivos o cero.

Se puede visualizar los datos con los valores numéricos o clasificar los resultados en positivos, negativos o neutros aplicando una regla que puede ser definida por el usuario de la herramienta.

Se puede decir que valores más grandes que 1 o 2 son positivos, de -1 para bajo negativos y neutros lo que no es positivo o negativo, pero la predicción puede no ser la mejor.

Función para hacer la analice de sentimiento

```
score.sentiment = function(sentences, pos.words, neg.words, .progress='none')
{
  require(plyr)
  require(stringr)

  # we got a vector of sentences. plyr will handle a list
  # or a vector as an "l" for us
  # we want a simple array ("a") of scores back, so we use
  # "l" + "a" + "ply" = "laply":
  scores = laply(sentences, function(sentence, pos.words, neg.words) {

    # clean up sentences with R's regex-driven global substitute, gsub():
    sentence = gsub('[[:punct:]]', '', sentence)
    sentence = gsub('[[:cntrl:]]', '', sentence)
    sentence = gsub('\\d+', '', sentence)
    # and convert to lower case:
    sentence = tolower(sentence)

    # split into words. str_split is in the stringr package
    word.list = str_split(sentence, '\\s+')
    # sometimes a list() is one level of hierarchy too much
    words = unlist(word.list)

    # compare our words to the dictionaries of positive & negative terms
    pos.matches = match(words, pos.words)
    neg.matches = match(words, neg.words)

    # match() returns the position of the matched term or NA
    # we just want a TRUE/FALSE:
    pos.matches = !is.na(pos.matches)
    neg.matches = !is.na(neg.matches)

    # and conveniently enough, TRUE/FALSE will be treated as 1/0 by sum():
    score = sum(pos.matches) - sum(neg.matches)

    return(score)
  }, pos.words, neg.words, .progress=.progress )

  scores.df = data.frame(score=scores, text=sentences)
  return(scores.df)
}
```

El código de abajo ejecuta la función y hace el cálculo del sentimiento.

analysis = score.sentiment(clean_text, pos, neg)

La visualización de los datos

La visualización de los datos es la parte más interesante para el usuario de la herramienta y también la parte donde se puede invertir muchísimo tiempo para generar distintos tipos de visualización de los datos.

Por una limitación de tiempo no se ha podido llegar en todas las visualizaciones idealizadas en el momento del proyecto, pero sí se ha podido llegar a algunas visualizaciones interesantes y capaces de permitir al usuario sacar algunas conclusiones de los datos.

Las visualizaciones han sido divididas en dos:

1) Visualizaciones utilizando la herramienta R;
2) Visualizaciones con la herramienta de BI (Business Intelligence) llamada Pentaho;

1) Visualizaciones utilizando la herramienta R

La herramienta R posibilita maneras de visualizar los datos y sacar conclusiones muy interesantes.

Abajo ejecutaremos algunos comandos en R capaces de analizar los datos.

```
# Visualización
table(analysis$score)
mean(analysis$score)
hist(analysis$score)
colnames(analysis)
View(analysis)
```

table(analysis$score)

```
> table(analysis$score)

 -6  -5  -4  -3  -2  -1   0   1   2
  2  15 117 108 273 756 572 128  17
```

mean(analysis$score)

```
> mean(analysis$score)
[1] -1.015594
```

Se puede ver que la media es negativa, en general se “puede” decir que hay más palabras negativas que positivas.

Para tener claro que estos valores son la realidad de la muestra de 2000 twittees, se recomienda al interesado que se debería mirar cada uno de los mensajes y hacer una clasificación manual de positivo, negativo y neutro

Por razones de tiempo y deseo, no se ha planteado en este trabajo hacer esta validación, lo que deja los resultados conseguidos más difíciles de ser interpretados.

hist(analysis$score)

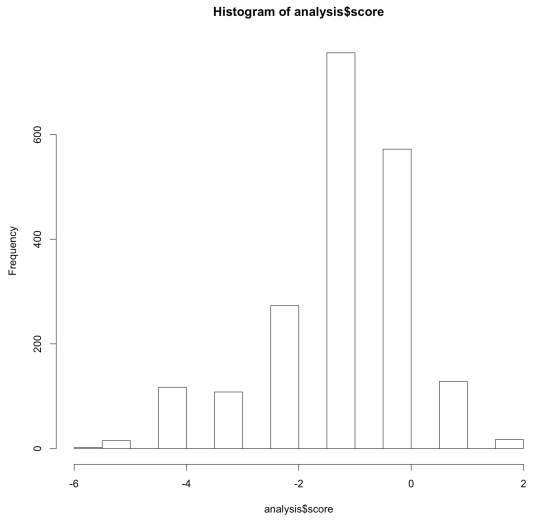


Figura 8 - Histograma de la Puntuación (Score)

En histograma arriba se puede ver que en los 2000 mensajes la puntuación (score) tiene un rango de -6 a 2, que hay más valores negativos o sea hay mas sentimiento negativo que positivo.

colnames(analysis)
> colnames(analysis)
[1] "score" "text"

View(analysis)

Se puede ver el texto y la puntuación (score).

Filter

	score	text
1	-2	rtno galo da madrugada não faltaram brincadeiras co...
2	0	rtdilma rousseff renuncia em agosto de
3	-1	rtgilmar mendes sobre janoadvogado de dilma
4	-1	rtate o reiespanha se chocou c gastos d dilma
5	0	rtdilma é a pior presidente da história da humanidade...
6	-2	no galo da madrugada não faltaram brincadeiras com...
7	-2	rtempara pagar as pedaladas e evitar o impeachmend...
8	-2	rtpuxa vida donaperdeu completamente juízo se for v...
9	0	rtisto é dilmapreso delcídio continua sendo líder de d...
10	1	rtse o twier começar a ordenarweets por popularidad...
11	-4	rtibope mostra rejeição devastadora ao governo dilm...
12	-2	riptwier a dilma que estãoe pagando para fazer isso
13	-1	vídeo impressionante mostra plateia virando de costa...
14	-1	hora de sacar dilma os petralhas e o pmdb do poder ...
15	0	uma ousadia de dilmaeditorialo estado de s paulovia
16	-3	enquanto o brasil batia panela contra o pt durante exi...

Showing 1 to 16 of 1,988 entries

Console ~/git/Bitbucket/u-tad/final-project/src/r-script/

```
> analysis = score.sentiment(clean_text, pos, neg)
Loading required package: stringr
> table(analysis$score)

 -6  -5  -4  -3  -2  -1   0   1   2
  2  15 117 108 273 756 572 128  17
> mean(analysis$score)
[1] -1.015594
> hist(analysis$score)
> colnames(analysis)
[1] "score" "text"
> 2+15+117+108+273+756+572+128+17
[1] 1988
> colnames(analysis)
[1] "score" "text"
>
> View(analysis)
>
```

Nube de Palabras

Otra técnica muy interesante en la minería de datos es lo que se llama la nube de palabras.

Utilizando la herramienta R y el código abajo mostrado se ha creado la nube de palabras.

```r
# Nube de Palabras
#install.packages(c("wordcloud","tm"),repos="http://cran.r-project.org")
library(tm)
library(wordcloud)
require(plyr)

tweet_corpus = Corpus(VectorSource(clean_text))
tdm = TermDocumentMatrix(tweet_corpus,
                         control = list(removePunctuation = TRUE,stopwords = c("machine", "learning", stopwords("english")),
                                        removeNumbers = TRUE, tolower = TRUE))
m = as.matrix(tdm) #we define tdm as matrix
word_freqs = sort(rowSums(m), decreasing=TRUE) #now we get the word orders in decreasing order
dm = data.frame(word=names(word_freqs), freq=word_freqs) #we create our data set
wordcloud(dm$word, dm$freq, random.order=FALSE, colors=brewer.pal(8, "Dark2")) #and we visualize our data
png("~/git/Bitbucket/u-tad/final-project/src/r-script/CloudDilma6Feb16.png", width=12, height=8, units="in", res=300)
wordcloud(dm$word, dm$freq, random.order=FALSE, colors=brewer.pal(8, "Dark2"))
dev.off()
```

Los datos son de una muestra recogida en el día 6 de febrero de 2016 cuando los brasileños están en el periodo de Carnaval (el Carnaval es una fiesta popular en Brasil celebrada todos los años en todo el país, es una fiesta conocida internacionalmente y tiene un gran impacto en el país).

Durante los días festivos de carnaval las personas suelen dejar de hablar de política, de sus problemas personales, de los problemas del país e intentan festejar de todas las formas posibles. Hay muchas personas no aficionadas por el carnaval que en estos días se quedan en casa con la familia o aprovechan para descansar de sus labores.

Es muy interesante mirar en la imagen de abajo la nube de palabras y ver el sentimiento negativo que existe en las palabras (muchas de ellas asociadas a personas y eventos relacionados con la corrupción o la insatisfacción del pueblo).

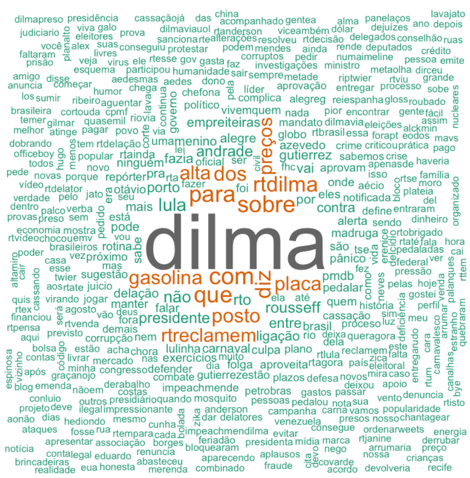


Figura 9 - Nube de Palabras

Visualizaciones con la herramienta de BI (Business Intelligence) llamada Pentaho;

La herramienta Pentaho ha sido utilizada para la creación de un cuadro de mando.

Los datos han sido almacenados en MySQL para posibilitar las consultas SQL ejecutadas por el cuadro de mando.

En la imagen de abajo se pueden ver con detalle las variables disponibles en un twitter que han sido almacenadas en el MySQL.

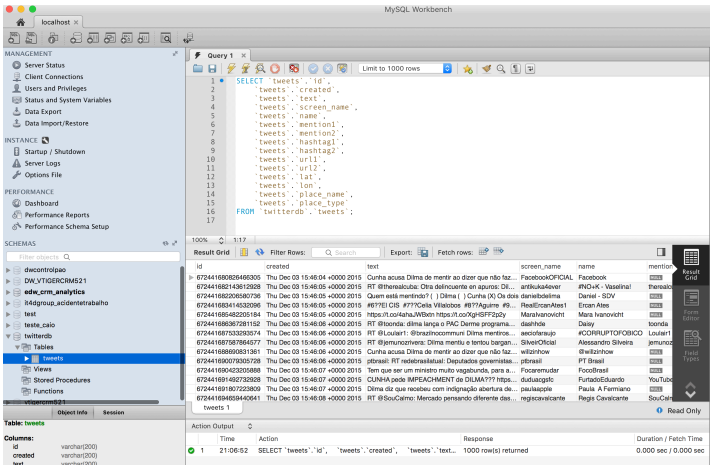


Figura 10 - Base de datos

Como se puede ver en la imagen abajo, la visualización de los datos se puede hacer también por medio de cuadros de mandos hechos con la herramienta de BI Pentaho.

Se han creado dos cuadros de mando para ayudar en la visualización de los datos.

Figura 11 - Cuadro de Mando creado con CTools/Pentaho

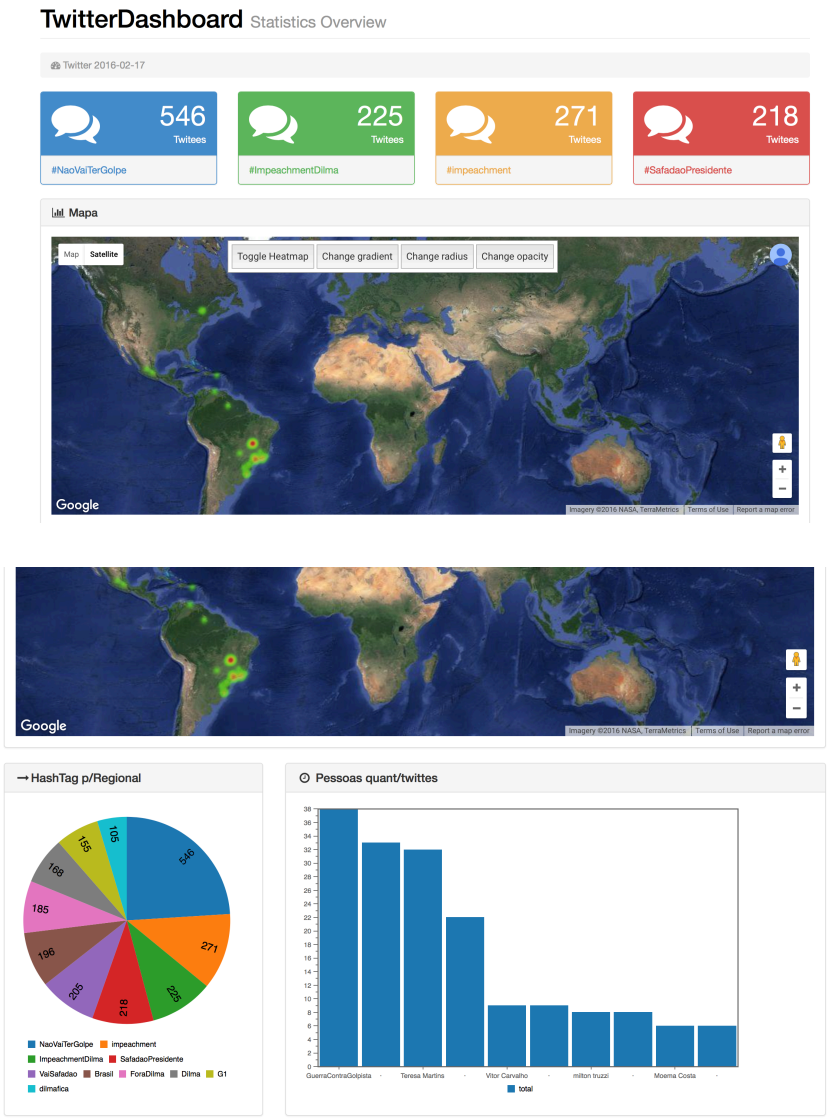


Figura 12 - Cuadro de Mando creado con CTools/Pentaho

Se puede decir que la primera versión de la herramienta ha producido resultados positivos y ha sido capaz de predecir el sentimiento general sobre la palabra #dilma.

11.2 Segunda versión

En esta versión se ha dado enfoque en Big Data Science, ya han sido añadido datos de la búsqueda de Google (que es otra fuente de datos) y la posibilidad de ser añadidas otras fuentes de datos en el futuro.

La versión final del software y arquitectura práctica se ha quedado como se puede ver abajo en la siguiente tabla y figura. Al largo del trabajo se explicarán todas las pruebas, integraciones y soluciones encontradas por el autor.

Capa Colecta de Datos	**Capa Data Lake**	**Capa Análisis de Sentimiento**	**Capa Visualización de los Datos**
1) Colectar los datos de distintas fuentes de datos	2) Almacenar los datos en una base de datos relacional (EDW) y en un entorno Big Data (Data Lake)	3) Análisis de sentimiento	4) Visualizar los datos en una herramienta Inteligencia de Negocio
Capa de Ingestión de los datos con Pentaho Data Integration leyendo los datos de Twitter y Google Search utilizando una integración con Python;	Los datos almacenados en *PostgreSQL*, Apache *HDFS* y *Apache Hive*;	La capa de predicción del sentimiento para actualizar el campo del sentimiento ha sido desarrollada utilizando el entorno R, actualmente no es posible actualizar en tiempo real el sentimiento, se utiliza una rutina en batch para predecir el sentimiento y actualizar la columna en la tabla almacenada con *PostgreSQL*. En el futuro se busca una solución para hacer en tiempo real. La nube de palabra ha sido creada con la herramienta R.	Los cuadros de mandos para la visualización de los datos han sido creados con *Pentaho* CTools, *HTML5 y R*;
Python + Pentaho Data Integration	PostgreSQL, Apache HDFS y Apache Hive	R	Pentaho CTools, HTML5 y R

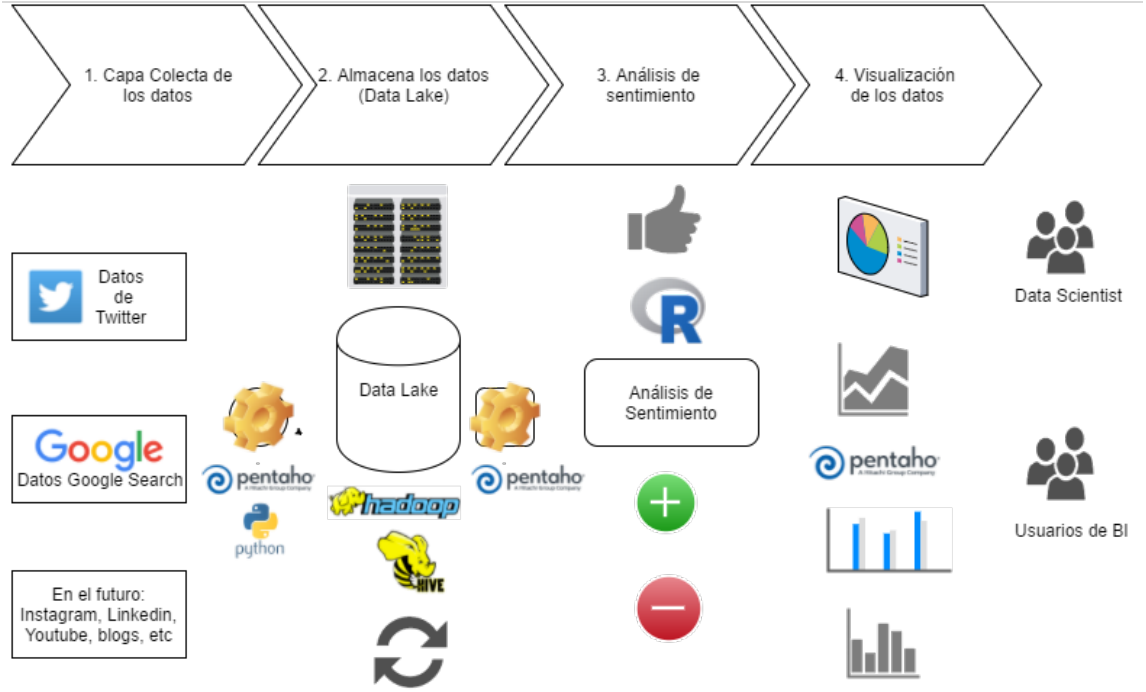


Figura 13 - Propuesta de Arquitectura Práctica

La nueva versión es la continuación y se ha invertido mucho esfuerzo en añadir un entorno Big Data para el proyecto y la posibilidad de añadir otras fuentes de datos.

Para el proyecto versión 2.0, se ha buscado probar distintas herramientas y se ha llegado a una propuesta final que explicaré con detalles.

El sistema global de análisis de sentimiento está dividido en 4 capas, son:

1) Capa de "Ingestión de Datos" de los datos y Almacenamiento de datos

Esta capa es responsable por hacer "*Data Ingestion*" y almacenamiento de los datos.

2) Preparación de los datos (*Data Preparation*)

La preparación de los datos (*Data Preparation*) es una parte importante del proceso siendo muy importante para las predicciones y la visualización de los datos.

3) Análisis de Sentimiento Global

La capa de análisis de sentimiento es donde se busca clasificar los mensajes en positivo y negativo.

4) Visualización de los datos

Se visualizan los datos con el apoyo de una herramienta de Business Intelligence.

Capa de "Data Ingestion" de los datos y Almacenamiento de datos

La colecta y almacenamiento de los datos es una parte muy importante del proceso, puede parecer algo simple cuando se trata de pocos datos, pero en este proyecto se ha vuelto algo muy complejo donde ha sido necesario probar distintas formas de ingestión de los datos.

El proceso de recogida y almacenamiento se ha alargado por muchos meses debidos a las pruebas realizadas con nuevas tecnologías y programas como python, R, Apache Spark, Apache Hadoop, Apache Hive, Impala, MongoDB y Apache Flume.

Durante muchos meses se han hecho pruebas y los resultados más importantes son los que se mencionará abajo en este trabajo.

Solución de ingestión de datos de twitter con shell script

Una de las primeras pruebas ha sido la de extraer los datos de Twitter directamente de la API de Streaming utilizando un shell script y almacenar los datos en un archivo json en una carpeta.

Esta solución resultó ineficiente cuando se vio que en pocas horas y días el archivo quedaba demasiado grande. Para el proyecto se han producido archivos con hasta 16 GB con tweets. Algunos de los archivos con los tweets recogidos se encuentran en https://github.com/caiomsouza/TwitterRawData y puede ser descargados.

```
#!/bin/bash
now="$(date)"
printf "Current date and time %s\n" "$now"

now="$(date +'%d/%m/%Y')"
printf "Current date in dd/mm/yyyy format %s\n" "$now"

echo "Starting backup at $now, please wait..."

#mkdir twitters-$(date '+%d%m%Y%H%M%S')

curl --get 'https://stream.twitter.com/1.1/statuses/filter.json' --data 'track=dilma' --header 'Authorization: OAuth oauth_consumer_key="",
```

Ha sido necesario utilizar el comando de abajo para eliminar la última línea del archivo .json con los twittees.

sed "$ d" original_json_file.js > new_json_file.js

Eso ha sido necesario porque siempre la última línea del archivo json estaba estropeada.

Solución con MongoDB

La prueba siguiente ha sido almacenar los tweets en una base de datos MongoDB, que resultó ser muy potente; incluso ha sido posible crear cuadros de mando con Pentaho (Herramienta de Inteligencia de Negocio) y leer los datos directamente de MongoDB.

Las pruebas con MongoDB ha sido documentadas y publicadas en el enlace https://github.com/caiomsouza/TwitterRawData/releases/tag/TwitterData4MongoDB-v.0.1

También se puede ver el cuadro de mando creado con la herramienta Pentaho leyendo los datos directamente de la base NoSQL MongoDB, por ahora no vamos enfocar en la visualización e interpretación de los datos.

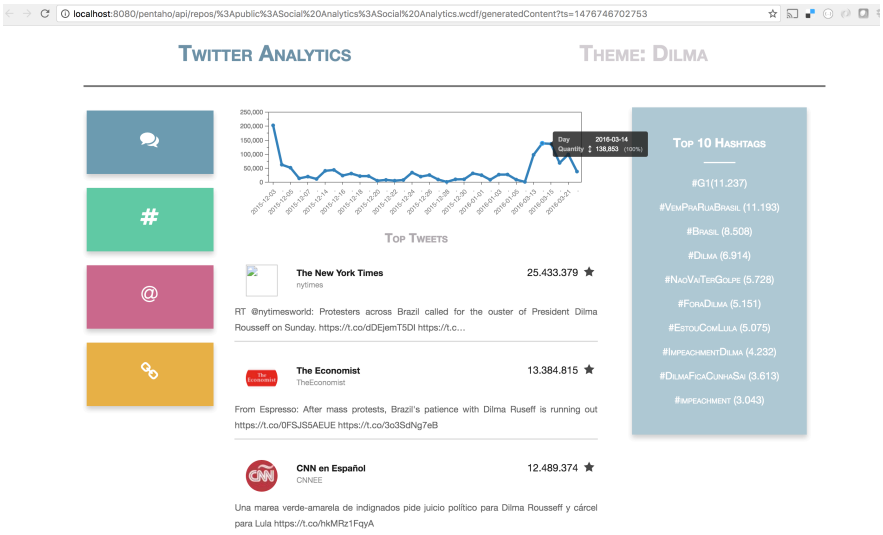
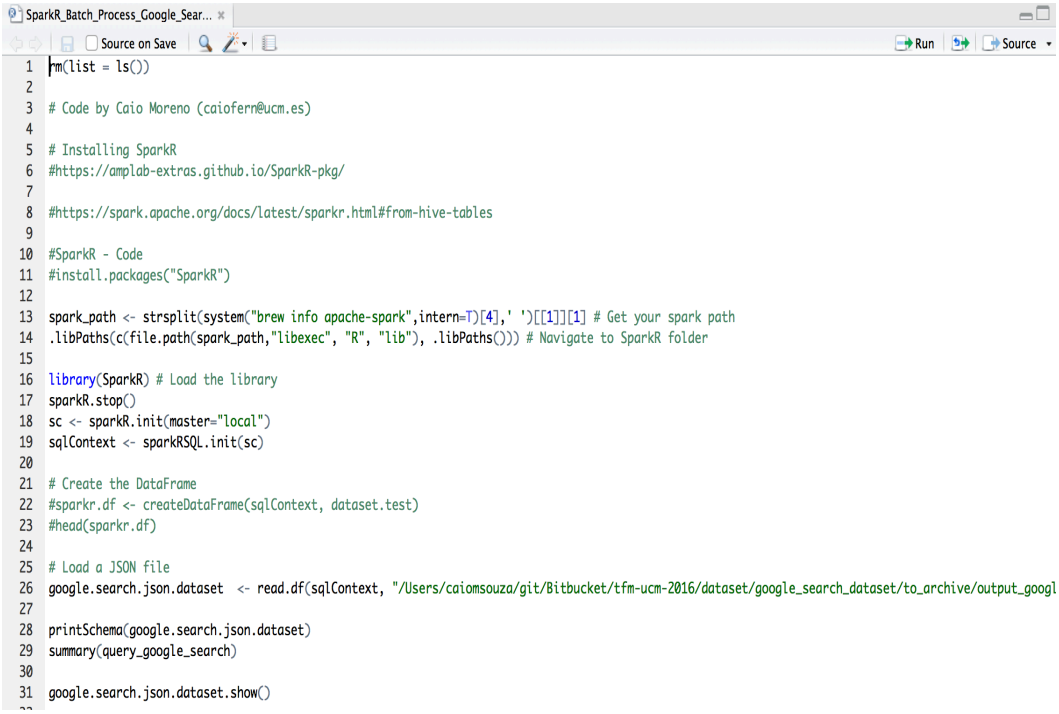


Figura 14 - Cuadro de mando leyendo los datos de MongoDB

La base de datos NoSQL MongoDB es muy potente y ha sido elegida como una solución muy eficaz para el almacenamiento de los datos y con una gran capacidad de integración con la herramienta de Business Intelligence Pentaho.

Solución con Apache SparkR SQL

Otra gran solución que se probó fue la ingestión y almacenamiento de los datos con Apache SparkR SQL, donde los archivos .json generados via shell script han sido enviados para Apache SparkR SQL utilizando la herramienta R.




```
rm(list = ls())

# Code by Caio Moreno (caiofern@ucm.es)

# Installing SparkR
#https://amplab-extras.github.io/SparkR-pkg/

#https://spark.apache.org/docs/latest/sparkr.html#from-hive-tables

#SparkR - Code
#install.packages("SparkR")

spark_path <- strsplit(system("brew info apache-spark",intern=T)[4],' ')[[1]][1] # Get your spark path
.libPaths(c(file.path(spark_path,"libexec", "R", "lib"), .libPaths())) # Navigate to SparkR folder

library(SparkR) # Load the library
sparkR.stop()
sc <- sparkR.init(master="local")
sqlContext <- sparkRSQL.init(sc)

# Create the DataFrame
#sparkr.df <- createDataFrame(sqlContext, dataset.test)
#head(sparkr.df)

# Load a JSON file
google.search.json.dataset  <- read.df(sqlContext, "/Users/caiomsouza/git/Bitbucket/tfm-ucm-2016/dataset/google_search_dataset/to_archive/output_googl

printSchema(google.search.json.dataset)
summary(query_google_search)

google.search.json.dataset.show()
```

Esta solución es bastante potente para trabajar con Apache Spark y su integración con R, siendo una buena opción entre las muchas opciones existentes.

Se ha generado un tutorial de cómo utilizar Apache Spark con el paquete SparkR para el entorno estadístico R.

En este enlace se explica como utilizar Apache Spark con SparkR:

https://github.com/caiomsouza/ucm/blob/master/tfm/sparkr-tutorial.md

En este enlace se puede ver un ejemplo de Apache SparkR con SparkR:

https://github.com/caiomsouza/ucm/blob/master/tfm/example_SparkR.R

Solución con Apache Flume

Se ha probado hacer la ingestión de los datos de tweets utilizando Apache Flume y el almacenamiento directamente en Apache HDFS. Esta solución ha demostrado ser una de las mejores soluciones cuando se piensa en escalabilidad, pero una solución costosa por la necesidad de tener un clúster Hadoop.

```
# Other config values specific to each type of channel(sink or source)
# can be defined as well
# In this case, it specifies the capacity of the memory channel
agent.channels.memoryChannel.capacity = 100

TwitterAgent.sources = Twitter
TwitterAgent.channels = MemChannel
TwitterAgent.sinks = HDFS

TwitterAgent.sources.Twitter.type = com.cloudera.flume.source.TwitterSource
TwitterAgent.sources.Twitter.channels = MemChannel
TwitterAgent.sources.Twitter.consumerKey = C
TwitterAgent.sources.Twitter.consumerSecret = 95k4BJav8Wc
TwitterAgent.sources.Twitter.accessToken = 204478988-LePQErpHejWdlP
TwitterAgent.sources.Twitter.accessTokenSecret = FvemJ4cBCjOd05ARSfynxIC

TwitterAgent.sources.Twitter.keywords = big data, pentaho, weka, kettle, hitachi, hds, kettle, mondrian, sap, orac

TwitterAgent.sinks.HDFS.channel = MemChannel
TwitterAgent.sinks.HDFS.type = hdfs
TwitterAgent.sinks.HDFS.hdfs.path = hdfs://quickstart.cloudera:8020/user/cloudera/stage/flume-tweets/tech-words/
TwitterAgent.sinks.HDFS.hdfs.fileType = DataStream
TwitterAgent.sinks.HDFS.hdfs.writeFormat = Text
TwitterAgent.sinks.HDFS.hdfs.batchSize = 1000
TwitterAgent.sinks.HDFS.hdfs.rollSize = 0
TwitterAgent.sinks.HDFS.hdfs.rollCount = 1000000

TwitterAgent.channels.MemChannel.type = memory
TwitterAgent.channels.MemChannel.capacity = 1000000
TwitterAgent.channels.MemChannel.transactionCapacity = 100
```

Figura 15 - Código HDFS + Apache Flume + Twitter

Se puede ver que los tweets son almacenados directamente en Apache HDFS en archivos de 3 MB.

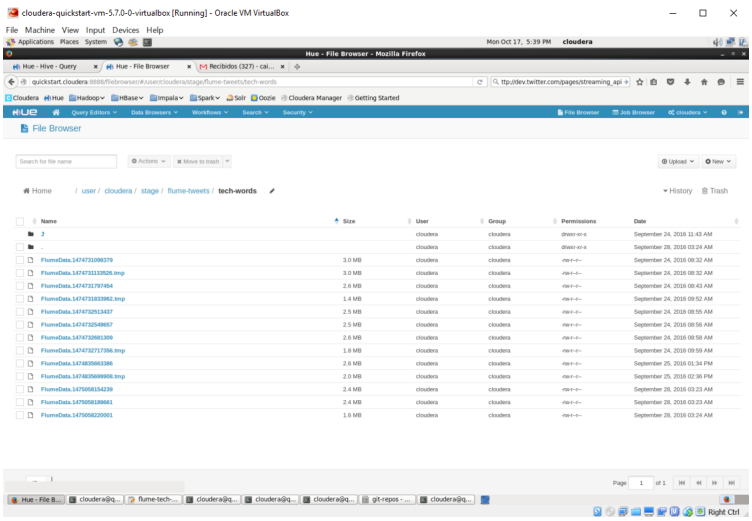


Figura 16 - Datos de Twitter almacenados en HDFS con Apache Flume

Para esta prueba ha sido necesario utilizar el entorno de Big Data de Cloudera, todas las pruebas han sido documentadas en el enlace https://github.com/caiomsouza/big-data-science.

Para este trabajo ha sido necesario utilizar Apache Flume para colectar los datos directamente de la API Streaming de Twitter y almacenar en Apache HDFS.

El entorno utilizado ha sido un entorno de Hadoop single node (único nodo) con 10 GB de RAM pero no ha sido suficiente para ejecutar por más de algunos minutos la solución, después de algunos minutos se generaba un error de *Out of Memory*, por cuestiones financieras no se ha probado con un clúster de servidores.

Solución con PDI (Pentaho Data Integration), R, Python y Apache HDFS

Se han creado y a veces adaptado algunas transformaciones (transformations) y trabajos (jobs) en la herramienta de integración de datos llamada PDI (Pentaho Data Integration).

Con PDI es posible programar de forma visual, se ha utilizado esta herramienta para recoger los datos de Twitter y Google Search y almacenar en una base de datos PostgreSQL. Otras pruebas han sido hechas enviando los datos directamente para Apache HDFS (Data Lake).

La solución de ingestion de los datos con PDI, R, Python y Apache HDFS es una solución muy robusta y se puede trabajar con muchos datos con esta solución.

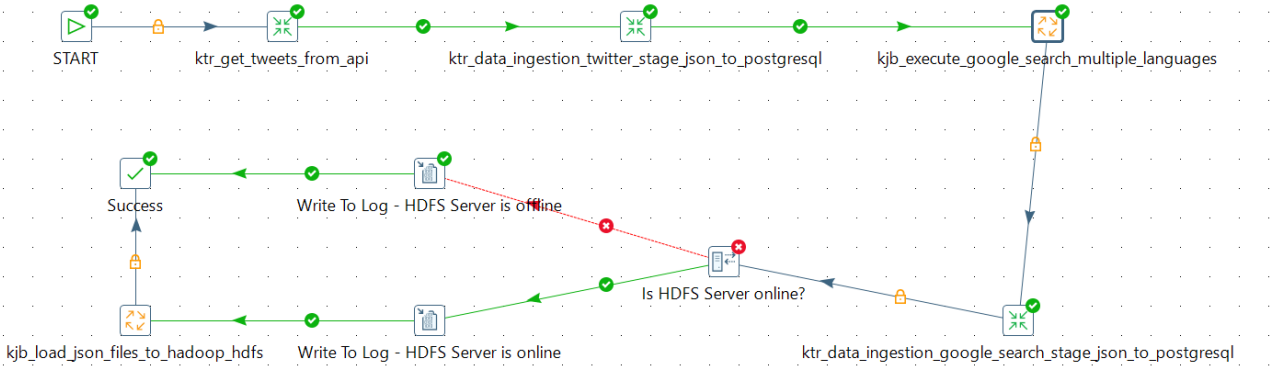


Figura 17: Trabajo (Job) para leer los datos de Twitter y Google Search y grabar en una base de datos PostgreSQL y en el Data Lake (HDFS y Hive)

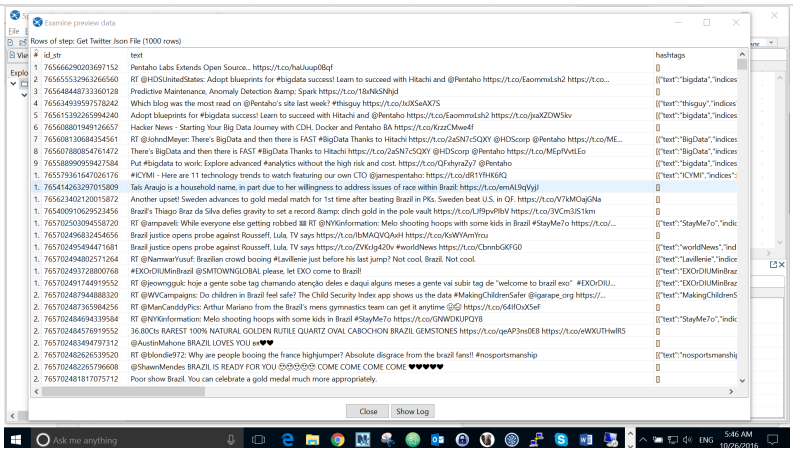


Figura 18 - Datos de Twitter colectados por la herramienta PDI

No se han puesto en este trabajo todas las imágenes de todos los trabajos (Jobs) y transformaciones (transformations) creados, para probar y verlos todos, se recomienda acceder al código del proyecto.

En la imagen de abajo se pueden ver los datos siendo almacenados en el Data Lake (HDFS).

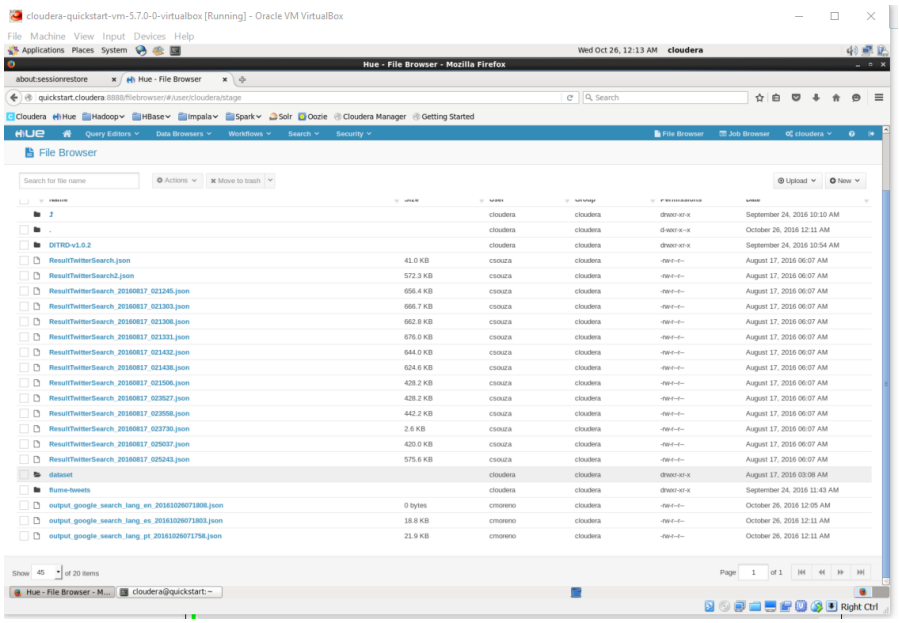


Figura 19 - Datos de Twitter y Google Search en el Data Lake colectados con PDI

Solución con Apache Hive

Se ha probado leer los datos de Twitter y Google Search y almacenar en Apache HDFS y cargar en Apache Hive para posibilitar ejecutar consultas SQL en Apache Hive, también se ha probado conectar Apache Hive con R y con el Pentaho BA Server para utilizar una herramienta OLAP sobre los datos de Apache Hive.

https://github.com/caiomsouza/ucm/blob/master/tfm/hive-tutorial.md

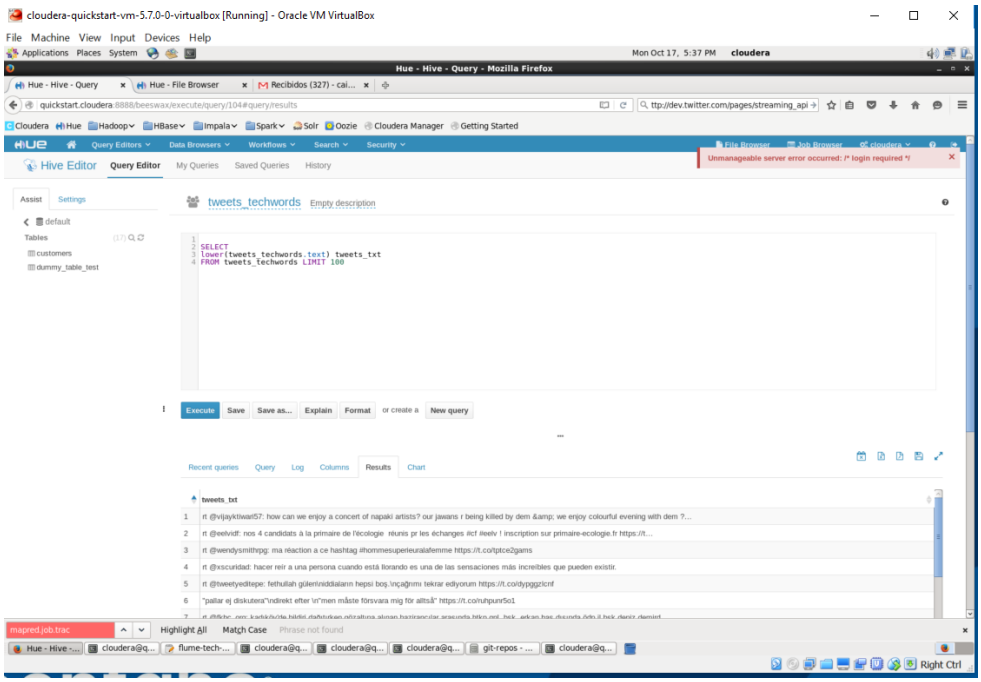


Figura 20 - Consulta a los datos de Twitter almacenados en Apache HDFS/Hive

Solución con Raspberry PI y Amazon S3

Una solución muy económica ha sido utilizar un mini ordenador Raspberry Pi Model B para colectar los datos de Twitter y del buscador de Google y enviar para el servicio de Amazon llamado S3. Esta solución es muy interesante por la parte conceptual, donde cada Raspberry Pi es un sensor y cada 10 minutos envía sus datos locales para una estructura de Big Data, en este caso el servicio S3. El servicio Amazon S3 es correspondiente con Apache HDFS, pues en Amazon se utiliza S3 para almacenar los datos.

Lo que más me gustó de esta solución es la simplicidad, coste muy bajo y capacidad de mantener muchos mini ordenadores recopilando datos y enviándolos a S3.

La solución completa se encuentra disponible en https://github.com/caiomsouza/raspberrypi

Elección de la mejor solución de ingestión y almacenamiento

Una de las grandes dificultades del proyecto fue elegir la mejor solución de ingestión y almacenamiento, no se ha podido llegar a una conclusión utilizando criterios de selección, pues cada solución presenta características distintas y todas son muy potentes.

Pero sí se ha elegido una solución final, la cual ha sido utilizar un cluster Hadoop para almacenamiento de los datos.

Los datos son almacenados en el ***"Data Lake"*** y se utiliza Apache Hive para las operaciones de Map Reduce.

La ventaja de utilizar esta solución es la facilidad de almacenar cualquier tipo de fichero en el Apache HDFS y después procesar los datos de la forma que sea necesaria.

La forma de ingestión de los datos puede ser definida de acuerdo con la necesidad. Una gran opción es tener mini ordenadores con Raspberry Pi para generar datos y enviar a Amazon S3.

Capas	Solución	Complejidad	Escalabilidad
Colecta de Datos y Data Lake	Raspberry PI + Amazon S3	Media	Alta
Data Lake	Apache Hive	Alta	Alta
Data Lake	Mongo DB	Media	Alta
Colecta de Datos y Data Lake	Apache Flume + HDFS	Alta	Alta
Colecta de Datos y Data Lake	PDI + Python + PostgreSQL + HDFS + Hive	Baja	Alta

La solución elegida ha sido PDI + Python + PostgreSQL + HDFS + Hive, por una decisión personal, pues es una solución de gran estabilidad y media-baja complejidad. Todas las otras soluciones son muy recomendables y pueden ser utilizadas para distintos casos de uso.

El coste de los datos

Cuándo se piensa en un proyecto de Big Data muchas veces no se piensa en un tema de capital importancia, que es ¿quién va pagar la cuenta del proyecto? Esta pregunta es la más importante que debe ser respondida en la fase inicial del proyecto, es decir, antes de todo hay que saber quién es la persona responsable para aprobar o no los costes de los datos y saber cuál es el *ROI (Return of Investiment)* esperado para el proyecto, es decir, qué lleva a esta persona o a la empresa a invertir en el proyecto.

Esta cuestión es muy simple de entender. Imagine que el dinero es suyo y le proponen almacenar una cantidad increíble de datos para usted y preparar unos informes, cuadros de mando e informes OLAP sensacionales, pero que le va costar una cantidad muy grande de dinero, ¿cuál sería su reacción?

La primera pregunta que un ejecutivo experimentado haría es: ¿cuál es el beneficio esperado de este proyecto? Cualquier proyecto tiene que pasar por un análisis de viabilidad técnica donde se analiza si este proyecto es o no importante para la empresa y cuáles serán la rentabilidad y los beneficios esperados.

Estimar el coste del proyecto es mucho más que estimar el coste de los datos, hay que tener en cuenta muchas cosas, pero para facilitar la tarea vamos trabajar, por ahora, solo con el coste de almacenamiento en Amazon S3. Para este proyecto se ha utilizado el coste por GB disponible en la calculadora de Amazon y se ha llegado al valor de $ 0,03 dólares por cada GB. Es decir, para almacenar 100 GB de datos hay que invertir como mínimo $ 2,89 dólares todos los meses. Este coste es solo para almacenar datos, a esto hay que añadir el coste de servidores, personal y todos los costes de un proyecto de Big Data. La idea de utilizar el coste de S3 es para dar una idea, muy por encima, del coste del almacenamiento de un GB de datos (no estamos hablado del coste de procesamiento de los datos y otros costes).

Servicio / Region	100 GB	1000 GB	10000 GB	100000 GB
Amazon S3 (US-EAST)	$2,89	$29,89	$295,41	$3218,62
Coste por GB	$0,0289	$0,02989	$0,029541	$0,0321862
Data Transfer Out (GB / Month)	0 GB	0 GB	0 GB	0 GB
Data Transfer In (GB / Month)	100 GB	1000 GB	10 TB	100 TB
PUT/COPY/POST/LIST Request	10000	10000	10000	10000
GET and Other Requests	10000	10000	10000	10000

Datos extraídos el día 18 de octubre de 2016 desde la página http://calculator.s3.amazonaws.com/index.html

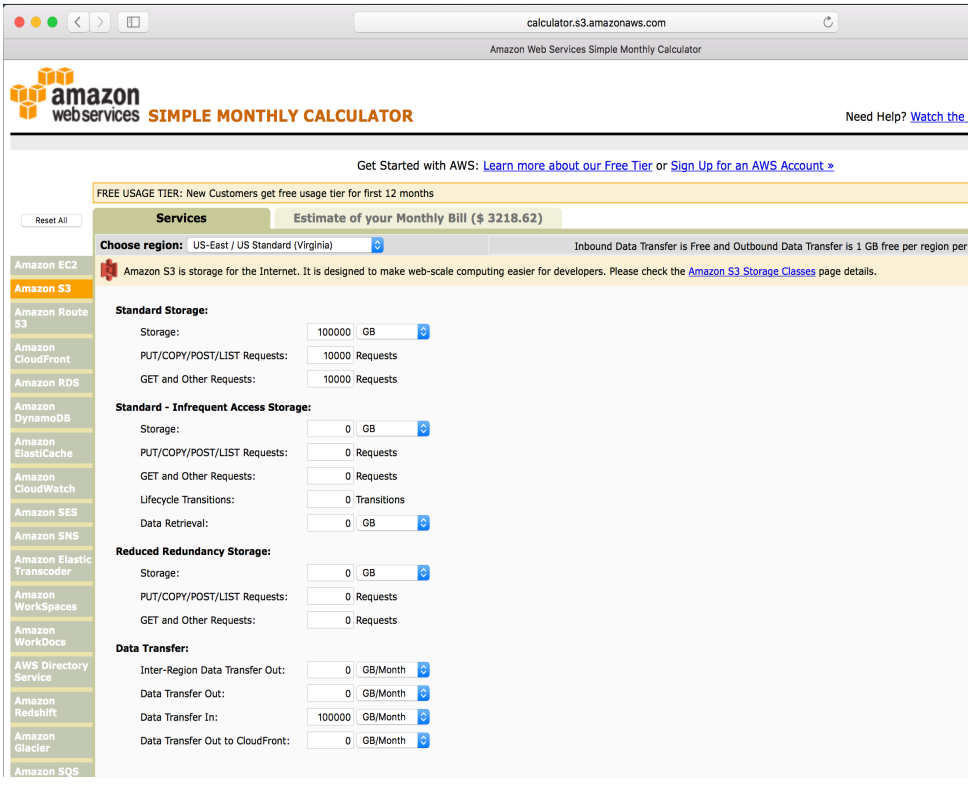
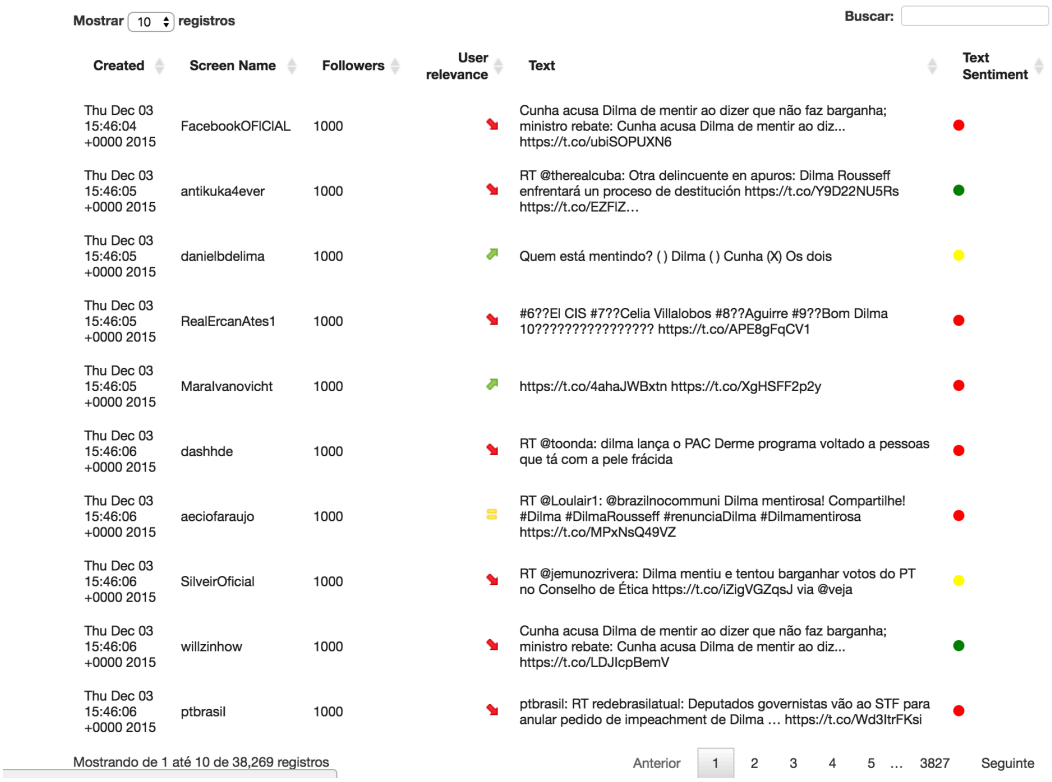


Figura 21 - Calculadora Amazon

Capa de visualización – Cuadro de Mando, una solución tradicional de Inteligencia de Negocio

Con el cuadro de mando de abajo se pueden ver los mensajes, la relevancia del usuario que es basada en la cantidad de seguidores y el sentimiento del mensaje.



Mostrar 10 registros — Buscar:

Created	Screen Name	Followers	User relevance	Text	Text Sentiment
Thu Dec 03 15:46:04 +0000 2015	FacebookOFICIAL	1000		Cunha acusa Dilma de mentir ao dizer que não faz barganha; ministro rebate: Cunha acusa Dilma de mentir ao diz... https://t.co/ubiSOPUXN6	
Thu Dec 03 15:46:05 +0000 2015	antikuka4ever	1000		RT @therealcuba: Otra delincuente en apuros: Dilma Rousseff enfrentará un proceso de destitución https://t.co/Y9D22NU5Rs https://t.co/EZFIZ...	
Thu Dec 03 15:46:05 +0000 2015	danielbdelima	1000		Quem está mentindo? ( ) Dilma ( ) Cunha (X) Os dois	
Thu Dec 03 15:46:05 +0000 2015	RealErcanAtes1	1000		#6??El CIS #7??Celia Villalobos #8??Aguirre #9??Bom Dilma 10???????????????? https://t.co/APE8gFqCV1	
Thu Dec 03 15:46:05 +0000 2015	Maralvanovicht	1000		https://t.co/4ahaJWBxtn https://t.co/XgHSFF2p2y	
Thu Dec 03 15:46:06 +0000 2015	dashhde	1000		RT @toonda: dilma lança o PAC Derme programa voltado a pessoas que tá com a pele frácida	
Thu Dec 03 15:46:06 +0000 2015	aeciofaraujo	1000		RT @Loulair1: @brazilnocommuni Dilma mentirosa! Compartilhe! #Dilma #DilmaRousseff #renunciaDilma #Dilmamentirosa https://t.co/MPxNsQ49VZ	
Thu Dec 03 15:46:06 +0000 2015	SilveirOficial	1000		RT @jemunozrivera: Dilma mentiu e tentou barganhar votos do PT no Conselho de Ética https://t.co/iZigVGZqsJ via @veja	
Thu Dec 03 15:46:06 +0000 2015	willzinhow	1000		Cunha acusa Dilma de mentir ao dizer que não faz barganha; ministro rebate: Cunha acusa Dilma de mentir ao diz... https://t.co/LDJIcpBemV	
Thu Dec 03 15:46:06 +0000 2015	ptbrasil	1000		ptbrasil: RT redebrasilatual: Deputados governistas vão ao STF para anular pedido de impeachment de Dilma ... https://t.co/Wd3ltrFKsi	

Mostrando de 1 até 10 de 38,269 registros — Anterior 1 2 3 4 5 ... 3827 Seguinte

Figura 22 - Cuadro de Mando con análisis de sentimiento y relevancia creado con Pentaho

Para determinar el sentimiento de un tweet es posible utilizar dos técnicas distintas: i) método basado en lexicón ii) método basado en aprendizaje de máquina (*machine learning*).

El método basado en aprendizaje supervisado utiliza un clasificador y un corpus y el método basado en lexicón es un aprendizaje no supervisado o semántico donde se utiliza un diccionario de palabras positivas y negativas.

Se sabe, gracias a estudios anteriores de otros investigadores de la comunidad académica mundial, que utilizar un clasificador basado en SVM y Naive Bayes supera el rendimiento del método basado en lexicón, pero la ventaja del método basado en lexicón es no tener que crear corpus para cada dominio, ahorrando tiempo y creando un método más genérico para distintos dominios.

Es posible mejorar muchísimo el rendimiento con un método ensamblado donde se utiliza una puntuación de sentimiento basado en el método de lexicón como una variable para el método de aprendizaje de maquina con SVM o Naive Bayes.

Actualmente se puede tener resultados positivos con los dos métodos. La precisión obtenida con clasificadores multi dominio (método basado en lexicón) son de 70 - 75% y la precisión en clasificadores específicos a partir del 80% (método basado en aprendizaje de máquina y un corpus).

El principal reto actualmente en la investigación del análisis de sentimiento es tratar la subjetividad. Esto es un gran reto para las personas; para las máquinas es aún más grande.

Este trabajo ha buscado encontrar un método automático de clasificar el sentimiento de un texto con la máxima precisión posible, pues con una cantidad de información tan grande es imposible hacer este trabajo manualmente. También es importante aclarar que el enfoque es hacerlo para el idioma portugués.

La imagen de abajo muestra otra técnica de minería de datos llamada nube de palabras (en este trabajo también se ha creado una nube de palabras de los Tweets).

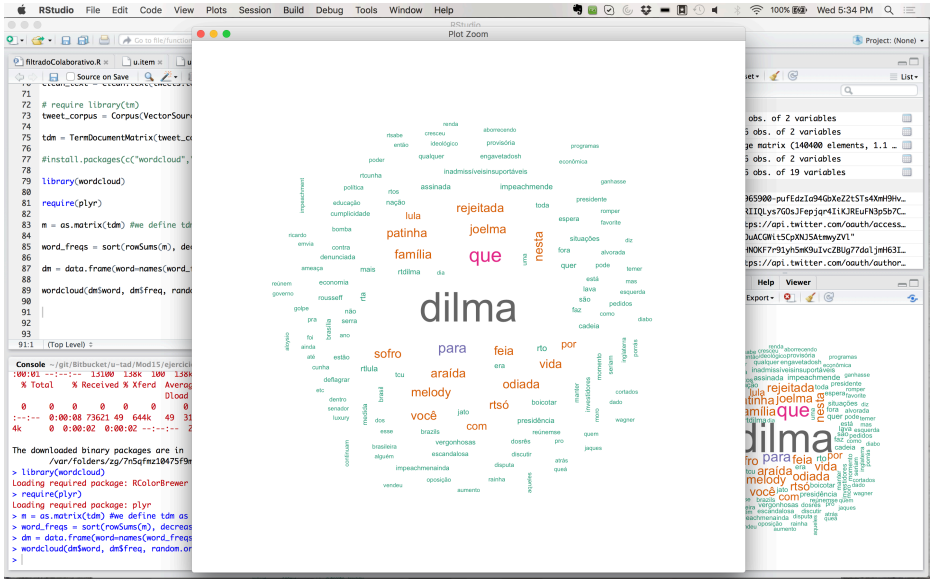


Figura 23 - Nube de Palabras

12. Material y Métodos

Se han utilizado para este trabajo los siguientes programas y materiales:

- Draw.io (https://www.draw.io);
- Entorno R y R Studio;
- Apache Spark, Apache SparkR, Apache Flume, Apache Hive;
- Pentaho BI Suite e Pentaho Data Integration;
- Python;
- PostgreSQL;
- MongoDB;
- Raspberry Pi 3 Model B; y
- Amazon AWS (S3 y EC2).

13. Conclusiones y trabajo futuro

El objetivo principal de este trabajo fue definir una arquitectura conceptual y practica de análisis de sentimiento global capaz de extraer y almacenar diversas fuentes con gran volumen de datos (Big Data), se ha utilizado *twitter y google search*, pero se creo una capa/sistema preparada para añadir otras más como: instagram, youtube, blogs, Facebook. Además, que generase conocimiento que se pudiera integrar en un entorno de Business Intelligence de las empresas donde los usuarios pudieran visualizar los datos por medio de una herramienta *OLAP* o por medio de cuadros de mandos para ayudar en la gestión de la crises y en la toma de decisión.

Así, se propuso la creación de una herramienta que utilizara las técnicas de Business Intelligence, Big Data y Data Science, y que a través de la aplicación y ejercicio de 4 actividades principales (colecta, almacenamiento, utilización de técnicas de análisis de sentimiento y visualización de datos), pudiera fornecer la información de los sentimientos generales de una determinada muestra, ayudando en la toma de decisiones y gestión de crisis. Dicha herramienta tiene una arquitectura basada en el uso de algunas otras herramientas (estas herramientas habían que ser de código abierto, una vez que se buscó crear una arquitectura capaz de escalar para gestionar muchísimos volúmenes de datos, y ser una solución de bajo coste para ser implantada en pequeñas, medianas y grandes empresas o gobiernos), y la arquitectura que se ha demostrado ser la más adecuada está formada de la siguiente manera:

- colecta de datos de distintas fuentes de datos como Twitter y Google utilizando Python y Pentaho Data Integration;
- almacenamiento de datos en una base de datos relacional y en un entorno de Big Data (HDFS y Hive);
- análisis de sentimiento utilizando R; y
- visualización del dato en una herramienta de inteligencia de negocio llamada Pentaho.

Se han testado varias configuraciones practicas, en busca de la que presentase los mejores resultados (y aquí definimos que los mejores resultados serian los resultados que mostraran los resultados que más se acercasen a lo sentimiento real de la muestra colectada).

Los resultados encontrados han sido que existen diversas opciones de extraer, almacenar, procesar y visualizar los datos, se puede decir que Python, Pentaho Data Integration, MongoDB, Hadoop HDFS, Hive y R son las mejores herramientas para la creación de la solución final.

Se concluye, basado en que ha sido posible identificar, en el período poco antes y durante el proceso de impechment, por medio del twitter y por las búsquedas en google, que el sentimiento global de las personas referente al impechment era positivo (las personas deseaban eso), y negativo referente a la presidente Dilma (se ha podido identificar muchísimos mensajes con sentimiento negativo antes mismo de empezar el proceso de impechment), y también que el resultado final, que ha sido la salida de Dilma del gobierno de Brasil y la llegada del vicepresidente al poder, reflejó exactamente el sentimiento general de la mayoría de las personas de la muestra colectada.

Para trabajos futuros se busca ser capaz de hacer una segmentación de las personas, creando grupos de personas basado en sus mensajes. También se busca mejorar la capa de análisis de sentimiento creando un modelo ensamblado con más capacidad predictiva y más entrenado para el dominio.

Se considera necesario también mejorar la capa de visualización creando nuevas visualizaciones con el intuito de responder más preguntas de negocio.

La capa de almacenamiento y procesamiento de los datos también se puede mejorar para ser capaz de hacer análisis predictivos en tiempo real.

Aún que se pueda decir que no se ha llegado a un modelo satisfactorio en termos de capacidad predictiva, se puede considerar que esta aceptable para una herramienta inicial. La solución en producción a pesar de sus debilidades aún se puede considerar que no tener nada.

El resultado es satisfactorio para el autor y se puede decir que se ha hecho las muchas pruebas con distintas tecnologías y se puede decir que se ha llegado a una arquitectura capaz de ser utilizada en producción por una empresa o organización para hacer de forma diaria la gestión de las crisis.

Lo que no se puede monitorear no se puede mejorar, entonces se puede decir que con esta herramienta en producción una organización es capaz de monitorear múltiples fuentes de datos y gestionar sus crisis a punto de tomar decisiones más acertadas y con más antelación.

También se recomienda que las organizaciones sigan teniendo personas para garantizar que lo informado por la solución refleja la realidad; no podemos dejar que las maquinas tomen las decisiones por nosotros, ellas solo nos van ayudar, pero no substituirnos en nuestras tareas.

14. Referencias

Cambria, E., Schuller, B., Xia, Y., & Havasi, C. (2013). New avenues in opinion mining and sentiment analysis. *IEEE Intelligent Systems*, *28*(2), 15-21.

Cambria, Erik, et al. "Semantic multi-dimensional scaling for open-domain sentiment analysis." (2013): 1-1.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine learning*, *20*(3), 273-297.

Dietterich, T. G. (2000, June). Ensemble methods in machine learning. In *International workshop on multiple classifier systems* (pp. 1-15). Springer Berlin Heidelberg.

Fersini, E., Messina, E., & Pozzi, F. A. (2014). Sentiment analysis: Bayesian ensemble learning. *Decision support systems*, *68*, 26-38.

Glance, N., Hurst, M., Nigam, K., Siegler, M., Stockton, R., & Tomokiyo, T. (2005, August). Deriving marketing intelligence from online discussion. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining* (pp. 419-428). ACM.

Go, A., Bhayani, R., & Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, *1*, 12.

Hu, M., & Liu, B. (2004, August). Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 168-177). ACM.

Julia Hanna. (2008). JetBlue's Valentine's Day Crisis. 20/08/2016, de Harvard Business School Sitio web: http://hbswk.hbs.edu/item/jetblues-valentines-day-crisis

Kadam, Sachin A., and Mrs Shweta T. Joglekar. "Sentiment Analysis, an Overview." International Journal of Research in Engineering & Advanced Technology, 1/4, p1 7 (2013).

Kouloumpis, E., Wilson, T., & Moore, J. D. (2011). Twitter sentiment analysis: The good the bad and the omg!. *Icwsm*, *11*, 538-541.

Kumar, Akshi, and Teeja Mary Sebastian."Sentiment analysis on twitter."IJCSI International Journal of Computer Science Issues 9.3 (2012): 372-378.

Mcdonald, R., Mohri, M., Silberman, N., Walker, D., & Mann, G. S. (2009). Efficient large-scale distributed training of conditional maximum entropy models. In *Advances in Neural Information Processing Systems* (pp. 1231-1239).

Melville, P., Gryc, W., & Lawrence, R. D. (2009, June). Sentiment analysis of blogs by combining lexical knowledge with text classification. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 1275-1284). ACM.

Mohammad, S. M., Kiritchenko, S., & Zhu, X. (2013). NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets. *arXiv preprint arXiv:1308.6242*.

Muchemi, Lawrence. *A social media sentiment analysis model to support marketing intelligence in Kenya*. Diss. University of Nairobi, 2015.

National Geographic. (2012). 14 de julio de 1789, la toma de la Bastilla. 10/09/2016, de National Geographic Sitio web: http://www.nationalgeographic.com.es/historia/grandes-reportajes/14-de-julio-de-1789-la-toma-de-la-bastilla_6776

Nigam, K., Lafferty, J., & McCallum, A. (1999, August). Using maximum entropy for text classification. In *IJCAI-99 workshop on machine learning for information filtering* (Vol. 1, pp. 61-67).

Pak, A., & Paroubek, P. (2010, May). Twitter as a Corpus for Sentiment Analysis and Opinion Mining. In *LREc* (Vol. 10, pp. 1320-1326).

Pang, B., Lee, L., & Vaithyanathan, S. (2002, July). Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10* (pp. 79-86). Association for Computational Linguistics.

RAE. (2016). Crisis. 03/08/2016, de RAE Sitio web: http://dle.rae.es/?id=BHwUydm

Rambocas, M., & Gama, J. (2013). *Marketing research: The role of sentiment analysis* (No. 489). Universidade do Porto, Faculdade de Economia do Porto.

Rodrigo Orihuela, Dina Bass. (2015). Help Wanted: Black Belts in Data. 08/11/2016, de Bloomberg Enlace web: http://www.bloomberg.com/news/articles/2015-06-04/help-wanted-black-belts-in-data

Steven Overly. (2013). As demand for big data analysts grows, schools rush to graduate students with necessary skills. 08/11/2016, de The Washington Post Enlace web: https://www.washingtonpost.com/business/capitalbusiness/as-demand-for-big-data-analysts-grows-schools-rush-to-graduate-students-with-necessary-skills/2013/09/13/afbafb3e-1a66-11e3-82ef-a059e54c49d0_story.html

Tan, S., & Zhang, J. (2008). An empirical study of sentiment analysis for chinese documents. *Expert Systems with Applications*, *34*(4), 2622-2629.

Thomas H. Davenport y D.J. Patil. (2012). Data Scientist: The Sexiest Job of the 21st Century. 08/11/2016, de Harvard Business Review Enlace web: https://hbr.org/2012/10/data-scientist-the-sexiest-job-of-the-21st-century

Turney, P. D. (2002, July). Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting on association for computational linguistics* (pp. 417-424). Association for Computational Linguistics.

Varghese, R., & Jayasree, M. (2013). A survey on sentiment analysis and opinion mining. *IJRET: International Journal of Research in Engineering and Technology eISSN*, *23191163*.

Vinodhini, G., & Chandrasekaran, R. M. (2012). Sentiment analysis and opinion mining: a survey. *International Journal*, *2*(6).

Wiebe, J., & Riloff, E. (2005, February). Creating subjective and objective sentence classifiers from unannotated texts. In *International Conference on Intelligent Text Processing and Computational Linguistics* (pp. 486-497). Springer Berlin Heidelberg.

Witten, I. H., & Frank, E. (2005). Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann.

Wöllmer, M., Weninger, F., Knaup, T., Schuller, B., Sun, C., Sagae, K., & Morency, L. P. (2013). Youtube movie reviews: Sentiment analysis in an audio-visual context. *IEEE Intelligent Systems*, *28*(3), 46-53.

Xia, R., Zong, C., & Li, S. (2011). Ensemble of feature sets and classification algorithms for sentiment classification. *Information Sciences*, *181*(6), 1138-1152.

Xia, Y. Q., Xu, R. F., Wong, K. F., & Zheng, F. (2007, August). The unified collocation framework for opinion mining. In *2007 International Conference on Machine Learning and Cybernetics* (Vol. 2, pp. 844-850). IEEE.

Xu, K., Liao, S. S., Li, J., & Song, Y. (2011). Mining comparative opinions from customer reviews for Competitive Intelligence. *Decision support systems*, *50*(4), 743-754.

Ye, Q., Zhang, Z., & Law, R. (2009). Sentiment classification of online reviews to travel destinations by supervised machine learning approaches. *Expert Systems with Applications*, *36*(3), 6527-6535.

Zeynep Tufekci: Machine Learning makes humans morals more important. TEDSummit. 2016. Enlace web: https://www.ted.com/talks/zeynep_tufekci_machine_intelligence_makes_human_morals_more_important