Universidad de Buenos Aires
Facultad de Ciencias Exactas y Naturales
Departamento de Computación

Tesis de Licenciatura

Modelos de sentimiento no dependientes de dominio en español

Alumno:
Maximiliano Neustadt
maxneust@gmail.com

Director:
Dr. José Castaño
jcastano@dc.uba.ar

1° de Junio de 2011

Agradecimientos

A mi director, José, por su importante guía en todo el proceso y por tenerme paciencia para desarrollar la mayor parte de la tesis a la distancia.

A todos los profesores de la carrera, por la buena onda y predisposición.

A los pibes con los que fui compartiendo la cursada, aquellas trasnochadas de TPs se sufrían bastante pero ahora se recuerdan con nostalgia.

A mi viejo, por la ayuda invaluable de sus consejos.

A mi vieja, por seguir hinchando para que termine la carrera incluso ya con 29 años, pero principalmente por estar siempre junto con mi viejo para darme una mano con lo que necesite.

A Lore, mi amor, también un poquito por unirse a mi querida madre para hinchar, y sobre todo por acompañarme y bancarme a lo largo de todos los buenos y malos momentos que me tocaron vivir a lo largo de la carrera.

Y al resto de mi familia y a mis amigos, por todos los buenos momentos compartidos, que hicieron este proceso más llevadero.

Índice general

Índice de cuadros

Capítulo 1

Resumen

El minado de opinión es un área de investigación relativamente joven (las publicaciones "fundacionales" - [Tur02] y [PLV02] son del año 2002), y su objetivo es extraer la orientación semántica de un conjunto de textos para clasificarlos de acuerdo a ella como positivas o negativas.

Los enfoques para atacar este problema son esencialmente dos:

1. **Orientación semántica**: en este enfoque lo que se busca es utilizar técnicas existentes de Procesamiento de Lenguaje Natural para encontrar la información semántica contenida en cada palabra y oración del documento y a partir de ello obtener un score global que permita clasificar al documento.

2. **Machine Learning - Text Classification**: las técnicas desarrolladas dentro de este grupo tratan al problema como uno tradicional de clasificación de textos, siendo las de mejores resultados Naïve Bayes, Maximum Entropy y Support Vector Machines[DLP03][CTC05][WBB$^+$03], [BHDM07], [PT09], [SN10].

Las técnicas del segundo enfoque obtienen muy buenos resultados si los textos a clasificar se encuentran dentro de un mismo dominio, pero al salirse del mismo la performance decae.

La ventaja del primer enfoque es que generalmente las técnicas existentes (y la propuesta en la presente tesis) utilizan recursos de orientación semántica ya disponibles que además son completamente independientes del dominio, por lo tanto los resultados de estos clasificadores no dependen de una etapa de entrenamiento, que es la que genera la dependencia al dominio al que pertenezcan las opiniones utilizadas en el proceso.

Gran parte de las investigaciones que se están realizando en la actualidad se encuentra enfocada hacia el idioma inglés, por lo que el objetivo de este trabajo será analizar el comportamiento de las técnicas existentes en corpus que se encuentren en español, analizando el comportamiento de los métodos existentes de Machine Learning y proponiendo dos nuevas técnicas dentro del enfoque de orientación semántica. Esto será luego evaluado en distintos dominios, para luego realizar un análisis detallado y sugerir mejoras para el problema de dependencia de dominio.

Abstract

Opinion mining is a relatively new research subject (the "foundational" publications - [Tur02] y [PLV02] were published in 2002), and its purpose is to extract the semantic orientation of a group of texts to clasify it according to that orientation into positive or negative.

The approaches to tackle this problem are essentially two:

1. **Semantic orientation**: in this approach the idea is to use existing Natural Language Processing techniques to find the semantic information contained in each word and sentence of the document and use it to obtain a global document score that would allow for its classification.
2. **Machine Learning - Text Classification**: the techniques developed in this group treat the problem as a standard text classification problem, and the current publications have generally found Naïve Bayes, Maximum Entropy and Support Vector Machines as the best performing ones.

The Machine Learning techniques have very good performance if the texts to classify are within a same domain, but if that's not the case the performance drops considerably.

The advantage of the semantic orientation approach is that the existing techniques (and the ones proposed in this thesis) use already-available semantic orientation resources that also are completely context-independent,

which makes the classifiers using those also context-independent.

Most of the current research is focused towards English, so the objective of this thesis will be to study the behavior of the techniques in spanish corpus, introducing the existing Machine Learning techniques and proposing two new ones within the semantic orientation approach. This will then be evaluated on different domains and insight will be given regarding improvements for the domain dependency problem of the Machine Learning approach.

Capítulo 2

Introducción

El minado de opinión es un área de investigación surgida hace sólo unos años, y ha cobrado mucha más relevancia con el surgimiento de la "web 2.0", debido al crecimiento exponencial de la información disponible para ser procesada. Consiste principalmente en:

- **Clasificación de polaridad de textos**
  Aquí el objetivo principal es determinar la polaridad de textos, a nivel de documento, sentencia, o "atributo" (es decir, de las características que componen al objeto sobre el que se está opinando). Además, una rama de esta línea busca obtener la polaridad en escala de "estrellas", no sólo de manera binaria en positiva o negativa.

- **Identificación de objetividad y subjetividad**
  En este caso lo que se busca es determinar qué partes del documento son objetivas y cuáles subjetivas, información que puede ser aplicada en tareas posteriores (por ejemplo, clasificar sólo sentencias subjetivas, ignorando las objetivas).

2.1. Aplicaciones

Hoy en día, gran parte del contenido disponible en la web contiene algún tipo de opinión sobre una amplia cantidad de tópicos. Esta información es

difícil de acceder o utilizar cuando no está centralizada (en sitios específicos para cargarla) o sin una evaluación definida (ejemplo: escala de estrellas para opiniones de películas), y es por ello que el minado de opinión ha cobrado mucha relevancia en los últimos años: procesar esta información para poder utilizarla de manera relevante se ha vuelto una fuente de información potencialmente amplísima en diversos contextos, principalmente en sitios de opiniones (que en este caso funcionarían a la inversa que los tradicionales: no tendrían que "convocar" a los usuarios para que escriban opiniones, sino que las buscarían y clasificarían proactivamente) o para realizar estudios de mercado acerca de productos o personas (por ejemplo, saber qué opina la gente en blogs o sitios similares acerca de la duración de la batería de los iPhone, o cuál fue la reacción de la gente ante alguna medida tomada por un político, etc).

2.2. Desafíos actuales

En la actualidad existen dos desafíos principales en el campo de minado de opinión: el primero es inherente al lenguaje humano: muchas veces las opiniones se expresan de tal manera que no es completamente claro si la opinión en cuestión es positiva o negativa, e involucra una gran cantidad de problemas de lenguaje natural (detección del uso de ironía, sarcasmo, condicionalidad, etc). El segundo es la dependencia de dominio: las técnicas más utilizadas en la actualidad, que son las de Machine Learning, tienen una alta dependencia del modelo que arman a partir del conjunto de entrenamiento, lo que provoca resultados pobres si el entrenamiento se realiza con opiniones de un dominio y se evalúa en otro.

2.3. Trabajo a realizar

El objetivo de este trabajo es, en primera instancia, atacar el problema de asignación de polaridad a textos a nivel de documento para opiniones en español, utilizando métodos existentes de Machine Learning; evaluando a la vez su comportamiento con respecto a la dependencia de dominio. Se

propondrán y evaluarán luego dos técnicas independientes de dominio, y finalmente se propondrán alternativas para mejorar la dependencia de dominio en los métodos de Machine Learning.

2.4. Organización del texto

El capítulo 2 presenta los métodos de Machine Learning que se utilizarán en la experimentación, describiendo a grandes rasgos los fundamentos de los modelos que forman para utilizar posteriormente a la hora de clasificar. Se presenta aquí también un método de convergencia hacia modelos independientes de dominio.

El capítulo 3 describe los métodos de Orientación Semántica, que son los independientes de dominio. Se presenta en primer lugar la estructura general, y luego cada uno de los métodos.

En el capítulo 4 se describe el corpus utilizado en la experimentación.

El capítulo 5 explica cómo se desarrollarán los experimentos, y describe los resultados obtenidos. Además, basado en los resultados se hace un análisis más detallado del comportamiento de los métodos de Machine Learning en un contexto multi-dominio.

El capítulo 6 sugiere el trabajo futuro a encarar en base a los resultados obtenidos.

Por último, el capítulo 7 presenta las conclusiones obtenidas a partir del análisis de la experimentación realizada.

Capítulo 3

Machine Learning - Text classification

Se han realizado muchos trabajos tratando al problema directamente como uno de clasificación de textos tradicional, tomando como "categorías" a los dos grupos semánticos representando opiniones positivas y negativas. En [PLV02] las técnicas elegidas son Naïve Bayes, MaxEnt y Support Vector Machines. Trabajos subsiguientes ([WBB$^+$03], [BHDM07], [PT09], [SN10] y otros) han utilizado también estas técnicas con buenos resultados, y por esa razón será con las que trabajaremos en esta tesis.

Debido a que estos métodos son netamente estadísticos suponemos que dependen fuertemente del contexto, y esto significaría que si las características más comunes de cada clase varía mucho entre dominios, la clasificación de documentos de un dominio no entrenado tendrá resultados pobres. En la etapa de resultados se hace un análisis detallado de esta situación, y en la sección "Extracción de características comunes" (3.5) se propone una alternativa que busca atenuar esta dificultad.

3.1. Feature Selection

Los métodos a utilizar, para clasificar un documento reciben como entrada un conjunto de "features" del mismo, y calculan la probabilidad de

pertenencia del documento a cada clase en base a esas características. Esta selección puede ser por caracteres, palabras o n-gramas, e incluso luego de haber pasado por un filtrado previo, como por ejemplo exclusión de las oraciones que sean detectadas como objetivas[1] [PL04]. Utilizar palabras es la alternativa que ha tenido consistentemente buenos resultados en los diversos trabajos ya realizados y por esa razón es la alternativa elegida en la etapa de evaluación.

En las siguientes secciones se presentarán cada uno de los métodos a utilizar.

[1] En el sentido netamente semántico; es decir, aquellas que no acarrean información respecto de la orientación semántica del texto.

3.2. Naïve Bayes

Si uno definiese un modelo probabilístico para un clasificador, tomando a C como la variable discreta que define las categorías representando a opiniones positivas y negativas y $\{F_1 \dots F_n\}$ el conjunto de "features" del documento a clasificar, dicho modelo se definiría de la siguiente manera:

$$P(C|F_1, F_2, \dots., F_n) \tag{3.1}$$

Es decir, la probabilidad de que dado un documento d con características F_i pertenezca a la clase $C = c_i$ está condicionada, naturalmente, por la probabilidad de ocurrencia de cada F_i. Utilizando al Teorema de Bayes, (3.1) se puede reescribir como:

$$P(C|F_1, F_2, \dots, F_n) = \frac{P(C) * P(F_1, \dots, F_n|C)}{P(F_1, \dots, F_n)} \tag{3.2}$$

Si observamos el numerador, aplicando primero la definición de Probabilidad Compuesta[2] (pc), y luego la regla de la cadena[3] (rc), obtenemos el siguiente resultado:

$$\begin{aligned} P(C) * P(F_1, F_2, \dots, F_n|C) &\overset{pc}{=} P(C, F_1, \dots, F_n) \\ &\overset{rc}{=} P(C) * P(F_1|C) * P(F_2|C, F_1) * \dots * \\ &\quad P(F_n|C, F_1, \dots, F_{n-1}) \end{aligned} \tag{3.3}$$

En este punto es donde el método obtiene el adjetivo de *Naïve*: asume que los "features" son probabilísticamente independientes, con lo cual utilizando la definición de probabilidad condicional[4] bajo esta asunción, (3.3) queda:

$$P(C) * P(F_1|C) * \dots * P(F_n|C) = P(C) \prod_{1 \leq i \leq n} P(F_i|C) \tag{3.4}$$

[2] $P(A_1, \dots, A_n) = P(A_1) * P(A_2, \dots, A_n|A_1)$

[3] $P(A_1, \dots, A_n) = P(A_1) * P(A_2|A_1) * P(A_3|A_1, A_2) * \dots * P(A_n|A_1, \dots, A_{n_1})$

[4] $P(A|B) = \frac{P(A,B)}{P(B)}$, pero si A y B son independientes, $P(A, B) = P(A) * P(B)$, por lo tanto $P(A|B) = \frac{P(A)*P(B)}{P(B)} = P(A)$

Cabe aclarar que dicho supuesto no parece intuitivo, ya que en cualquier texto sus términos son generalmente dependientes[5]. A pesar de esta fuerte decisión, el método ha sido extensamente utilizado en clasificación de textos con resultados satisfactorios.

Si se reemplaza en (3.1) lo obtenido en (3.4) nos queda que:

$$P(C|F_1, F_2, \ldots, F_n) = \frac{P(C) \prod_{1 \leq i \leq n} P(F_i|C)}{P(F_1, \ldots, F_n)} \tag{3.5}$$

En nuestro caso P(C) puede ignorarse, ya que trabajaremos siempre con un corpus balanceado (misma cantidad de opiniones positivas y negativas) para evitar polarizaciones hacia una categoría u otra dadas simplemente por una mayor cantidad de opiniones de esa categoría (en el caso más extremo, si se tienen 1000 positivas y 1 negativa, de todas maneras términos claramente negativos tendrán orientación positiva debido a que en un corpus conformado de esa manera aparecerán mayormente en opiniones anotadas como positivas). Nótese también que $P(F_1, \ldots, F_n)$ es un dato conocido - la frecuencia de los "features" en el documento a clasificar - y que por ser conocido y constante para todas las clases, se suele ignorar; y más relevantemente, que la productoria es un dato fácilmente obtenible luego de haber corrido al clasificador con un conjunto de entrenamiento.

De esta manera, partiendo del cálculo de una probabilidad *a posteriori* para determinar el grado de pertenencia a una clase, obtuvimos una expresión equivalente que se basa exclusivamente en la evidencia y los datos obtenidos en etapa de entrenamiento, y es con esta expresión que Naïve Bayes realiza la clasificación de acuerdo a la clase para la que obtenga mayor probabilidad. Formalmente, siendo $\{F_1, \ldots, F_n\}$ los "features" del documento D, la clasificación es realizada de la siguiente manera:

[5]Por ejemplo, no es correcto asumir que la aparición de la palabra *reproductor* no condiciona la probabilidad de aparición de la palabra *mp3* ya que, en efecto, son altamente dependientes.

$$clase(D) = \underset{c \in C}{\operatorname{argmax}} \frac{P(C = c) \prod_{1 \leq i \leq n} P(F_i | C = c)}{P(F_1, \ldots, F_n)} \tag{3.6}$$

3.3. Maximum Entropy

Este método, al igual que Naïve Bayes, también se basa en la definición de un modelo probabilístico para determinar el grado de pertenencia de un documento d (con "features" $\{f_1, \ldots, f_n\}$) a una categoría c. Dicho modelo es el siguiente:

$$P_{ME}(c|d) = \frac{1}{Z(d)}\ exp(\sum_{1 \leq i \leq n} \lambda_{i,c} F_{i,c}(d,c)) \quad , \tag{3.7}$$

donde:

- $Z(d)$ es una función de normalización para que (3.7) sea efectivamente una distribución de probabilidad, y está definida de la siguiente manera:

$$Z(d) = \sum_{c_j \in c} exp(\sum_{1 \leq i \leq n} \lambda_{i,c_j} F_{i,c_j}(d,c_j)) \quad , \tag{3.8}$$

- $F_{i,c}$ es una función de los "features" del documento y las clases, y se utiliza para restringir al modelo probabilístico de manera tal que lo observado en la etapa de entrenamiento se refleje en el modelo utilizado para clasificar nuevos documentos. En general, se define de la siguiente manera:

$$F_{i,c}(d,c) = \begin{cases} 1 & \text{si } frecuencia(f_i, d) > 0 \\ 0 & \text{sino} \end{cases}$$

- $\lambda_{i,c}$ son los parámetros que indican el peso de cada f_i, donde valores elevados indican una mayor probabilidad de pertenencia a c. La obtención de los valores se realiza resolviendo un problema de optimización a través del algoritmo *Improved Iterative Scaling*[SDPL97]

El modelo definido de esta manera elige valores para $\lambda_{i,c}$ de manera tal que se maximiza la entropía de la distribución obtenida si se tiene en cuenta además la restricción de que los valores esperados de $F_{i,c}$ con respecto

al modelo son iguales a los valores esperados con respecto a los datos de entrenamiento[SDPL97]. La idea detrás de este planteo es simple: maximizar la cantidad de información extraída en la etapa de entrenamiento, sin hacer ningún tipo de asunción respecto de los datos y manteniendo consistencia con respecto a los mismos. Esta es una diferencia clave con respecto a Naïve Bayes, que asume independencia de "features", y es por eso que en ciertos contextos donde la dependencia es fuerte, este método responde mejor[KNM99].

3.4. SVM

Este método es un clasificador que asigna clases de acuerdo a una combinación lineal de los "features" del documento. De manera intuitiva, el modelo de este clasificador representa al conjunto de entrenamiento como puntos en un espacio n-dimensional, siendo n la cantidad de "features" a considerar, y busca encontrar un hiperplano que maximice la distancia entre los puntos de cada categoría[6]. Los puntos de categorías distintas más cercanos entre sí son denominados *support vectors*, y son los únicos puntos utilizados como límite para clasificar. De esta manera, la decisión de si un documento pertenece a una clase u otra se realiza simplemente representando al documento en el espacio correspondiente y observando de qué lado del hiperplano cae. La figura 3.1 representa gráficamente esta situación.

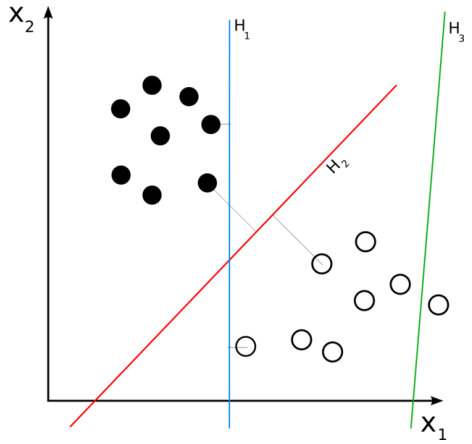

Figura 3.1: Posibles hiperplanos clasificadores. H3 no es un hiperplano que divida a los documentos en las categorías apropiadas, H2 sí pero con un margen pequeño y H1 es el buscado, de máximo margen

[6]El cálculo de este valor a partir del conjunto de entrenamiento se resuelve como un problema de optimización sujeto a una definición inicial genérica del hiperplano. La definición inicial es lineal, pero luego se extendió a diversas versiones no-lineares utilización funciones kernel[CV95].

Maximizar el margen de decisión es intuitivamente positivo por dos razones principales:

- Debido a que los puntos cercanos a la superficie de decisión (es decir, el área determinada por el hiperplano resultante y los support vectors) representan aquellos con menor nivel de certeza de pertenencia a cualquiera de las clases, al tener una mayor superficie las decisiones se toman con mayor certeza, y esta es la diferencia principal con clasificadores probabilísticos como Naïve Bayes o MaxEnt donde una pequeña variación en el documento o en el valor de los "features" elegidas puede causar una clasificación errónea.
- Debido al mayor tamaño de la superficie de decisión, hay una menor cantidad de opciones que cumplan con las restricciones impuestas a la hora de calcularlas, por lo tanto se reduce el margen de error por una elección demasiado ajustada al conjunto de entrenamiento.

3.5. Extracción de características comunes

Debido a la dependencia de dominio de las técnicas vistas en esta sección, se propone a continuación un método que permita conformar un conjunto de "features" lo más universal posible, en el sentido de que los elementos que la componen funcionan como buenos (o en el peor de los casos, neutros) clasificadores independientemente del contexto al que se apliquen. La idea es la siguiente:

1. Entrenar al clasificador con documentos de un mismo dominio. Extraer del resultado un número predeterminado de "features" con mayor frecuencia, f_1, tomándolas como "características universales" (f_u).

2. Para cada nuevo dominio que se desea clasificar, entrenar un nuevo clasificador y obtener las características correspondientes, f_2.

3. Incluir de a una cada característica $c_{i_2} \in f_2$ en f_1, y volver a clasificar los documentos del corpus existente con $\{c_{i_2}\} \cup f_1$. En caso de que esa característica no haya empeorado la performance del clasificador, se la **incluye** en f_u.

4. Incluir de a una cada característica $c_{i_1} \in f_1$ en f_2, y volver a clasificar los documentos del nuevo dominio con $\{c_{i_1}\} \cup f_2$. En caso de que esa característica empeore la performance del clasificador, se la **excluye** de f_u.

De esta manera, suponemos que a medida que se vayan agregando dominios, se convergerá a un conjunto relativamente estable de características independientes de dominio. El riesgo que se corre aquí es acabar con un conjunto excesivamente pequeño que termine degradando demasiado la performance del clasificador. De todas maneras, si ese llega a ser el caso, significaría que no es posible conformar un conjunto universalmente utilizable, debido a que cada dominio tiene características muy propias que no pueden ser extrapolables a dominios distintos.

Capítulo 4

Clasificadores por Orientación Semántica

Este tipo de clasificadores utilizan inicialmente una técnica de procesamiento de lenguaje natural denominada part-of-speech tagging (o "etiquetado morfosintáctico") para extraer información sintáctica de cada documento, separándolo en oraciones y luego dentro de cada oración obteniendo la categoría léxica de sus palabras.

Una vez que el corpus está etiquetado con esta información, se procede a extraer la orientación semántica de las palabras o frases y consecuentemente de las oraciones, y a partir de sus resultados individuales a obtener una puntuación global para el documento.

La extracción de información semántica es generalmente realizada mediante una fuente externa, la cual ya sea por poseer una etapa previa de entrenamiento o por estar compuesta por un gran número de documentos, posee un grado de independencia de contexto mayor que los corpus que generalmente se utilizan en los clasificadores intra-dominio de Machine Learning. La mayoría de los trabajos se basan en [Tur02], que utiliza el algoritmo PMI-IR (Pointwise Mutual Information and Information Retrieval) para medir la similaridad de pares de palabras, tomando como corpus algún motor de

búsqueda web; otra alternativa con buenos resultados es la utilización de SentiWordNet[BS10], un diccionario de orientación semántica basado en el contenido de WordNet[MBF$^+$90], una base de datos léxica disponible online.

En las siguientes secciones se explicará, implementará y evaluará una alternativa que utiliza PMI-IR y una que utiliza SentiWordNet, ambas con 4 etapas de procesamiento bien definidas, donde sólo la segunda y tercera se diferencian:

1. Etiquetado morfosintáctico del documento.
2. Extracción de términos relevantes para la clasificación a realizar.
3. Cálculo de la orientación semántica de cada término extraído, que será mayor a 0 si es positiva o menor si es negativa.
4. Determinar la orientación del documento sumando los valores obtenidos en 3).

4.1. Etiquetado morfosintáctico

Esta tarea es común a los dos métodos a presentar, razón por la cual se presenta a continuación. Existen varias herramientas disponibles para esta tarea, siendo FreeLing[PCR+10] la mejor entrenada para el español. El proceso consiste en 4 etapas:

I) **Tokenizado**: toma como entrada texto plano y genera una lista de objetos *palabra*.

II) **Separación de oraciones**: separa las *palabras* en las correspondientes *oraciones*.

III) **Análisis morfosintáctico**: anota cada *palabra* de las *oraciones* con los valores morfosintácticos que encuentra más apropiados, aplicando secuencialmente procesos específicos de cada clase (además de la categoría sintáctica, detección de números, fechas, nombres propios, frases y otros).

IV) **Etiquetado morfosintáctico**: toma la salida de III y aplica un proceso de lematización (es decir, obtención de la forma más común de una palabra dado el contexto y el "part of speech", por ejemplo la versión en singular de un sustantivo o la forma infinitiva de un verbo[1]) y luego de desambiguación de sentido para quedarse con la etiqueta más apropiada para cada palabra (en términos del contexto en el que se encuentra).

La salida de este proceso es un archivo con una palabra del documento original por línea, aumentada con la correspondiente información morfosintáctica. Las oraciones se separan con una línea en blanco.

El cuadro 5.1 muestra un ejemplo. La primera letra indica la categoría morfosintáctica, y las siguientes son específicas de la categoría correspondiente[2].

[1] Notar que esto es distinto a stemming, proceso mediante el cual se obtiene la raíz de la palabra - o sea, el prefijo común a las distintas formas posibles de la misma

[2] La información completa del significado de cada etiqueta está disponible en *http://nlp.lsi.upc.edu/freeling/doc/userman/parole-es.html*

Palabra	**Lema**	**Etiqueta**	**Significado etiq.**
Luego	luego	RG	Categoría: R - Adverbio Tipo: G - General
resulta	resultar	VMIP3S0	Categoría: V - Verbo Tipo: M - Principal Modo: I - Imperfecto Tiempo: P - Presente Persona: 3 - Tercera Cardinalidad: S - Singular Género: 0 - No definido
que	que	CS	Categoría: C - Conjunción Tipo: S - Subordinada
la	el	DA0FS0	Categoría: D - Determinante Tipo: A - Artículo Persona: 0 - No definida Género: F - Femenino Modo: I - Imperfecto
buena	bueno	AQ0FS0	Categoría: A - Adjetivo Tipo: Q - Calificativo Grado: 0 - No definido Género: F - Femenino Cardinalidad: S - Singular Función: 0 - No definida
mujer	mujer	NCFS000	Categoría: N - Sustantivo Tipo: C - Común Género: F - Femenino Cardinalidad: S - Singular Clasif Semántica: 00-No def. Grado: 0 - No definido
está	está	RG	Categoría: R - Adverbio Tipo: G - General
especializada	especializar	VMP00SF	Categoría: V - Verbo

			Tipo: M - Principal Modo: P - Participio Tiempo: 0 - No definido Persona: 0 - No definida Cardinalidad: S - Singular Género: F - Femenino
en	en	SPS00	Categoría: S - Adposición Tipo: P - Preposición Forma: S - Simple Género: 0 - No definido Cardinalidad: 0 - No def.
libros	libro	NCMP000	Categoría: N - Sustantivo Tipo: C - Común Género: M - Masculino Cardinalidad: P - Plural Clasif Semántica: 00-No def. Grado: 0 - No definido
para	para	SPS00	Categoría: S - Adposición Tipo: P - Preposición Forma: S - Simple Género: 0 - No definido Cardinalidad: 0 - No def.
adolescentes	adolescente	NCCP000	Categoría: N - Sustantivo Tipo: C - Común Género: C - Común Cardinalidad: P - Plural Clasif Semántica: 00-No def. Grado: 0 - No definido
y	y	CC	Categoría: C - Conjunción Tipo: C - Coordinada
novelas	novela	NCFP000	Categoría: N - Sustantivo Tipo: C - Común

			Género: F - Femenino Cardinalidad: P - Plural Clasif Semántica: 00-No def. Grado: 0 - No definido
románticas	románticas	AQ0FP0	Categoría: A - Adjetivo Tipo: Q - Calificativo Grado: 0 - No definido Género: F - Femenino Cardinalidad: P - Plural Función: 0 - No definida
.	.	Fp	Categoría: F - Puntuación Tipo: p - punto

Cuadro 4.1: Ejemplo de etiquetado morfosintáctico

4.2. SO-PMI

4.2.1. Extracción de términos

Una vez que los documentos han sido etiquetados, se procede a extraer aquellas frases que se consideren indicadoras de orientación semántica. Diversos estudios, como [HW00], [Wie00], y [WBB+01], denotan que en general son los adjetivos -y en menor medida los adverbios- los que suelen indicar más fuertemente algún tipo de sentimiento asociado a la oración en cuestión; y basándose en estas conclusiones se decide por extraer frases que posean alguno de los dos. [Tur02] sugiere que no se extraigan simplemente las palabras, sino bigramas que las contengan, ya que en ciertos casos la orientación semántica depende de lo que esté siendo calificado: a priori, el adjetivo "increíble" tiene connotación positiva, como en el caso de "una vista increíble" al evaluar un hotel, pero si la frase es "una increíble desorganización" al hablar de un tour turístico, la connotación es claramente negativa. El autor define un conjunto de reglas con la estructura de los bigramas a seleccionar, pero como está pensada para el inglés para este trabajo se decidió adoptar la propuesta para el español de [FC08], que puede verse en el cuadro 4.2.

1er palabra	**2da palabra**	**3er palabra (no extraída)**
ADJ	SUST	cualquiera
SUST	ADJ	¬ SUST
ADV	ADJ	¬ SUST
ADV	VERB	cualquiera
VERB	ADV	cualquiera

Cuadro 4.2: Reglas para extracción de frases

A modo de ejemplo, en el siguiente párrafo se marcan en **negrita** (con la forma a la que corresponden entre paréntesis) las frases que serían seleccionadas para determinar orientación semántica con este proceso:

"Sinceramente, la película me ***desagradó muchísimo*** *(VERB+ADV). Las escenas de acción son* ***muy decepcionantes*** *(ADV+VERB), las ex-*

*plosiones son las **más falsas** (ADV+ADJ) que he visto; y las actuaciones dejan muchísimo que desear. Además, me encontré con la **triste sorpresa** (ADJ+SUST) de que Steven Seagal muere a los 5 minutos de iniciada la película. ¡Yo la había ido a ver por él! La escena inicial, en el avión, es la única en la que hay algo de tensión en toda la película, el resto es **aburrimiento puro** (SUST+ADJ). En corto: **Mala trama** (ADJ+SUST) **peor ejecución** (ADJ+SUST)*[3].

4.2.2. Cálculo de orientación semántica de los términos extraídos

Para cada frase extraída se procede a calcular el valor de su orientación semántica mediante una adaptación de PMI-IR (pointwise mutual information-information retrieval), presentado en la ecuación 4.1

$$PMI(t_1, t_2) = \log_2(\frac{p(t_1 \wedge t_2)}{p(t_1) \wedge p(t_2)}) \tag{4.1}$$

En esta ecuación, $p(t_1 \wedge t_2)$ es la probabilidad de que el término 1 (t_1) y el término 2 (t_2) co-ocurran en el corpus de búsqueda. Si los términos son estadísticamente independientes, entonces esta probabilidad es igual a $p(t_1) * p(t_2)$. Por lo tanto, el resultado de $\frac{p(t_1 \wedge t_2)}{p(t_1)*p(t_2)}$ es una medida del grado de dependencia estadística de los términos y el resultado en base/escala log_2 de esta división es entonces la cantidad de información obtenida respecto de la presencia de una palabra cuando se observa la otra. En términos más coloquiales, el resultado de esta ecuación representa cuán seguido aparece t_1 junto t_2. Con respecto al corpus de búsqueda, teniendo en cuenta los resultados de los trabajos que utilizaron a la web, se eligió el mismo camino para este trabajo; en particular al buscador Yahoo! ya que la obtención y parseo de los resultados es más simple que con otros. Este buscador provee

[3]Notar que aquí se hubiera aplicado la segunda regla (SUST+ADJ) para "trama peor" si no existiera la condición que excluye frases con esa forma pero cuya tercer palabra es también un sustantivo. Esa condición permite evitar este tipo de situaciones (peor corresponde a ejecución y no a trama), y se extiende también al caso de adverbios (por ejemplo, si la frase hubiese sido "trama extremadamente mala, peor ejecución", la regla ADV+ADJ no se hubiera aplicado para "mala peor" por la misma razón.

el operador binario NEAR, que devuelve resultados donde ambos términos aparezcan a 10 o menos palabras de distancia.

Como la intención aquí es determinar la orientación semántica, la fórmula se aplica tomando a t_1 como la frase a evaluar, y a t_2 como semilla representativa de cada clase (se entiende como semilla representativa a una palabra que define claramente la pertenencia a una clase). La propuesta inicial de [Tur02] es utilizar la palabra *excellent* para orientación positiva, y *poor* para orientación negativa, por lo que la ecuación para determinar la orientación semántica de t_1 sería:

$$SO(t_1) = PMI(t_1, \text{“}excellent''\text{)} - PMI(t_1, \text{“}poor''\text{)}, \tag{4.2}$$

que queda de la siguiente forma, luego de reemplazos y reglas del logaritmo, y tomando la probabilidad de ocurrencia como la cantidad de hits en el corpus (es decir, la cantidad de resultados devuelta por el buscador para la consulta “t_1 NEAR t_2”):

$$SO(t_1) = \log_2\left(\frac{hits(t_1 \wedge excellent) * hits(poor)}{hits(t_1 \wedge poor) * hits(excellent)}\right) \tag{4.3}$$

De esta manera, la frase tendrá orientación positiva si el resultado es mayor o igual a cero, y negativa en caso contrario.

Para obtener una mejor muestra de la relación semántica entre la frase y cada clase, proponemos evaluar cuáles serían los resultados si se aumentasen las semillas a un conjunto en lugar de una sola palabra [TL03] y [FC08], elegido de manera tal que cada palabra sea independiente de contexto, y presentado a continuación:

- **Positivas**: excelente, bueno, buenísimo, superior, extraordinario, magnífico, exquisito, positivo, genial, grandioso
- **Negativas**: pésimo, malo, malísimo, inferior, deplorable, detestable, horrible, negativo, atroz, fatal

En este caso, la ecuación 4.2 pasa a ser:

$$SO(t_1) = \sum_{p \in posWords} PMI(t_1, p) - \sum_{n \in negWords} PMI(t_1, n), \tag{4.4}$$

que con las mismas operaciones algebraicas aplicadas previamente queda:

$$SO(t_1) = \log_2\left(\frac{\prod_{p \in posWords} hits(t_1 \wedge p) * \prod_{n \in negWords} hits(n)}{\prod_{n \in negWords} hits(t_1 \wedge n) * \prod_{p \in posWords} hits(p)}\right) \tag{4.5}$$

4.2.3. Clasificación del documento

Una vez obtenidos los resultados para cada frase, se suman los resultados individuales y se clasifica al documento como positivo si el total es mayor o igual a cero, o negativo en caso contrario.

Es importante aclarar que en los resultados se pudo comprobar empíricamente lo expuesto en [BO69]: el ser humano tiende a expresar más frecuentemente términos positivos que negativos. Debido a esto, el valor de "accuracy" inicial del clasificador fue muchísimo más alto para las opiniones positivas que para las negativas, razón por la cual se le agregó una etapa de supervisión para detectar un umbral que maximice el número de opiniones positivas que obtienen un valor total mayor a ese umbral y a la vez maximice el número de opiniones negativas que obtienen un valor por debajo de ese umbral.

4.3. SentiWordNet

La idea de este método surge a partir de [BS10], donde se describe a SentiWordNet, un diccionario léxico que aumenta a WordNet[MBF+90] con información de orientación semántica: para cada synset[4], provee una terna de scores numéricos entre 0 y 1, indicando cuán positivos, negativos y objetivos son los términos incluidos en el synset en relación a la opinión/percepción/semántica asociada a cada uno de ellos (la suma de los tres valores es 1). Teniendo esto en cuenta, puede hacerse un buen aprovechamiento de esta información haciendo una selección apropiada de los términos que puedan llegar a ser más relevantes para definir la orientación semántica del documento.

Es importante aclarar que SentiWordNet está disponible sólo para el idioma inglés, pero de todas maneras, dado que los traductores disponibles en la actualidad se comportan de manera muy precisa para palabras o frases cortas, se optó por utilizar este método realizando una traducción previa del término para el cual se desea calcular la orientación semántica, sin que esto haga decaer el valor de "accuracy" del método.

4.3.1. Extracción de términos

De la misma manera que en SO-PMI, se selecciona para cálculo de orientación semántica a aquellos términos que contengan adjetivos o adverbios. A diferencia de SO-PMI, se evalúa solamente a la palabra en cuestión, ya que se asume que como se está promediando el valor del synset (proceso detallado en la siguiente sección), se está teniendo en cuenta la variación que pueda existir entre contextos semánticos. El cuadro 4.3 presenta un ejemplo que ilustra tal situación: en ambos casos el adjetivo se utiliza para enfatizar lo inusual de la característica que se está calificando, pero en el primero es aquella que genera asombro positivo, mientras que en el segundo lo que genera la falta de credibilidad es un hecho negativo.

[4]Conjunto de palabras agrupadas por su similitud semántica (sinónimos)

Scores	Definición
(+): 0.5 (-): 0 (**Obj**): 0.5	beyond belief or understanding; "the book's plot is simply unbelievable"
(+): 0.25 (-): 0.375 (**Obj**): 0.375	having a probability too low to inspire belief ; "an unbelievable and sad attempt of a football player"

Cuadro 4.3: Ejemplo de synset de SentiWordNet: presenta scores para los distintos contextos de la palabra "unbelievable".

4.3.2. Cálculo de orientación semántica de los términos extraídos

La idea inicial es básica: para cada adjetivo extraído, calcular la diferencia entre el score positivo otorgado por SentiWordNet y el score negativo. Es importante aclarar que el score para cada orientación es un promedio de todos los synsets del adjetivo en cuestión; esto se hace para obtener un valor lo más absoluto posible en términos del contexto semántico.

Además, luego de hacer un análisis de las oraciones se decidieron agregar otros factores que ayudasen a mejorar la precisión del verdadero valor semántico de cada adjetivo, presentados a continuación:

4.3.2.1. Intensificación adverbial

Intuitivamente, está claro que los adverbios, utilizados antes de un adjetivo, suelen profundizar o suavizar su fuerza calificativa. Por ejemplo, *"la trama fue buena"* es una afirmación semánticamente menos positiva que *"la trama fue **muy** buena"*, mientras que *"el plato fue abundante"* es claramente más positivo que decir *"el plato fue **realmente poco** abundante"*. Por esta razón se ha decidido incluir un intensificador de la puntuación otorgada a los adjetivos. Para saber el nivel de intensificación, se decidió por el armado de un mini-diccionario manualmente, dado que no hay ningún recurso disponible en la actualidad y que la cantidad de adverbios intensificadores es

relativamente baja. Cada entrada del diccionario contiene al adverbio, y un valor entre -1 y 1 que indica el nivel de intensificación (por ejemplo, *poco* tiene como valor -0.4, *espectacularmente*, 1 y *normalmente*, 0). Una vez armado el diccionario, para cada adjetivo a clasificar se analizan las palabras que lo preceden, y por cada predecesor inmediato que sea adverbio, si existe una entrada en el diccionario, se actualiza el score del adjetivo a_1:

$$Score(a_1)_i \leftarrow Score(a_1)_{i-1} + Score(a_1)_{i-1} * Int\_value(adv_i) \tag{4.6}$$

donde i indica el $i - esimo$ adverbio inmediatamente precedesor de a_i. A modo de ejemplo, *"gracioso"*, que se traduce a *"funny"* en inglés, tiene como score 0.25, y en el diccionario generado manualmente, *"muy"* tiene un valor intensificante de 0.6 (60 %), por lo tanto el score del término *"muy muy gracioso"* se calcula en dos pasos; primero el algoritmo detecta al segundo "muy", y reemplazando en (4.6),

$$\begin{aligned} Score(\text{“}gracioso''\text{)}_1 &\leftarrow Score(\text{“}gracioso''\text{)}_0 + Score(\text{“}gracioso''\text{)}_0 * Int\_value(\text{“}muy''\text{)} \\ &= 0{,}25 + 0{,}25 * 0{,}6 \\ &= 0,4 \end{aligned}$$

luego detecta el primer "muy" y vuelve a intensificar el score obtenido hasta el momento,

$$\begin{aligned} Score(\text{“}gracioso''\text{)}_2 &\leftarrow Score(\text{“}gracioso''\text{)}_1 + Score(\text{“}gracioso''\text{)}_1 * Int\_value(\text{“}muy''\text{)} \\ &= 0{,}4 + 0{,}4 * 0{,}6 \\ &= 0,64 \end{aligned}$$

De esta forma se obtiene un resultado final que representa de mejor manera la orientación semántica de la oración en cuestión.

4.3.2.2. Cambio de polaridad

El algoritmo inicial sólo busca adjetivos y para cada uno de ellos calcula su puntuación. El problema es que hay casos donde el adjetivo en realidad

está siendo negado, y eso no se está teniendo en cuenta hasta aquí. Detectar los casos donde esto ocurra no es una tarea trivial, y las opciones a considerar son tres[TBT+11]:

1. Generación y análisis de un árbol de dependencias
2. Backtracking hasta encontrar un conector (pero, aunque, y, etc) o un símbolo de puntuación.
3. Búsqueda dentro de una ventana.

La primer alternativa es la más "correcta", pero las pruebas que se realizaron para obtención de árboles de dependencia en sentencias en español no arrojaron buenos resultados. Respecto de 2 y 3, en las experimentaciones ha sido la búsqueda dentro de una ventana fija la que brindó mejores resultados, por lo que será la adoptada en los experimentos finales. Suponemos que la razón de esto es que ciertos conectores, como la coma, suelen aparecer entre la negación y el adjetivo sin necesariamente estar conceptualmente separados, como en la oración *"Nadie, ni siquiera el dueño, lo considera bueno"*, y en casos como ese el backtracking se detiene antes de encontrar la negación.

Una vez detectada una oración que está siendo negada, las alternativas son dos[TBT+11]:

- **Inversión del score**: esta es la más intuitiva, pero realmente no es la mejor para muchos casos[LS09]. Por ejemplo, si *"muy buena"* tiene un score de 0.8 y *"mala"* un score de -0.7, si el enfoque fuera invertir el score ante una negación para *"no muy buena"* obtendríamos -0.8, lo cual no sería representativo, ya que ese término tiene una orientación mucho menos negativa que *"mala"*.
- **Shifting del score**: aquí lo que se hace es sumar (o restar, según corresponda) un valor fijo al score, de esta manera, si fijásemos ese valor en 0.6, *"no muy buena"* tendría un valor final de 0.2, lo cual suena mucho más razonable que en el caso de inversión de score.

4.3.2.3. Bloqueo de oraciones en tiempo subjuntivo

El idioma español hace un uso extenso del tiempo subjuntivo para expresar oraciones en sentido potencial: *"Si el postre hubiera sido sabroso, la cena habría sido perfecta"* o *"Esperaba que fuera tan buena como la primera parte"*. Se decidió ignorar por completo estas oraciones (utilizando la información morfosintáctica de las oraciones: aquellas que posean verbos en este modo, se ignoran), ya que muchas veces no es claro -ni siquiera para un supervisor humano- si estos casos aportan positiva o negativamente al score global del documento.

4.3.3. Clasificación del documento

El comportamiento de este método es similar al de SO-PMI: para cada documento obtiene el score de cada término de interés (en este caso incluyendo todos los "decoradores" mencionados en la sección anterior) y realiza la suma total para clasificar al documento como positivo si esa suma lo es, o negativo en caso contrario. Cabe aclarar que para este clasificador se realizará el mismo proceso de detección de umbral para balancear entre scores positivos y negativos.

Capítulo 5

Corpus

El corpus de prueba debía cumplir las siguientes características:

- Estar en idioma español.
- Contener revisiones de productos/servicios con algún tipo de puntaje asociado a cada una de ellas
- Facilidad de extracción del texto de la opinión y el puntaje asociado.
- Diversidad de dominios (ej: opiniones de hoteles, libros, artículos electrónicos, etc), con una cantidad razonable de opiniones en cada uno (mínimo 1000), y balanceada en términos de las calificaciones.
- Opiniones de al menos uno o dos párrafos o 200 palabras, y bien escritas (pocos errores de ortografía, correcta estructura gramatical).

El lugar más accesible para obtener el corpus deseado es la web, y muchos sitios (cualquiera que sea de opiniones de algún tipo) cumplen con las primeras tres características. El problema principal son la cuarta y quinta, ya que hay pocos sitios en español que posean una buena cantidad de opiniones de dominios diferentes; y obtenerlas de distintos sitios (uno por cada dominio), tiene el problema de que las pruebas multidominio se verían seriamente afectadas por la calidad de opiniones de cada sitio: si uno tomase,

por ejemplo, los comentarios y calificaciones de películas de usuarios de cinesargentinos.com y luego para el dominio de artículos electrónicos un sitio de opiniones profesionales, cualquier clasificador funcionará mucho mejor en el segundo caso que en el primero por el simple hecho de la riqueza del texto, independientemente del nivel de dificultad propio del dominio[1]. Además, los resultados de las pruebas de entrenamiento en un dominio y testeo en otro serían también excesivamente sensibles a la calidad de cada grupo.

El sitio ciao.es es uno de los primeros sitios de opiniones en español, con una vasta cantidad disponible para diversos dominios. Una gran ventaja de este sitio es que la calidad de las opiniones es generalmente homogénea independientemente del dominio, lo que lo convierte en el candidato ideal para conformar el corpus sobre el cual se experimentará.

Para "crawlear" el sitio y obtener las opiniones junto con la puntuación para conformar el corpus de experimentación, se utilizó HtmlCleaner[2], una librería que toma como entrada código HTML y provee una estructura de árboles similar al Document Object Model tradicional, permitiendo una fácil navegación por el documento. Utilizando esta librería e inspeccionando un poco la estructura DOM[3] de las páginas de ciao (que para facilidad del proceso se mantiene constante a lo largo de las distintas páginas y dominios de opiniones), se pudieron generar con relativa facilidad archivos de texto plano conteniendo cada opinión y cuyo nombre está compuesto por la cantidad de estrellas otorgada al objeto de opinión y el nombre del mismo (ej: *3_CanonPowershotA320*). El proceso se inicia en el directorio raíz del dominio a crawlear, y funciona como una araña tradicional, siguiendo los links de interés hasta llegar a cada una de las opiniones. Al terminar, el proceso

[1] Las características propias de cada dominio inciden en el grado de dificultad de la clasificación. Por ello, se analizaron algunas opiniones de los distintos dominios y de acuerdo a lo que uno supone a priori, se pudo ver que las de películas por ejemplo, tienen una alta frecuencia de adjetivos que no son necesariamente indicadores de orientación semántica: *"trama impredecible"*, *"actuación impactante"*, *"final inesperado"*; mientras que las opiniones de, por ejemplo, artículos electrónicos, tienen en general mayor frecuencia de buenos orientadores: *"excelente definición"*, *"muy buen precio"*, *"diseño exquisito"*.

[2] http://htmlcleaner.sourceforge.net/

[3] Domain Object Model, que es el modelo que define la estuctura de las páginas HTML.

deja todos los archivos (uno por opinión) en un directorio determinado, que es luego utilizado de diversas maneras para realizar las experimentaciones.

El puntaje asignado por el usuario en cada opinión es extraído fácilmente a partir del DOM. La escala de puntajes es de 1 a 5 estrellas, y a efectos de este trabajo las opiniones de 1 y 2 estrellas son consideradas negativas y las de 3, 4 y 5, positivas. A continuación se muestra un ejemplo de una opinión de 4 estrellas en el dominio de hoteles:

> *He conocido recientemente el Hotel NH Argüelles que está en la madrileña calle de Vallehermoso, muy cerca de la Plaza de Cristo Rey. Es un hotel de 3 estrellas, correcto, con una aceptable relación calidad/precio, teniendo en cuenta lo que cuestan los hoteles en Madrid. Tiene buenos detalles como persianas automáticas (suben y bajan con un botón), canal plus, secador de pelo, minibar, prensa diaria gratuíta, y para mi gusto, lo mejor, el set de baño, que además de ser completísimo (peine, set de afeitado, kit dental, gel, champú, crema hidratante) es de alta calidad y tiene una fragancia a frutas cítricas deliciosa, mejor que algunos productos de marca que se compran en perfumerías. El mobiliario y la decoración están bien, son nuevos y cuidados, en colores azules y marrones coordinados, la cama es enorme, firme, y lo más llamativo: es altísima. No sé cómo hará para subirse ahí una persona mayor o con problemas de movilidad, si te caes de ahí dando vueltas por la noche tienes altas probabiliaddes de partirte el cuello ;P Bromas a parte, la habitación está bien: buena iluminación, la calefacción funciona bien, la bañera cuenta con media mampara...Pero el hotel tiene también algunos inconvenientes; Por ejemplo, la zona donde está es realmente complicada para aparcar el coche (hay muy poco espacio) y el parking del hotel es carísimo (12 euros al día si no recuerdo mal). Otra cosa a tener en cuenta, me pareció un poco pequeña la habitación y el baño, todo un poco justito, y además la habitación olía un poco a tabaco cuando llegué, a pesar de que la mujer de la limpieza había dejado el balcón abierto. El desayuno buffet no lo probé, pero debía ser bueno porque costaba también lo suyo: otros 12 euros por persona, y no ofrecen más comidas, en todo caso tienen concertado un servicio con Pizza Hut (creo) para recibir a domicilio en el hotel la pizza, pero claro, te sale al doble que si la pides en tu casa. En líneas generales no está mal, pero tiene algunos detalles que le hacen bajar en la valoración general. El precio por noche es de 50 euros.*

En general las demás opiniones son similares a esta, lo que permitió confeccionar un corpus de una calidad más que aceptable: los documentos son de una extensión similar a la de los experimentos realizados en los trabajos referenciados aquí ([PLV02], [**?**], [FC08], etc) y están mayormente correcta-

mente escritos tanto semántica como sintácticamente.

Características del corpus por dominio

A continuación se presenta un cuadro con las características más relevantes de las opiniones de cada dominio en el corpus utilizado.

Dominio	**# Documentos**	**PPD**	**PAD**
Prod. Electrónicos	488 pos. - 492 neg.	386	19
Hoteles	2234 pos. - 2021 neg.	436	22
Videojuegos	992 pos. - 1028 neg.	325	17
Películas	633 pos. - 671 neg.	529	29

Cuadro 5.1: Características del corpus por dominio
PPD: Cantidad de palabras promedio por documento
APD: Cantidad de adjetivos promedio por documento

Capítulo 6

Experimentación y análisis de resultados

Esta sección se desarrollará de la siguiente manera:

- En la primera etapa de experimentación analizaremos cada método de forma individual y dentro de un mismo dominio, para 4 dominios diferentes.
- La segunda etapa consistirá en la evaluación y análisis del comportamiento de los métodos de Machine Learning cuando son entrenados con un dominio pero evaluados con otro.
- Luego se analizarán los resultados de la implementación de la sección "Extracción de características comunes" (3.5).
- En la última etapa se presentará la evolución del método de Machine Learning que resulte más efectivo cuando se entrena con más de un dominio y evalúa con uno distinto.

6.1. Análisis individual de métodos, intra-dominio

En esta sección se presentarán y analizarán los resultados[1] de la aplicación de los distintos métodos a 4 dominios diferentes: productos electrónicos, videojuegos, películas y hoteles. Estos dominios han sido seleccionados debido a sus diferentes características semánticas, principalmente para hacer un análisis más abarcativo del comportamiento de cada método pero también para poder evaluar de la mejor manera posible el problema de la dependencia de dominio en los métodos de Machine Learning. Con respecto a los conjuntos de entrenamiento y evaluación, se hizo para todos los casos cross-validation en 10 folds, con 90 % de opiniones para entrenamiento y 10 % para evaluación en cada uno de ellos.

Con respecto a la implementación, para los casos de Machine Learning existen las librerías Weka[2] y Mallet[3], que proveen implementaciones de varios algoritmos de Machine Learning, incluidos los que se utilizarán en esta tesis. Para Naïve Bayes y SVM se utilizaron los algoritmos provistos en Weka y para Maximum Entropy el de Mallet (debido a que tiene una implementación más rápida y con mejores resultados que Weka). Por otro lado, para los casos de orientación semántica la implementación se hizo a mano.

6.1.1. Machine Learning

Para estos métodos se decidió por no realizar ningún tipo de feature selection o preprocesamiento: analizar los resultados de las combinaciones de las distintas alternativas disponibles para cada método y para cada dominio es una tarea prolongada que está fuera del alcance de esta tesis, y que en todo caso se puede encarar ya conociendo el comportamiento de los métodos "baseline" que sí serán analizados aquí. Evaluaremos las opiniones de 3 maneras:

[1] Utilizando "accuracy" como medida - elegida por ser la más representativa de la eficacia de los métodos, más aún teniendo en cuenta que no se realiza ningún tipo de selección de rasgos, por lo que precisión y recall dan resultados muy similares.

[2] http://www.cs.waikato.ac.nz/ml/weka

[3] http://mallet.cs.umass.edu

I. **Sin modificaciones**: cada opinión es tratada tal cual se ha obtenido.

II. **Stemming**: Utilización del Snowball Stemmer para español[4] para obtener los stems de las palabras y utilizar a ellos en la etapa de entrenamiento para obtener los pesos que serán utilizados en la etapa de evaluación. La ventaja que se supone a priori al realizar este proceso es que distintas variaciones de buenos indicadores de orientación semántica pertenecerán al mismo stem, aumentando el peso del mismo (por ejemplo, *excelente, excelentísimo, excelentes, excelentemente*, etc pertenecen a *excelent*).

III. **Stemming + Stopwords**: Al proceso de stemming le agregamos la exclusión de palabras comunes [5] para evitar que sean seleccionadas como "features" e influyan en los scores, hecho que es recomendable evitar dado que son palabras que no tienen ningún tipo de orientación semántica en ningún contexto[6].

[4] http://snowball.tartarus.org/algorithms/spanish/stemmer.html

[5] Artículos, preposiciones y otros provenientes de esta lista: *ante, antes, desde, el, la, lo, las, los, aqui, dentro, cuando, donde, con, entre, con, sin, yo, tu, su, aquel, de, para, por, que, porque, por que*. La preposición *bajo* no se incluye en la lista debido a que puede tener significado semántico.

[6] A priori suponemos que esto será beneficioso en Naïve Bayes solamente, ya que es el único método basado principalmente en la frecuencia de aparición de palabras

6.1.1.1. Naïve Bayes

Como se mencionó al describirlo en la sección 3.2, a pesar del fuerte supuesto de independencia entre "features", este método ha sido aplicado con éxito en clasificación de texto tradicional y también en minado de opinión de inglés. La tabla 6.1 presenta los resultados de la experimentación con opiniones en español del corpus preparado para esta tesis.

Dominio	**Sin modif.**	**Stem**	**Stem+Stopword**
Prod.Electrónicos	79.5 %	81.5 %	82.78 %
Videojuegos	81.5 %	82.85 %	84.97 %
Películas	80.13 %	81.47 %	81.94 %
Hoteles	88.82 %	89.8 %	89.96 %

Cuadro 6.1: "Accuracy" de Naïve Bayes dentro de un mismo dominio

Los resultados han sido especialmente precisos para el caso de Hoteles, y en todos los casos considerar sólo los stems y eliminar stopwords ha sido beneficioso - este es un resultado esperable ya que al ser un modelo basado principalmente en la frecuencia de aparición de palabras, agrupar aquellas con un mismo stem y eliminar stopwords, hacen que cada feature seleccionada sea más representativa y significativa. Los mejores resultados en el caso de hoteles se deben a dos razones:

- Principalmente, a que a diferencia de los otros dominios, el conjunto de características de un hotel es bastante acotado (ubicación, amplitud de habitación, antigüedad, limpieza), por lo que las opiniones en general van en torno a estas características, facilitando la tarea de "modelado" de opiniones de cada categoría en la etapa de entrenamiento: las opiniones utilizadas en la etapa de evaluación tendrán un nivel de similitud mayor que en casos donde las características del objeto que está siendo evaluado son mucho más diversas (por ejemplo, en productos electrónicos).

- En segundo lugar, las opiniones de este dominio están -en general- mejor expresadas, con menos errores de ortografía y descripciones más

detalladas.

Características de los "features" extraídos A continuación analizaremos las características del top 20 de "features" con peso más alto para cada categoría (eliminando stopwords) dentro del dominio de videojuegos[7]. Los cuadros 6.2 y 6.3 presentan los resultados.

Feature	**Peso**
estrategia	0.0027
misiones	0.0024
puedes	0.0024
armas	0.0023
gran	0.0022
tiempo	0.0022
mejor	0.0022
poder	0.0022
tambien	0.0022
jugar	0.0022
tipo	0.0021
mundo	0.0021
vez	0.0021
historia	0.0021
unidades	0.0020
enemigos	0.0020
muchos	0.0020
bien	0.0020
todas	0.0020
primera	0.0020

Cuadro 6.2: 20 "features" con mayor peso en opiniones positivas

Como puede apreciarse en las tablas, en ambas categorías hay una importante presencia de palabras específicas del dominio, que no necesariamente son indicadoras de orientación: juego, misiones, armas, jugar, sonido, etc; otro grupo de palabras muy comunes en el vocabulario cotidiano (pero no

[7]No se presentan los resultados para los demás dominios porque las tendencias fueron similares, hecho que también se repite más adelante en MaxEnt y SVM.

Feature	**Peso**
nada	0.0036
verdad	0.0030
graficos	0.0029
bien	0.0028
peor	0.0028
mal	0.0028
juegos	0.0026
jugabilidad	0.0025
sonido	0.0025
jugar	0.0025
vez	0.0024
todo	0.0024
aburrido	0.0024
tan	0.0024
mejor	0.0023
os	0.0023
todos	0.0022
creo	0.0022
dinero	0.0021
calidad	0.0021

Cuadro 6.3: 20 "features" con mayor peso en opiniones negativas

al punto de ser consideradas stopwords): nada, poder, tipo, vez, etc; y sólo en el caso de las negativas aparece una palabra inequívocamente indicadora de orientación semántica: aburrido. Este es un resultado esperable, ya que Naïve Bayes es un modelo probabilístico basado exclusivamente en la frecuencia de las palabras, por lo que se pueden dar casos como el de "verdad", una palabra que a priori no indica ningún tipo de orientación semántica en el dominio de videojuegos, pero que es la segunda palabra más frecuente entre las opiniones negativas y no aparece ni en el top 20 de las positivas. Un problema directo de esta situación es que está indicando una alta dependencia de dominio, por lo que suponemos que un clasificador NB entrenado en un dominio y testeado con otro verá su performance seriamente afectada (ver sección para discusión y resultados de esto).

Como segundo análisis, es interesante ver si palabras que a priori son indicadoras de orientación semántica tienen mayor probabilidad asignada en la categoría correspondiente. El cuadro 6.4 presenta una comparación de dichos valores, donde se puede comprobar que en todos los casos, la orientación que uno asignaría a priori es la que efectivamente tiene mayor probabilidad. Un aspecto interesante a remarcar es que debido a que este método se basa exclusivamente en frecuencia de aparición se dan casos como el de malo y excelente: ambas tienen mayor peso en la categoría correspondiente, pero al ser "malo" una palabra utilizada más comunmente para dar connotación negativa que "excelente" para dar connotación positiva, el peso negativo asignado a la primera es mucho más alto que el positivo asignado a la segunda.

Feature	**Peso Neg**	**Peso Pos**
malo	**0.0026**	0.0009
excelente	0.0004	**0.0010**
aburrido	**0.0025**	0.0004
entretenido	0.0010	**0.0013**
corto	**0.0011**	0.0007
genial	0.0004	**0.0010**

Cuadro 6.4: Pesos para fuertes orientadores semánticos

Es importante volver a destacar que de estos resultados se desprende una aparente fuerte dependencia de contexto: los "features" con mayor nivel de pertenencia a una u otra clase no son intuitivamente indicadores "universales" de orientación semántica como los expresados en 6.4, por lo que a pesar de su simpleza y muy buenos resultados intra-dominio, es altamente factible que no sea un método apropiado para un clasificador más universal.

Casos de error En algunos casos donde la clasificación es errónea, es realmente complicado incluso para un ser humano elegir la correcta, y a continuación se presentan algunos ejemplos:

> *"Me compre este mp3 hace poco, por que pense es de SONY y hay garantia de calidad, pero una vez q lo probe note que tiene un siseo que es audible en plena reproduccion de las canciones y radio, el uso del sonicstage es engorroso y obligatorio, haciendo a un lado estos defectos la bateria carga muy rápido el diseño es elegante y a alto volumen no se nota mucho el siseo, los audifonos con los que vienen son de gama baja asi que necesitan unos de gama alta para apreciar mejor el sonido q es lo mejor aunque no tenga MEGABASS. Como todos los productos WALKMAN hasta la fecha estos no graban voz ni tampoco lo que escuchas de la radio."*

> *"Es un reproductor muy cómodo y práctico que cabe en cualquier bolsillo y la calidad de su pantalla es excelente al igual que su radio, pero no es oro todo lo que reluce. Le encuentro varios fallos: - La carcasa es débil, a los dos dias se me desmonto la pieza de la botonera sin haberle dado ningun golpe ni nada.- La forma de sujección de los auriculares con el reproductor es básicamente un hilo cutre enganchado a una anilla medio suelta que sale de dentro del reproductor.- La calidad de la pantalla es excelente, y es muy práctica para la oscuridad, pero con el gran inconveniente que al ser de luz, no se apaga (como pudiera pasar con otros reproductores), por lo que siempre esta en azul/amarillo y gastando batería.- No dispone de bloqueo cuando esta apagado, para que no se encienda solo pero si dispone de bloqueo una vez encendido.- La mala calidad de los auriculares integrados. Pero tambien hay cosas buenas, como su reducido tamaño y peso, todas las prestaciones extras que tiene y la calidad de sonido, es muy limpia. Además trae un cargador adicional de corriente para no depender de un ordenador."*

En ambos casos la opinión es realmente ambigua: bien podría ser que la primera es negativa y la segunda positiva, pero es al revés (en la primera el

score es 4 de 5 y en la segunda 2 de 5). En este tipo de casos, es esperable que este y cualquier otro clasificador falle, por más preciso que sea. A pesar de esto, de todas maneras hubo casos donde la opinión expresaba claramente la orientación correspondiente y sin embargo fue clasificada erróneamente. Los casos analizados mostraron que esto se debió, en general, a que la opinión no poseía un número considerable de los "features" con mayor peso en la categoría a la que pertenecía, por lo tanto la información extraída a partir del entrenamiento no podía ser aprovechada de la mejor manera en estos casos. El siguiente es un ejemplo de una opinión clasificada como negativa cuando en realidad es positiva (score 5 de 5) en el dominio de videojuegos, notar que sólo hay tres palabras del top 20 mostrado en el cuadro 6.2 (en negrita), y seis del cuadro 6.2 (subrayadas):

> *"Es super gracioso ver como llegas a la tierra siendo un angelito regordito y te tienes que meter en el cuerpo de alguien. Eso sí, como el 'alguien' te vea intentarlo, irá a por tí dandote patadas en el culo, je, je, je. Se trata de uno de esos juegos al estilo Half Life. Tendrás que arreglar el* ***mundo*** *(te manda Dios nada menos!) siguiendo sus* ***misiones****, y para ello te tienes que meter en el cuerpo de quien más te convega según el caso.Tienes varias* ***armas*** *para utilizar y el movimiento, la jugabilidad y los gráficos son muy buenos aunque, como en otros juegos, es posible que te quedes atrapada en alguna fase sin saber que leches hacer. No está nada mal.*

A pesar de estas falencias, por su simplicidad, este método es una alternativa muy viable si la clasificación se realizará intra-dominio.

6.1.1.2. Maximum Entropy

A continuación veremos los resultados obtenidos con el método de Maximum Entropy.

Dominio	**Sin modif.**	**Stopword**
Prod.Electrónicos	91.03 %	89.07 %
Videojuegos	94.44 %	92.52 %
Películas	89.66 %	83.41
Hoteles	95.32 %	92.37 %

Cuadro 6.5: "Accuracy" de Maximum Entropy dentro de un mismo dominio

Aquí también Hoteles es la de mejores resultados, y los resultados han sido siempre superiores a Naïve Bayes. Stopwords (las mismas que se utilizaron en Naïve Bayes) empeoró la performance en todos los casos, y esto se debe a que como este método no basa el cálculo de los pesos en base a la frecuencia de términos, eliminar aquellos que aparecen frecuentemente puede tener resultados completamente aleatorios, ya que ciertas stopwords pueden tener algún tipo de orientación semántica en determinados dominios (por ejemplo, *alrededor* para el caso de hoteles, puede tener orientación positiva si se utiliza en opiniones positivas para decir "alrededor del hotel había muy buenos restaurantes").

Los cuadros 6.6 y **??** presentan los 20 "features" con mayor score[8] en cada categoría dentro del dominio de videojuegos.

En este caso, a diferencia de Naïve Bayes, los "features" con mayor peso son -salvo excepciones- mucho más representativas de cada categoría: *encanta, genial, perfecto, impresionante* son ejemplos muy ilustrativos de opiniones positivas, mientras que *peor, aburrido, malo, decepcionado* son claros ejemplos de palabras presentes en opiniones negativas. Ciertas palabras indican que este método es realmente efectivo para extraer indicadores de categoría realmente aislados (en términos de frecuencia) y no intuitivos:

[8]Aquí score es el valor $\lambda_{i,c}$ para cada f_i y cada clase c, obtenido en la resolución del problema de optimización presentado en la sección de introducción a MaxEnt.

Feature	**Peso**
adictivo	0.52
estrategia	0.51
paciencia	0.44
misiones	0.44
armas	0.43
mejores	0.42
encanta	0.41
batallas	0.41
genial	0.39
sonidos	0.39
mundo	0.39
commandos	0.38
buenos	0.37
enganchado	0.35
crear	0.35
muchisimas	0.34
objetivo	0.34
fallout	0.32
perfecto	0.31
impresionante	0.31

Cuadro 6.6: 20 “features” con mayor peso en opiniones positivas

Feature	**Peso**
peor	0.86
consola	0.69
aburrido	0.68
lento	0.61
fifa	0.57
malo	0.56
nada	0.56
repetitivo	0.56
demasiado	0.47
mierda	0.47
compre	0.39
unico	0.39
decepcionado	0.39
mala	0.39
dinero	0.37
comprar	0.36
futbol	0.36
resto	0.36
pobre	0.36
decepcionante	0.32

Cuadro 6.7: 20 "features" con mayor peso en opiniones negativas

por ejemplo, en el caso de las positivas, aparecen palabras como *commandos, fallout*, que a priori no indican ningún tipo de orientación semántica, pero que están presentes en un número muy reducido de opiniones con esa orientación[9].

El aspecto más relevante de estos cuadros es que parecen ser particularmente útiles para la obtención de un conjunto de features "universal", el significado de la mayoría de los "features" que aparecen es, o bien suficientemente global como para ser buen indicador en un gran número de dominios (los ya mencionados *encanta, genial, perfecto, impresionante* para positivas y *peor, aburrido, malo, decepcionado* para negativas) o bien lo suficientemente específico como para no generar ruido al utilizarlas en otros dominios (*adictivo, commandos, fallout* en un caso o *consola, fifa* en el otro, que aparecen por el simple hecho de que se mencionan casi exclusivamente en una u otra categoría).

Casos de error Aquí en muchos casos el problema fue el mismo que en Naïve Bayes: o bien las opiniones eran muy ambiguas, o bien hubo un error de anotación - como el siguiente ejemplo, donde el juego fue calificado como negativo (1 estrella de 5) pero la opinión es claramente positiva:

> *"Este juego es especial para mi, esta muy bien, para mi es el mejor, es de estrategia y tienes que conseguir dinero, tienes que trabajar y ganarte el sueldo. Que opino: me gusta mucho sobre todo por los gráficos, tiene personas en 3D, puedes comprar puedes realizar vuelos etc... puedes hablar con la gente, es divertido, hasta que te juegas todas las pantallas con los 4 jugadores que tienes pasan meses, puedes elegir pantallas, jugar en red contra 4 personas, o sea 4 personas jugando a trabes de Internet, y lo mas importante, requiere menos velocidad que un Pentium 1, la verdad es que con el 1 te sobra velocidad, incluido con tarjeta grafica de 32 MB de ram, como el mío."*

Está claro que en este tipo de casos no hay nada que se pueda hacer, la clasificación realizada es la correcta, el problema aquí fue el creador de la

[9]Esto también podría verse como un aspecto negativo, ya que hace al clasificador más sensible a expresiones extraordinarias, pero suponemos que si están presentes mayormente en una de las dos categorías, es más probable que una aparición en el conjunto de evaluación indique pertenencia a la misma clase donde ocurrió en la etapa de entrenamiento que a la otra.

opinión, que la calificó erróneamente.

En los demás casos de error, el problema también es similar al de Naïve Bayes: las opiniones erróneamente clasificadas tenían un número considerable de palabras con pesos altos en la categoría incorrecta, como en el siguiente ejemplo, evaluado de manera negativa (1 estrella de 5) pero clasificado como positivo por MaxEnt (se ponen en negrita negrita las palabras con alto peso positivo, notar que ninguna del top 20 de negativas aparecen en esta opinión):

> *"Llegué a mi casa, y me dispuse a instalar el juego con la ilusión de encontrarme con una segunda parte tan* ***impresionante*** *como la primera. Luego de infinita* ***paciencia*** *esperando que concluya la instalación, lo inicias y te encuetras con una presentación digna de un juego de bajo presupuesto. Una vez que inicias el modo de un jugador te encuentras con una de las mejores partes del juego: el entrenamiento. Si ya jugaste la primera parte se te va a hacer bastante familiar, pero de todas formas completa la misión porque es la única parte del juego donde pasarás* ***buenos*** *momentos; por todo lo demás puedes tirarlo a la basura o devolverlo a la tienda.*

Nuevamente, no hay mucho margen de acción en este tipo de casos, pero es un problema claramente acotado por el mismo algoritmo de optimización del clasificador: los "features" que terminan con mayor peso luego de la etapa de entrenamiento, poseen ese peso porque aparecen más en una clase que en otra[10], por lo tanto casos como el de recién pueden darse solamente en un número limitado de casos: los "features" que pertenecían a la clase opuesta a la de la opinión (impresionante, paciencia, buenos) dejarían de pertenecer a ella si aparecen en más opiniones de la otra clase.

[10]No necesariamente con mucha frecuencia, con una sola aparición en una clase y no aparecer en la otra puede llegar a ser suficiente

6.1.1.3. SVM

Para este clasificador, se optó por la implementación que resuelve el problema de optimización para entrenamiento con el algoritmo *Sequential Minimal Optimization* y kernels polinomiales o RBF[Pla98]. La razón de elección de esta alternativa es su probado buen rendimiento e importantes mejoras en el tiempo de ejecución respecto de la implementación original, propuesta en [CV95]. Los resultados, en 6.8.

Dominio	**Sin modif.**	**Stem**	**Stem+Stopword**
Prod.Electrónicos	79.1 %	80.6 %	80.6 %
Videojuegos	82.46 %	84.5 %	83.89 %
Películas	78.2 %	78.47 %	78.84 %
Hoteles	85.4 %	87.29 %	87.05 %

Cuadro 6.8: "Accuracy" de SVM dentro de un mismo dominio

Se repite nuevamente la tendencia con hoteles, que ha sido el dominio más preciso. Stemming mejoró, y eliminación de stopwords tuvo resultados variables. Las mejoras en stemming, al igual que en Naïve Bayes, son esperables debido al agrupamiento de palabras con misma semántica. A juzgar por los resultados, los cambios en la representación vectorial de los documentos de entrenamiento provocados por la inclusión o exclusión de stopwords derivan en hiperplanos que no necesariamente son mejores en un caso u otro cuando se realiza el testeo y suponemos que esto es consecuencia de un resultado equivalente a lo expresado para MaxEnt: como este método no basa el cálculo de los pesos en base a la frecuencia de palabras, eliminar palabras que aparecen frecuentemente puede tener resultados completamente aleatorios, ya que ciertas stopwords pueden tener algún tipo de orientación semántica en determinados dominios. Comparativamente, tuvo resultados superiores a Naïve Bayes pero inferiores a MaxEnt.

Como se puede observar en los cuadros 6.9 y 6.10, con los distintos pesos de los features dentro del dominio de videojuegos, a diferencia de Naive Bayes, y similarmente a MaxEnt, el método le da peso a palabras que no necesariamente aparecen frecuentemente, pero si aparecen solo en una de las

dos categorias, pero extrae una menor cantidad de términos semánticamente universales en términos de indicación de pertenencia a opiniones de una u otra clase. En este caso los pesos en realidad indican las coordenadas de cada "feature" en el espacio vectorial, por eso los de la orientación negativa tienen valores negativos.

Feature	**Peso**
básicamente	0.97
directx	0.93
pequeño	0.92
todavia	0.88
clásico	0.87
violencia	0.86
pasada	0.86
editor	0.82
cumple	0.82
titulo	0.81
paciencia	0.78
disponibles	0.78
estrategia	0.78
doom	0.77
diferencia	0.76
encanta	0.75
duro	0.73
sonidos	0.73
impresionante	0.70
adictivo	0.64

Cuadro 6.9: 20 "features" con mayor peso en opiniones positivas

Casos de error Este método tuvo problemas con el mismo tipo de opiniones que MaxEnt. Para esos detalles, ver esta misma subsección en la sección de MaxEnt.

Feature	**Peso**
mierda	-1.31
caballo	-1.17
decepcionante	-1.16
quieren	-1.12
play	-1.11
repetitivo	-1.11
consola	-1.09
trabajo	-0.99
wii	-0.98
fallo	-0.97
control	-0.93
versiones	-0.92
street	-0.90
error	-0.85
lento	-0.84
interfaz	-0.83
anillo	-0.83
compreis	-0.82
salto	-0.78
decepcionado	-0.78

Cuadro 6.10: 20 "features" con mayor peso en opiniones negativas.

6.1.2. Orientación Semántica

En esta sección se analizarán los clasificadores de orientación semántica, evaluados con los mismos dominios que en los métodos de Machine Learning.

6.1.2.1. SO-PMI

Para evaluar este método se tuvo que reducir la cantidad de opiniones tenidas en cuenta debido a que la obtención de los scores es sumamente lenta. Analizando los resultados con los métodos de Machine Learning, en general a partir de 100 opiniones el resultado se empieza a estabilizar, por lo que se eligieron esa cantidad de opiniones por dominio para la evaluación de este método. El cuadro 6.11 presenta los resultados obtenidos.

Dominio	**"Accuracy"**
Prod.Electrónicos	65 %
Videojuegos	73.77 %
Películas	63 %
Hoteles	83.75 %

Cuadro 6.11: "Accuracy" de PMI dentro de un mismo dominio

Igual que en Machine Learning, Hoteles es la que mejores resultados obtiene, aunque en este caso la razón principal no es la cantidad reducida de características de ese dominio, sino más bien que las opiniones están mejor expresadas, por lo que las frases extraídas obtienen mejores resultados al hacer las búsquedas en el motor.

Casos de error En general, el problema más importante y degradante de performance de este método es que para frases que no son claramente indicadoras de orientación semántica los resultados son mezclados (por ejemplo, "*segunda parte* NEAR excelente" tiene 4030 resultados, mientras que la misma frase con "malo" sólo tiene 680), y el hecho de que la ocurrencia de este tipo de frases en general no indica orientación semántica hace que esta sea la razón principal por la cual se asignan clasificaciones erróneas, obteniendo

una performance que, al igual que como se verá en el caso de SentiWordNet, está bastante por debajo de lo que se puede obtener intra-dominio con los métodos de Machine Learning.

6.1.2.2. SentiWordNet

El cuadro 6.12 presenta los resultados obtenidos.

Dominio	**N**	**IA**	**IA+CP**	**IA+CP+ES**
Prod.Electrónicos	60.17 %	60.41 %	61.22 %	60.83 %
Videojuegos	65.32 %	65.70 %	65.86 %	65.89 %
Películas	66.05 %	66.36 %	66.51 %	66.12 %
Hoteles	72.81 %	74.01 %	74.13 %	73.89 %

Cuadro 6.12: "Accuracy" de SentiWordNet dentro de un mismo dominio. N=Sin adicionales, IA=Intensificación adverbios, CP=Cambio de polaridad, ES=Exclusión de frases en subjuntivo/condicional

Nuevamente el dominio de hoteles es el de mejores resultados, por las mismas razones que en SO-PMI. En todos los casos, considerar tanto intensificación adverbial como cambio de polaridad para términos que poseen una negación cercana han mejorado los scores. Excluir frases expresadas en modo subjuntivo o condicional no ayudó salvo en el caso de videojuegos (y con un margen muy pequeño). Esto se debe a que la mayoría de estas frases expresadas en potencial en realidad sí están expresando algún tipo de orientación, como los siguientes dos ejemplos:

- *"Si la habitación hubiese tenido vista al mar, hubiese sido perfecta. Pero bueno, tampoco se puede pedir todo"*.
- *"He pagado por un hostal pésimo, como si hubiese estado en un hotel de 3 estrellas"*.

En el primer caso se está ignorando el score que aporta *perfecta*, pero la oración tiene efectivamente una connotación positiva, por lo que al eliminarla se pierde esta información. En el segundo, pasa lo mismo pero en el caso negativo.

Casos de error A excepción de las optimizaciones agregadas, este método se basa exclusivamente en los valores obtenidos de SentiWordNet, y por ello cualquier error en los mismos afecta directamente el resultado de la clasificación. Luego de hacer un análisis de los resultados de las evaluaciones, se pudo verificar que en las que fueron erróneas hay un número considerable de adjetivos con valores equivocados. El cuadro 6.13 presenta algunos ejemplos que han provocado clasificaciones erróneas (recordar que "+" es el valor que provee SentiWordNet para indicar la orientación semántica positiva de la palabra, "-" la negativa y "Obj" la objetiva -es decir, sin orientación, algo factual-).

Adjetivo	**Traducción**	**Valor**
viejo	old	**(+)**: 0.375 **(-)**: 0.125 **(Obj)**: 0.5
antihigiénico	unsanitary	**(+)**: 0.625 **(-)**: 0.125 **(Obj)**: 0.25
confortable	comfortable	**(+)**: 0.250 **(-)**: 0.375 **(Obj)**: 0.375
barato	inexpensive	**(+)**: 0.000 **(-)**: 0.625 **(Obj)**: 0.375

Cuadro 6.13: Adjetivos con orientaciones erróneas. Los primeros dos tienen orientación positiva pero tienen una connotación real claramente negativa, y el tercero y cuarto, viceversa.

Otro problema es que como este método se basa exclusivamente en la presencia de adjetivos, hay cierto tipo de opiniones que no aportan información en ese sentido: principalmente aquellas que relaten experiencias vividas con el producto en lugar de las características, como el siguiente ejemplo:

> *"Cuando fui a comprármelo pensé que estaba comprando uno de los mejores reproductores de mp3 del mercado, lo cual fue un error ya que lo he tenido que llevar al servicio técnico tres veces y ya desconfío de que me lo hayan arreglado. La primera vez se me borraron todas las canciones y no me dejaba meter mas diciendo que estaba lleno. La segunda vez no me cargaba la batería durandome unas dos horas solamente, cuando me lo arreglaron me lo devolvieron sin radio y me lo tuvieron que llevar otra vez. La tercera vez cada vez que encendía la radio se me apagaba el mp3. Ahora todo parece solucionado pero desconfío porque conozco a tres personas que lo compramos y a las tres les ha pasado lo mismo."*

Esto se repite en muchos casos en distintos dominios, y la única forma de tener en cuenta la información semántica en este tipo de opiniones es considerando también verbos y sustantivos, pero para estos el problema de valores erróneos en SentiWordNet se potencia aún más que para adjetivos (se intentó escalar los valores devueltos para verbos y sustantivos dividiéndolos por distintas constantes pero en ningún caso se mejoró el score que sólo con adjetivos). Por estas razones, este método termina teniendo una performance inferior a los de Machine Learning entrenados y evaluados dentro de un mismo dominio.

6.1.2.3. Conclusión resultados Orientación Semántica

Como se vio en las respectivas tablas, los métodos presentados no tienen muy buena performance si se los compara con los resultados intra-dominio de los métodos de Machine Learning. De todas maneras, como estos métodos no poseen ningún tipo de dependencia de dominio (ya que la etapa de "entrenamiento" es previamente realizada de alguna manera por el buscador o por el diccionario de SentiWordNet; en el primer caso, la cantidad y diversidad de datos accesibles a través de un motor de búsqueda hace que las búsquedas por cercanía propuestas en el método que lo utiliza abarque un amplio espectro de contextos y dominios, y en el segundo, el "entrenamiento" es el ya realizado armado manual del diccionario semántico de SentiWordNet), pueden llegar a ser realmente útiles si no se posee un conjunto de entrenamiento que pueda ser utilizado en un clasificador de Machine Learning, o si el clasificador del método de Machine Learning a utilizar fue entrenado con un dominio muy distinto al que se desea clasificar.

6.2. Generalización multidominio

Como se pudo ver en el análisis intra-dominio, la asignación de pesos a "features" de los métodos de Machine Learning es dependiente del dominio con que se los entrena (Naïve Bayes pareció ser el que más sufre este problema y MaxEnt el que menos), con lo cual suponemos que su performance decaré al evaluar documentos de un dominio distinto.

La generalización planteada en la sección "Extracción de características comunes" (3.5) es sólo posible de realizar en Naïve Bayes, ya que es el único método que asume independencia entre los features, por lo que es posible agregar o quitar los que se considere necesarias del conjunto utilizado para evaluar, además de que los otros dos métodos realizan un proceso de optimización de los scores para los "features" obtenidos del conjunto de entrenamiento, con lo cual realizar algún cambio de valores en los parámetros obtenidos (o agregar uno nuevo) a partir de la información extraída de documentos de un nuevo dominio demandaría volver a ejecutar el proceso de optimización, lo cual es equivalente a simplemente considerar a los corpus de entrenamiento de cada dominio como uno solo, y esa alternativa será evaluada en la siguiente sección.

El siguiente cuadro presenta, dada la aplicación del algoritmo a dos dominios, los resultados de la evaluación sobre un tercer dominio con el conjunto de "features" standard y el conjunto "optimizado". Presentamos aquí los resultados de sólo un par de dominios de entrenamiento (Videojuegos y productos electrónicos) evaluado con los dos restantes. Es importante destacar que los resultados sin optimización ya de por sí son muy pobres, hecho que confirma la suposición de alta dependencia de dominio de Naïve bayes. Las tendencias presentadas se repitieron en otras combinaciones evaluadas: la diferencia entre no realizar el proceso aquí propuesto y sí realizarlo no representó en ningún caso una mejora mayor al 3 %.

Como vemos, la mejora existe, pero es sólo marginal. Es importante aclarar también que se ha tenido en cuenta la posibilidad de realizar más iteraciones con más dominios para analizar la potencial convergencia a un conjunto "universal" de "features", pero los resultados iniciales mostraron

Dominio	**Sin optimización**	**Con optimización**
Películas	55.32 %	57.47 %
Hoteles	58.44 %	59.42 %

Cuadro 6.14: Resultados con universalización de "features"

que se pierde mucha información en cada iteración del algoritmo debido a la eliminación de "features" con orientación semántica relativamente absoluta pero que para un dominio en particular influyen negativamente, entonces son eliminadas. Por ejemplo, "arcaico" tiene connotación generalmente negativa, puede ser utilizado de esa manera tanto para describir a un producto electrónico como para una habitación de un hotel, pero hay opiniones de videojuegos que describen al mundo o época donde se desarrolla utilizando esa palabra, sin que esto indique orientación negativa, por lo que incluir esta feature como indicadora negativa en el conjunto universal hace que el score cuando se corre el clasificador entrenado para videojuegos con esta feature agregada empeore, y el resultado de esto es la exclusión de esta palabra del conjunto universal cuando en todos los dominios menos uno el peso hubiese sido apropiado.

Debido a lo recientemente mencionado, a los resultados altamente superiores obtenidos con MaxEnt dentro de un mismo dominio, y a que suponemos que a partir de los resultados vistos en la evaluación intra-dominio de MaxEnt, el método regula los pesos de los "features" semánticamente más universales de manera apropiada (y no supervisada), se decidió no profundizar en la alternativa de esta sección y evaluar en la siguiente sección el comportamiento de MaxEnt a medida que se van agregando dominios al corpus.

6.3. Evolución en relación a la ampliación del corpus

Como vimos en la experimentación intra-dominio, MaxEnt y SVM son dos métodos que extraen una buena cantidad de "features" que son indicadoras aparentemente universales de orientación semántica, por lo que suponemos que al ir agregando documentos de distintos dominios, los"features" que se mantengan con mayor peso orientativo de cada clase serán los "universales", y los que sean específicos de cada dominio irán siendo "empujados" hacia posiciones más bajas, influyendo de manera decreciente en aquellos dominios donde su presencia no es necesariamente indicadora de algún tipo de orientación (o aún peor, donde su presencia indica pertenencia al dominio opuesto). En esta sección se analizarán los resultados de experimentos realizados en esta dirección. Debido a que los features con mayor peso de cada categoría en MaxEnt fueron las que parecían más independientes de dominio, decidimos presentar aquí los valores obtenidos con ese clasificador (sin eliminar stopwords, ya que como se vio en la sección de resultados, fue la alternativa que obtuvo mejores resultados intra-dominio).

6.3.1. Resultados MaxEnt

El cuadro 6.15 muestra los resultados para los diversos dominios. Se evaluó a cada uno por separado, y para cada caso se entrenó al clasificador a utilizar con uno, dos o tres dominios distintos al que se desea evaluar, utilizando la misma cantidad de instancias (1000) para cada uno de ellos para evitar excesiva influencia de los que poseen mayor cantidad de opiniones. En el caso de uno y dos dominios de entrenamiento, se realizaron todas las combinaciones para la cantidad correspondiente y se promedió el resultado (por ejemplo: dado un dominio, si el resultado de evaluarlo con el clasificador entrenado con cada uno de los otros dominios fue de 60 %, 70 % y 80 % de "accuracy" respectivamente, entonces el valor para ese dominio con un solo dominio de entrenamiento es de 70 %).

Salvo en el caso de videojuegos, agregar dominios ha mejorado el valor

Dominio test	**1 dom. de entrenamiento**	**2**	**3**
Prod.Electrónicos	70.8 %	73.16 %	74.2 %
Videojuegos	66.43 %	67.7 %	66.5 %
Películas	66.1 %	69.23 %	70.9 %
Hoteles	74.5 %	78.3 %	80.7 %

Cuadro 6.15: "Accuracy" de MaxEnt con múltiples dominios de entrenamiento

de "accuracy" . Como ya se mencionó, para los casos de 1 y 2 dominios de entrenamiento, los resultados expuestos en el cuadro reflejan el promedio de los resultados realizando todas las combinaciones. Como es de esperarse, los resultados para cada una de estas elecciones varían de acuerdo al nivel de semejanza entre los dominios: al evaluar opiniones de videojuegos, si el entrenamiento se realiza con opiniones de productos electrónicos, el valor de "accuracy" es del 64.7 %, con opiniones de hoteles es del 61.6 % y con opiniones de películas, del 73 %[11]. Esto obviamente se ve reflejado en los "features" con más peso de cada categoría: como se puede ver en los cuadros 6.16 y 6.17, en el caso de películas, los "features" son mucho más compatibles con opiniones de videojuegos que en los otros dos.

Otro aspecto a analizar es qué pasa con los pesos de los "features" cuando se agregan dominios. El cuadro 6.18 muestra los 10 "features" con mayor peso de cada categoría para el clasificador entrenado con los 3 corpus. Como se esperaba, los "features" que son específicos de cada dominio pierden peso en el ranking global (como *"instrucciones"*), o directamente no aparecen (*"madrid", "ibiza", "portatil"*), y se aprecia claramente cómo los que son fuertes indicadores semánticas de orientación ganan posiciones.

[11]Por estas diferencias es que puede llegar a darse un caso como el de videojuegos, donde considerar todos los dominios puede tener resultados inferiores al promedio de la consideración de cada uno por separado o en subconjuntos.

Prod. Electrónicos	**Hoteles**	**Películas**
instrucciones	gran	gran
capacidad	buena	papel
musica	grande	entretenida
manejo	ciudad	tema
salida	madrid	excelente
funciones	gusto	mejor
pesa	buen	normal
buena	punto	muestra
excelente	amable	espectacular
precio	servicio	viaje

Cuadro 6.16: 10 "features" con mayor peso positivo para los dominios de entrenamiento

Prod. Electrónicos	**Hoteles**	**Películas**
meses	viejo	mala
mala	mala	peor
deja	estrellas	argumento
sale	peor	desear
portatil	ibiza	actor
peor	muebles	aburrida
servicio	acondicionado	dinero
nada	ruido	bodrio
dinero	aspecto	intenta
imposible	deja	previsible

Cuadro 6.17: 10 "features" con mayor peso negativo para los dominios de entrenamiento

Positivas	**Negativas**
excelente	mala
gran	peor
grande	deja
genial	desear
pega	pobre
instrucciones	pocas
buena	meses
espectacular	imposible
muestra	viejo
gusto	dinero

Cuadro 6.18: "Features" con mayor peso para cada categoría con 3 dominios de entrenamiento

6.3.2. Estabilización y Comparación con Naïve Bayes

Aquí veremos cómo se comporta esta estrategia si se agregan más dominios, buscando que en algún punto se llegue a estabilizar. Se agregaron 3 dominios más: libros (aprox. 700 opiniones), vehículos -autos, motos, etc- (aprox. 500 opiniones) y artículos deportivos (aprox. 200 opiniones). Se presentan también los resultados obtenidos con Naïve Bayes, a efectos comparativos. Los valores son el promedio de todas las combinaciones entre los distintos dominios: es decir, se tomó a uno para testeo y al resto para evaluación, proceso que se repitió para todos ellos y se promedió el resultado.

Clasif.	**1 dom.**	**2**	**3**	**4**	**5**	**6**
Naïve Bayes	67.02 %	70.34 %	71.47 %	71.92 %	72.06 %	72.07 %
MaxEnt	69.46 %	72.1 %	73.1 %	73.76 %	73.98 %	74.01 %

Cuadro 6.19: MaxEnt vs Naïve Bayes en multidominio, las columnas son la cantidad de dominios de entrenamiento

A partir de 5 dominios los resultados parecen estabilizarse. De todas maneras, es importante aclarar que estos son los valores promedio obtenidos al

evaluar las distintas combinaciones; para algunos dominios, la diferencia siguió siendo considerable (por ejemplo, en prod. electrónicos los scores fueron 76.82 % y 77.34 % con 5 y 6 dominios de entrenamiento).

Con respecto a los valores comparativos, Naïve Bayes nuevamente obtiene buenos resultados, pero MaxEnt también lo supera en el caso multidominio, de acuerdo a lo que se había supuesto a partir del análisis de los "features" de mayor peso de cada clase en uno y otro clasificador.

Basándose en los resultados obtenidos en esta sección, se puede concluir que con un número considerable de dominios diferentes, el conjunto de "features" resultante es lo suficientemente robusto como para obtener buenos resultados con nuevos dominios no entrenados, ya que como se pudo ver, a medida que se va entrenando el clasificador con nuevos, los"features" con mayor peso se van estabilizando hacia un conjunto universal.

6.4. Precisión, Recall y F-Measure

En esta sección se presentan los cuadros con los valores de precisión (para cada categoría, cantidad de documentos correctamente clasificados sobre el total de documentos etiquetados en esa categoría), recall (para cada categoría, total de documentos correctamente clasificados sobre el total de documentos clasificados como pertenecientes a esa categoría) y F-Measure (media armónica de los valores anteriores) para cada clasificador en cada dominio (hay leves diferencias en los valores de "accuracy" con respecto a la sección de evaluación individual de métodos, lo cual se debe simplemente a la selección aleatoria de los conjuntos de entrenamiento y evaluación en cada uno de los 10 folds de cross-validation).

Clasif.	**Accuracy**	**Precisión**	**Recall**	**F-Measure**
Naïve Bayes	82.95 %	82.86 %	82.9 %	82.88 %
MaxEnt	91.05 %	91.01 %	91.04 %	91.03 %
SVM	83.08 %	83.64 %	82.81 %	83.22 %
SWN	66.15 %	66.07 %	65.84 %	65.96 %
SO-PMI	73.46 %	73.40 %	73.44 %	73.42 %

Cuadro 6.20: Valores para el dominio de Videojuegos

Clasif.	**Accuracy**	**Precisión**	**Recall**	**F-Measure**
Naïve Bayes	81.57 %	81.61 %	81.65 %	81.63 %
MaxEnt	87.33 %	87.44 %	87.41 %	87.42 %
SVM	78 %	78.02 %	78.04 %	78.03 %
SWN	66.23 %	66.13 %	66.07 %	66.10 %
SO-PMI	64 %	63.97 %	63.94 %	63.95 %

Cuadro 6.21: Valores para el dominio de Películas

Clasif.	**"Accuracy"**	**Precisión**	**Recall**	**F-Measure**
Naïve Bayes	82 %	82.13 %	82.13 %	82.13 %
MaxEnt	91 %	91.02 %	90.94 %	90.98 %
SVM	79 %	78.81 %	80 %	79.4 %
SWN	61.53 %	61.6 %	61.63 %	61.62 %
SO-PMI	65.68 %	65.59 %	65.62 %	65.61 %

Cuadro 6.22: Valores para el dominio de Productos Electrónicos

Clasif.	**Accuracy**	**Precisión**	**Recall**	**F-Measure**
Naïve Bayes	90.23 %	90.29 %	90.22 %	90.25 %
MaxEnt	94.41 %	94.42 %	94.41 %	94.42 %
SVM	84.41 %	84.51 %	84.22 %	84.37 %
SWN	74.88 %	74.82 %	74.77 %	74.79 %
SO-PMI	83.72 %	83.72 %	83.71 %	83.72 %

Cuadro 6.23: Valores para el dominio de Hoteles

La similitud entre los valores se debe a que no hubo selección de rasgos en el caso de los clasificadores de Machine Learning, y en los casos de Orientación Semántica debido a que se clasificó teniendo en cuenta un umbral que balanceó los resultados de cada categoría.

De todas maneras, un análisis que sí resulta interesante es ver qué ocurre con los casos de malas anotaciones (es decir, cuando el creador de la opinión realiza una calificación inconsistente con su verdadera orientación semántica): en teoría, si se desea detectar, por ejemplo, las opiniones de un dominio que son positivas pero anotadas erróneamente como negativas, se puede utilizar un clasificador con un umbral alto de pertenencia a la clase positiva (lo cual aumenta la precisión pero obviamente reduce el recall). De esa manera, aquellas opiniones que siguen siendo clasificadas como positivas pero tienen anotación negativa *deberían* estar mal anotadas. Se hizo un análisis manual en los dominios de videojuegos y hoteles con los distintos clasificadores, aumentando el umbral hasta el punto previo a donde las malas anotaciones

empezasen a quedar fuera y en general siempre aparecían junto a ellas un conjunto más grande de opiniones que estaban correctamente anotadas, simplemente mal clasificadas y presentando las características ya explicadas en las secciones de **Casos de falla** de los clasificadores. Es decir, la precisión para detectar malas anotaciones de esta manera es baja, a pesar de tener -obviamente- alto recall.

A modo de ejemplo, estos son los resultados obtenidos para el clasificador de MaxEnt (sin stopwords) en el dominio de Hoteles, con un umbral de 0.99 en cada una de las categorías:

- **Confusion Matrix**

	Pos. (Clasif.)	Neg. (Clasif.)	Total
Pos. (Real)	119	12	131
Neg. (Real)	9	120	129

- **Resultados**

  De los 9 falsos positivos dentro de las opiniones clasificadas como positivas, 1 de ellas era la mal anotada como negativa (1 de 5 estrellas, que obtuvo un score de 0.9978 para la clase positiva), y lo mismo para las 2 mal anotadas en la clase negativa. El resto de las opiniones fueron simplemente mal clasificadas. En general, para otros dominios y clasificadores el comportamiento fue similar, y en ningún caso se pudo lograr identificar las malas anotaciones con precisión aceptable.

Capítulo 7

Trabajo futuro

En esta sección se hará un análisis de los aspectos a mejorar de los métodos utilizados en este trabajo, y se propondrán alternativas para profundizar la efectividad de la tarea.

7.1. Orientación Semántica

- **SentiWordNet**: El mayor problema de este método es que los valores de los adjetivos en muchos casos no reflejan el peso real, lo cual afecta el score final de cada documento. Una forma muy simple de mejorar esto es ajustar esos valores manualmente. El otro aspecto a mejorar es la detección de negación de los adjetivos: la implementación propuesta lo hace dentro de una ventana predeterminada, una alternativa más apropiada sería obtener un árbol de dependencia sintáctica que permita determinar si existe relación entre las negaciones de una oración y los adjetivos que posee - sin embargo, esta alternativa no es posible hoy en día debido a la limitada precisión de los generadores de árboles de dependencia disponibles para el español.
- **SO-PMI**: Una alternativa para este método es acotar el "corpus" utilizado para calcular #hits, ya que como se vio en el análisis del método, la cantidad de información disponible en la web hace que ciertos resultados no sean representativos. Una forma simple de realizar esto es

obtener un corpus lo suficientemente grande de opiniones, y realizar las búsquedas de cercanía dentro del mismo. De todas maneras, para poder obtener valores realmente relevantes, como se mencionó previamente el corpus debería tener un tamaño considerable, con la demanda de tiempo y sanitizado de datos que esto conlleva.

7.2. Machine Learning

Para este tipo de métodos, más allá de ampliar el tamaño de los corpus, una alternativa que podría brindar resultados interesantes en caso de conocer los dominios a priori y ya tenerlos entrenados, sería realizar una etapa previa de clasificación (que hoy en día si se conocen las categorías es una tarea de alta "accuracy"), de manera tal que para cada documento a clasificar se obtenga el dominio al que pertenece, y con esa información utilizar un clasificador de opinión entrenado específicamente para ese dominio, aprovechando así la alta "accuracy" que tendrá al estar todo dentro del mismo contexto.

Además, realizar selección de rasgos y aplicar técnicas como las aplicadas para el método que utiliza a SentiWordNet pueden ser efectivas para mejorar los valores de "accuracy" de los clasificadores.

Capítulo 8

Conclusiones

En este trabajo se estudiaron diversas técnicas de minado de opinión para textos en español. Dentro de este marco, se evaluaron técnicas en los dos campos más estudiados para atacar este problema en el idioma inglés.

Para el caso de orientación semántica, específicamente en SO-PMI, se obtuvieron resultados comparables a los trabajos realizados en inglés[**?**], resultados que, dependiendo del dominio, pueden tener mejores resultados que el enfoque de Machine Learning. Se propuso una nueva técnica que obtuvo resultados comparables, pero que con una corta etapa de supervisión tiene margen para mejorar.

Para el caso de Machine Learning se obtuvieron muy buenos resultados si el entrenamiento y evaluación era dentro de un mismo dominio, y se pudo ver como a medida que el corpus va cubriendo una mayor cantidad de dominios, la performance al clasificar documentos de un dominio no entrenado empieza a mejorar, debido a una refinación y universalización automática del conjunto de pesos para "features" de cada categoría semántica.

Lo más importante a destacar es que el análisis realizado nos permite concluir que las técnicas de orientación semántica pueden funcionar muy bien en determinados dominios, pero en otros pueden llegar a estar incluso

por debajo de los resultados de Machine Learning entrenado con un dominio distinto al de evaluación. De todas maneras, estos métodos tienen un buen margen de mejora si se supervisan los valores utilizados para calcular la orientación semántica, por lo que potencialmente pueden tener resultados comparablas a Machine Learning intra-dominio. Por otro lado, los métodos de Machine Learning -especialmente MaxEnt-, muestran un gran potencial incluso en relación a su mayor falencia - la dependencia de dominio; ya que a medida que el corpus se va ampliando los "features" específicos de cada dominio son "empujados" hacia posiciones inferiores en términos de peso en cada categoría, convergiendo a un conjunto en general universalmente apropiado.

Bibliografía

[BHDM07] Boiy, E ; Hens, P ; Deschacht, K ; Moens, Marie-Francine: Automatic Sentiment Analysis in On-line Text Concepts of Emotions in Written Text Concept of Emotions. In: *Proceedings ELPUB2007 Conference on Electronic Publishing*, 2007, S. 349–360

[BO69] Boucher, Jerry D. ; Osgood, Charles E.: The Pollyanna hypothesis. In: *Journal of Verbal Learning and Verbal Behaviour* 8 (1969), S. 1–8

[BS10] Baccianella, Andrea Esuli S. ; Sebastiani, Fabrizio: SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In: *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, 2010

[CTC05] Cheng-Tao Chu, Pei-Chin W. Ryohei Takahashi T. Ryohei Takahashi: Classifying the Sentiment of Movie Review Data. In: *CS 224N Final Project Report* (2005)

[CV95] Cortes, Corinna ; Vapnik, Vladimir: Support-vector networks. In: *Machine Learning* 20 (1995), S. 273–297

[DLP03] Dave, Kushal ; Lawrence, Steve ; Pennock, David M.: Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In: *Proceedings of the 12th international conference on World Wide Web*, ACM, 2003, S. 519–528

[FC08] F.L. Cruz, F. Enríquez-F. Javier O. J.A. Troyano T. J.A. Troyano: Clasificación de documentos basada en la opinión: Experimentos con un corpus de de críticas de cine en español. In: *Boletín de la Sociedad Española para el Procesamiento del Lenguaje Natural* (2008)

[HW00] Hatzivassiloglou, Vasileios ; Wiebe, Janyce M.: Effects of Adjective Orientation and Gradability on Sentence Subjectivity, 2000, S. 299–305

[KNM99] Kamal Nigam, John L. ; McCallum, Andrew: Using maximum entropy for text classification. In: *Proc. of the IJCAI-99 Workshop on Machine Learning for Information Filtering*, 1999, S. 61–67

[LS09] Liu, Jingjing ; Seneff, Stephanie: Review Sentiment Scoring via a Parse-and-Paraphrase Paradigm. In: *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*. Singapore : Association for Computational Linguistics, August 2009, S. 161–169

[MBF+90] Miller, George A. ; Beckwith, Richard ; Fellbaum, Christiane ; Gross, Derek ; Miller, Katherine: WordNet: An online lexical database. In: *International Journal of Lexicography* 3 (1990), S. 235–244

[PCR+10] Padró, Lluís ; Collado, Miquel ; Reese, Samuel ; Lloberes, Marina ; Castellón, Irene: FreeLing 2.1: Five Years of Open-Source Language Processing Tools. In: *Proceedings of 7th Language Resources and Evaluation Conference (LREC'10)*. La Valletta, Malta, May 2010

[PL04] Pang, Bo ; Lee, Lillian: A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts. In: *Proceedings of the ACL*, 2004, S. 271–278

[Pla98] Platt, John C.: Fast training of support vector machines using sequential minimal optimization. In: *Advances in Kernel Methods: Support Vector Learning*, MIT Press, 1998, S. 185–208

[PLV02] Pang, Bo ; Lee, Lillian ; Vaithyanathan, Shivakumar: Thumbs up? Sentiment Classification using Machine Learning Techniques. In: *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2002, S. 79–86

[PT09] Prabowo, Rudy ; Thelwall, Mike: Sentiment analysis: A combined approach. In: *Journal of Informetrics* 3 (2009), Nr. 2, S. 143–157

[SDPL97] Stephen Della Pietra, Vincent Della P. ; Lafferty, John: Inducing features of random fields. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1997, S. 380–393

[SN10] Shein, Khin Phyu P. ; Nyunt, Thi Thi S.: Sentiment Classification Based on Ontology and SVM Classifier. In: *Communication Software and Networks, International Conference on* 0 (2010), S. 169–172. ISBN 978–0–7695–3961–4

[TBT+11] Taboada, Maite ; Brooke, Julian ; Tofiloski, Milan ; Voll, Kimberly ; Stede, Manfred: Lexicon-Based Methods for Sentiment Analysis. In: *Computational Linguistics* (2011), April, S. 1–41

[TL03] Turney, Peter D. ; Littman, Michael L.: Measuring praise and criticism: Inference of semantic orientation from association. In: *ACM Transactions on Information Systems* 21 (2003), S. 315–346

[Tur02] Turney, Peter D.: Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In: *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, 2002, S. 417–424

[WBB+01] Wiebe, Janyce ; Bruce, Rebecca ; Bell, Matthew ; Martin, Melanie ; Wilson, Theresa: A Corpus Study of Evaluative and Speculative Language. In: *Proceedings of the 2nd ACL SIGdial Workshop on Discourse and Dialogue*, 2001, S. 186–195

[WBB+03] Wiebe, Janyce ; Breck, Eric ; Buckley, Chris ; Cardie, Claire ; Davis, Paul ; Fraser, Bruce ; Litman, Diane ; Pierce, David ; Riloff, Ellen ; Wilson, Theresa ; Day, David ; Maybury, Mark: Recognizing and organizing opinions expressed in the world press. In: *In Working Notes - New Directions in Question Answering (AAAI Spring Symposium Series*, 2003, S. 24–26

[Wie00] Wiebe, Janyce M.: Learning Subjective Adjectives from Corpora. In: *In AAAI*, 2000, S. 735–740