

Ensemble algorithm with syntactical tree features to improve the opinion analysis

Algoritmo de ensamble con introducción de la estructura morfosintáctica para la mejora del análisis de opinión

Rafael del-Hoyo-Alonso ITAINNOVA C/ María de Luna, nº 7. 50018 Zaragoza, Spain rdelhoyo@itainnova.es	María de la Vega Rodrigalvarez-Chamarro ITAINNOVA C/ María de Luna, nº 7. 50018 Zaragoza, Spain vrodrigalvarez@itainnova.es	Jorge Veja-Murguía ITAINNOVA C/ María de Luna, nº 7. 50018 Zaragoza, Spain jveamurguia@itainnova.es	Rosa María Montañes-Salas ITAINNOVA C/ María de Luna, nº 7. 50018 Zaragoza, Spain rmontanes@itainnova.es
--	---	---	--

Resumen: En este artículo se describe cómo la integración en modo ensamble de varias técnicas de análisis de opinión permite mejorar la precisión del análisis u otros problemas de PLN donde el tamaño disponible del corpus de entrenamiento es pequeño en comparación al espacio de hipótesis y por tanto se deben explorar diversas estrategias. Dentro de las estrategias a introducir, se presenta una nueva forma de introducir el análisis morfo-sintáctico con objeto de encontrar relaciones que los métodos tradicionales no son capaces de realizar.

Palabras clave: Ensamble, árbol de Sintaxis, análisis de opinión, MaxEnt

Abstract: This article describes how the assemble of several opinion analysis techniques can improve the accuracy or in other NLP problems where the available size of the training corpus is small compared to the space of hypotheses and therefore should be explored a range of strategies. One of the strategies is based in a new way to include morpho-syntactic features to find relationships that traditional methods are not able to perform.

Keywords: Boosting, Sentiment Analysis, syntactic parser, MaxEnt

1 Introducción

El análisis de opinión es uno de los campos del análisis lingüístico donde las técnicas de minería de textos y de datos existentes es un excelente campo de batalla. Dentro del certamen TASS¹ 2015 (Villena-Román et al., 2015), se posibilita, a los diferentes grupos de investigación, compartir experiencias y permitir analizar el éxito de las diferentes aproximaciones. Este trabajo se enmarca dentro del reto primero de análisis de sentimiento.

En el análisis del sentimiento, la mayoría de las aproximaciones emplean algoritmos de aprendizaje automático (*Machine Learning*) (Pang et al., 2002) para construir clasificadores en base a un corpus predefinido donde se ha anotado la polaridad de la opinión

manualmente. En esta dirección, la mayoría de los estudios (Feldman, 2013) buscan la manera mejor de obtener las características (*features*) del texto para introducir a un clasificador. Esta extracción de características de basan en diferentes transformaciones cómo pueden ser TF-IDF (del inglés Term frequency – Inverse document frequency), word2vector (Mikolov et al., 2013) o a través de la búsqueda de diccionarios donde se introduzca conocimiento previamente preestablecido por psicólogos o lingüistas y de de la búsqueda de expresiones polarizadas.

Encontrar un algoritmo de aprendizaje supervisado eficiente no es una tarea fácil, en especial cuando el problema es altamente complejo cómo en el caso del análisis del lenguaje natural. Cada algoritmo se caracteriza porque emplea una representación diferente de los datos. Encontrar una buena representación de estos algoritmos, que permita generalizar el problema a resolver, requiere de tiempo y de

¹ TASS (Taller de Análisis de Sentimientos en la SEPLN) website. <http://www.daedalus.es/TASS>.

múltiples ensayos previos. Cada método o aproximación al problema de análisis de opinión han demostrado que ciertas partes del espacio de datos es mejor modelarla por un método de clasificación en comparación a otros. El empleo de diferentes algoritmos o aproximaciones al problema puede proporcionar información complementaria importante sobre la representación de los datos, uno de los objetivos que queremos tratar con este artículo. Esto ha originado que utilizar una combinación o ensamble de clasificadores sea una buena alternativa para el análisis de un corpus lingüístico. Varios métodos para la creación de ensamble de clasificadores ya han sido propuestos, en análisis de opinión dentro del TASS 2013 (Martínez-Cámara et al., 2013).

Un ensamble de clasificadores se define cómo la utilización de un conjunto de clasificadores que combinan de alguna forma las decisiones individuales de cada uno de ellos para clasificar nuevas instancias (Dzeroski & Zenki, 2000). Esta técnica es común en el mundo del *Machine Learning*. Existen varias razones que justifican el ensamble de clasificadores. Algunas de éstas, son: i) los datos para el entrenamiento pueden no proveer suficiente información para elegir un único mejor clasificador debido a que el tamaño disponible en estos datos es pequeño en comparación al espacio de hipótesis, este es el caso más común que nos vamos a encontrar en el análisis del lenguaje natural y en especial en el caso del TASS; ii) la combinación redundante y complementaria de clasificadores mejora la robustez, exactitud y generalidad de toda la clasificación; iii) diferentes clasificadores utilizan diferentes técnicas y métodos de representación de los datos, lo que permite obtener resultados de clasificación con diferentes patrones de generalización; iv) los ensambles son frecuentemente mucho más exactos que los clasificadores individuales. Por el contrario, podemos encontrar que la solución global de diferentes clasificadores pueda enmascarar la existencia de un clasificador excelente. Existen diferentes técnicas de ensamble, *bagging*, *boosting*, *stacked generalization*, *random subspace method*. En la aproximación seguida hemos utilizado una técnica de *bagging*, entrenando cada algoritmo con una muestra del 80% del total, donde en vez de votar la mejor de las salidas se entrena un nuevo clasificador (*meta-learner*), al igual que en el modelo de *stacked generalization*.

Cada algoritmo utilizado intenta explotar una variante del análisis de opinión con objeto de igualar el algoritmo de aprendizaje y de esta forma evitar desviaciones en el algoritmo de aprendizaje en el experimento, por el clasificador utilizado. El clasificador utilizado en todos los casos ha sido el modelo de clasificación de máxima entropía (*MaxEnt*). *MaxEnt* pertenece a la familia de los clasificadores exponenciales o *log-lineales* que extraen un conjunto de características (*features*) de la entrada, las combinan de modo lineal y utiliza el resultado cómo un exponente. *MaxEnt* es uno de los algoritmos más usados actualmente para análisis de Opinión.

Un paso más allá de la identificación de nuevas características o la introducción de intensificadores o negaciones en el análisis sintaxis del lenguaje es la introducción del análisis morfológico y análisis del árbol sintáctico dentro del análisis del sentimiento. Uno de los modelos que más éxito ha tenido en el análisis y la captura de los efectos del árbol sintáctico es el modelo *Recursive Neural Tensor Network* (RNTN), (Socher et al., 2013) o modelos cómo *convolutional neural network* (Kalchbrenner et al., 2014), dentro de la familia de algoritmos denominados *deep learning*. RNTN captura el cambio en la sintaxis cuando se realiza un cambio en la opinión permitiendo alcanzar mayores tasas de éxito que los métodos tradicionales de n-gramas a cambio de necesitar un corpus de entrenamiento de gran tamaño y basado en frases cómo el *Stanford Sentiment Treebank* (Socher et al., 2013), no siendo por tanto extrapolable a la competición TASS. Inspirado en esa idea y con la introducción del ensamblado de algoritmos se ha paliado el efecto negativo del reducido tamaño del corpus de entrenamiento.

Finalmente diferentes pruebas y el prototipado de los algoritmos se ha realizado mediante la herramienta *Moriarty Analytics* del Instituto Tecnológico de Aragón.

En el siguiente artículo se describirá la arquitectura del algoritmo propuesto, explicando cada una de las aproximaciones utilizadas. A continuación, se realizará un resumen de los resultados obtenidos. Finalmente se describirán unas conclusiones y trabajo futuro.

2 Arquitectura del Algoritmo

La composición de algoritmos permite explorar aproximaciones distintas dentro del corpus de entrenamiento. Los algoritmos utilizados han sido 4 que intentan cubrir diferentes aspectos, cada uno de ellos se ha entrenado con un 80% del corpus distinto y un 20% se ha dejado para validar el resultado.

El primer algoritmo consiste en la utilización de una aproximación simple de *Machine Learning* tras una normalización básica del texto. El segundo algoritmo consiste en la utilización del algoritmo basado en diccionario Afectivo de Cynthia M. Whissell (Whissell et al., 1986). El tercer algoritmo consiste en la utilización de otro diccionario pero en este caso se utilizará un mayor número de dimensiones de la psicóloga Margaret M. Bradley (Bradley & Lang, 1999). Finalmente, como cuarto algoritmo, se introduce un último método que incorpora la información del árbol de sintaxis y la información morfológica.

Para ello todos los algoritmos poseen una etapa de normalización, esta normalización consiste en la lematización, la eliminación de *stopwords* (sin eliminar las negaciones), eliminación de URLs y menciones.

Con objetivo de evitar una desviación debida al clasificador se ha optado en todos los casos por utilizar el mismo y evitar las desviaciones motivadas por el tipo de clasificador. Este seleccionado ha sido el de máxima entropía (*MaxEnt*).

Finalmente la selección del resultado final fue realizada con un segundo clasificador que fue entrenado con el global de los datos. En este caso se seleccionó un algoritmo *Random Forest*, con objeto de poder interpretar los resultados obtenidos, evitar sobre ajuste al no ser un algoritmo paramétrico y por haber obtenido los mejores resultados con otros algoritmos en un 10 *cross validation*.

2.1 Aproximación básica de Machine Learning

El primer algoritmo consiste en la introducción de un modelo de extracción de *features* automático mediante la utiliza del algoritmo tf-idf, y un máximo de 3 n-gramas. Estas *features* son introducidas al modelo de máxima entropía.

2.2 Diccionario Afectivo DAL

Esta aproximación se basa en la utilización de un diccionario afectivo, *Dictionary of Affect in Language* construido por Cynthia M. Whissell.

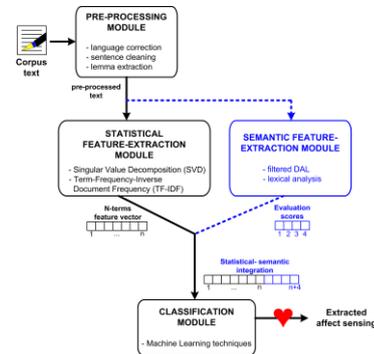


Figura 1 Arquitectura básica del algoritmo basado en DAL.

Una descripción del algoritmo utilizado puede verse en (Del-Hoyo et al., 2009).

2.3 Diccionario ANEW (Affective Norms for English Words)

Esta aproximación se basa en la utilización de un diccionario afectivo *Affective Norms for English Words* (ANEW; Bradley & Lang, 1999) y en este caso el diccionario afectivo en español, (Redondo et al., 2007). La utilización de este lexicón se ha utilizado por varios autores cómo (Gökçay et al., 2012; Gilbert, 2014)

El lexicón ANEW provee de un conjunto de 1.034 palabras etiquetadas emocionalmente. El diccionario de ANEW se ha etiquetado en tres dimensiones placer, activación y dominancia (pleasure, arousal, y dominance). La valencia va desde un valor de 1 hasta nueve siendo 5 el valor considerado como neutro.

Existen diferentes formas de introducirlo por diferentes autores desde el análisis de la suma de la valencia del texto o introducir un clasificador de los textos en función de las 3 dimensiones de cada texto. En nuestro caso, discretizaremos cada una de las dimensiones en 5 etiquetas, y se definirán al igual que en el caso anterior de un nuevo número de características con las que entrenaremos el algoritmo de clasificación.

2.4 Introduciendo la estructura topológica en la ecuación

Otra forma para poder evaluar y realizar un análisis de opinión dentro de un texto es estudiando su estructura morfo-sintáctica.

En primer lugar, el texto ha sido dividido en frases. Utilizando las librerías proporcionadas por *Freeling* (Padró & Stanilovsky, 2012), se ha analizado cada una de las frases morfológicamente, identificando cada una de las palabras que forman parte de ella y etiquetándolas. Para el idioma castellano, las etiquetas utilizadas son las propuestas por el grupo *EAGLES*². Por ejemplo, en la frase “Hay varios cruceros atracados en el puerto de Santa Cruz”, el análisis morfológico queda de la siguiente forma:

Palabra	Etiqueta
Hay	VMIP3S0
varios	DI0MPO
cruceros	NCMP000
atracados	VMP00PM
en	SPS00
el	DA0MS0
puerto	NCMS000
de	SPS00
Santa Cruz	NP00G00

Tabla 1: Análisis Morfológico.

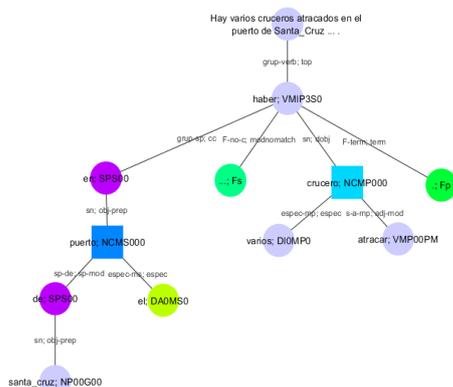


Figura 2: Árbol sintáctico obtenido.

Además, se genera el árbol sintáctico de cada una de las sentencias, generando un conjunto de grafos. El grafo resultante es el que se muestra en la figura 2.:

Una vez obtenido el grafo, donde se pueden ver de forma más sencilla las relaciones

existente entre las diferentes palabras de una frase, se obtienen todos los caminos (o *paths*) existentes con cierto grado de profundidad. Siendo cada *path* una nueva característica a introducir al algoritmos *MaxEnt*.

Para detectar todos los *paths* existentes en una frase, se ha generado un algoritmo que detecta todos los nodos terminales del grafo, es decir, que no tienen ningún hijo y obtiene el camino existente entre el nodo inicial y el nodo final. Un *path* viene descrito por la propiedad *chunk* y *parser* que define la arista (unión entre dos vértices) y el análisis morfológico simplificado de la palabra (vértice)

Por ejemplo, para la rama: “haber;en;puerto;de;Santa_Cruz”, el *path* resultante sería:

grup-verb;top/VM/grup-sp;cc/S/sn;obj-prep/NC00/sp-de;sp-mod/S/sn;obj-prep/NPG0

Todos los posibles *paths* obtenidos para esa rama con una longitud mínima de *x* y una longitud máxima de *y*, siendo el nivel de expansión en las pruebas realizadas de *x=3* e *y=5*, serían las mostradas en la tabla siguiente.

Paths Posibles Rama
grup-verb;top/VM/grup-sp;cc
grup-verb;top/VM/grup-sp;cc/S
grup-verb;top/VM/grup-sp;cc/S/sn;obj-prep
VM/grup-sp;cc/S
.....
sp-de;sp-mod/S/sn;obj-prep
sp-de;sp-mod/S/sn;obj-prep/NPG0
S/sn;obj-prep/NPG0

Tabla 2: Ejemplo de cómo se mapea un grafo en un conjunto de términos (posibles paths) del modelo.

Este proceso se seguiría para todas las ramas y se obtendrían todos los *paths* existentes para una frase. Por tanto todos los *paths* se incorporan cómo nuevas *features* para que, de esta forma, el algoritmo *MaxEnt* pueda encontrar las relaciones internamente no solamente de los pesos de las palabras por su semántica sino que pueda introducir la relación sintáctica y morfológica de las oraciones. Esta transformación, a diferencia de algoritmos cómo el RNTN que se desea mantener la estructura topológica de la entrada a través de la utilización de tensores, pretende mapear

² <http://www.ilc.cnr.it/EAGLES96/home.html>

espacios de dimensiones variables en un espacio de dimensión menor.

3 Resultados

A continuación en la tabla 3 se detallan la precisión en los datos test y desviación estándar esperada de cada uno de los algoritmos de forma independiente mediante la utilización de una evaluación *10 cross validation*.

	Basic	ANEW	DAL	Syntactic
Accu.	44.8%	43.8%	46.9%	41.3%
Desv.	1.1%	1.7%	1.6%	1.4%

Tabla 3: Resultados estimados en test obtenidos por cada una de los algoritmos mediante un 10 cross validation.

Se puede observar cómo la aproximación DAL obtiene unos mejores resultados en cambio, el algoritmo de análisis sintáctico obtiene peores resultados. Esto es debido a la explosión de nuevas características que genera este sistema, y para explotar mejor estas características, se necesita de un corpus mayor, cómo puede ser el *Sentiment Treebank*.

5 levels, Full test corpus				
	Accu.	P	R	F1
Ensem.	0.535	0.421	0.445	0.433
DAL.	0.524	0.419	0.397	0.408
5 levels, 1k corpus				
	Accu.	P	R	F1
Ensem.	0.405	0.401	0.392	0.396
DAL.	0.384	0.400	0.373	0.386
3 levels, Full test corpus				
	Accu.	P	R	F1
Ensem.	0.610	0.498	0.497	0.497
DAL.	0.594	0.484	0.482	0.483

Tabla 4: Resultados obtenidos en la prueba con el modelo ensembled y el mejor de los 4 modelos por separado, en cada una de las evaluaciones.

La composición de algoritmos se ha realizado a través de un modelo de Random Forest obteniendo una precisión de clasificación con los datos totales de entrenamiento de un 93%. Esto es así ya que se han utilizado un 80% de cada uno de ellos para el entrenamiento y todos ellos han obtenido alrededor de un 90% de precisión con los datos de entrenamiento,

por tanto era de esperar encontrar esos datos de precisión. Pero sí que se ha analizado si la inclusión de los datos sintácticos mejora o no el sistema, y se ha visto que la precisión del algoritmo mejoraba en un 2% con respecto a no introducir la entrada correspondiente al algoritmo sintáctico, y esta mejora era mayor que la desviación encontrada en los datos obtenidos por las n iteraciones realizada por el *cross-validation*.

Para validar globalmente el sistema correctamente utilizamos la validación de TASS, para ello vamos a comparar el mejor de los algoritmos por separado de forma independiente al algoritmo compuesto. Se puede ver en la tabla 4 cómo en todos los casos el modelo compuesto gana al modelo DAL de forma independiente. Analizando los resultados por clases el sistema puede verse que el sistema compuesto se comporta de la misma forma que los sistemas individuales, obteniendo mejores resultados en las etiquetas Positivas que las Negativas, y fallando más en diferenciar la etiqueta NEU de NONE. Utilizando datos de evaluación se puede analizar cómo en determinados casos el modelo Sintáctico encuentra relaciones que no son encontradas por los otros modelos. Vamos a poner varios ejemplos a continuación.

“Y además no llueve! <http://t.co/hXo9c8p9>”

“1.Primer acto. Mario Monti entra en escena cambiando el panorama. Merkel con Berlusconi no hubiera confiado en las promesas de IT”

En ambos casos se obtiene el siguiente resultado:

	Basic	ANEW	DAL	Syntactic
Predic.	None	None	None	N

Tabla 5: Resultados de opinión negativa dónde la información morfo-sintáctica ofrece un valor añadido.

Puede verse que en frases donde existe una negación pero no existe ninguna palabra “afectiva” puede verse que subjetivamente es negativa. En cambio los otros algoritmos al no encontrar ningún vocabulario específico nos indican un valor nulo. Por el contrario, en varios casos el “Si” en oraciones condicionales enfatiza y modifica las predicciones.

“Jejeje. Si lo escuche. RT @cris_rc: ”@SilviaSanz7: Jajajajaja Suena @AlejandroSanz en el (cont) <http://t.co/OVHwMetV>”

	Basic	ANEW	DAL	Sintactic
Predic.	P	P	P	P+

Tabla 6: Resultados de opinión dónde la información morfo-sintáctica ofrece un valor añadido.

4 Conclusiones

Se ha presentado la composición de algoritmos cómo herramienta para la mejora de la precisión del análisis de sentimiento u otros problemas de NLP donde el tamaño disponible del corpus de entrenamiento es pequeño en comparación al espacio de hipótesis y por tanto se deben explorar diversas estrategias. Por otra parte, se presentado una nueva forma de introducir el análisis morfo-sintáctico con objeto de encontrar nuevas relaciones.

Bibliografía

Bradley, M. M., & P. J. Lang, (1999). Affective norms for English words (ANEW): Instruction manual and affective ratings (pp. 1-45). Technical Report C-1, The Center for Research in Psychophysiology, University of Florida.

Del-Hoyo, R., I. Hupont, F. J. Lacueva, & D. Abadía, (2009, November). Hybrid text affect sensing system for emotional language analysis. In Proceedings of the international workshop on affective-aware virtual agents and social robots (p. 3). ACM.

Dzeroski, S. & B. Zenki (2000). Is Combining, Classifiers Better than Selecting the Best One. International Conference on Machine Learning (ICML): 123-130.

Feldman, R. (2013). Techniques and applications for sentiment analysis. Communications of the ACM, 56(4), 82-89.

Gilbert, C. H. E. (2014). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text.

Gökçay, D., E. İşbilir, & G. Yıldırım, (2012, December). Predicting the sentiment in sentences based on words: An exploratory study on ANEW and ANET. In Cognitive Infocommunications, 2012 IEEE 3rd International Conference on (pp. 715-718)..

Kalchbrenner, N., E. Grefenstette, & P. Blunsom, (2014). A convolutional neural network for modelling sentences. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. June, 2014

Martinez-Cámara, E., Y. Gutiérrez-Vázquez, J. Fernández, A. Montejo-Ráez & R. Muñoz-Guillena. (2013) Ensemble classifier for Twitter Sentiment Analysis. TASS 2013

Mikolov, T., K. Chen, G. Corrado, & J. Dean, (2013). Efficient estimation of word representations in vector space. in International Conference on Learning Representations

Padró, L., & E. Stanilovsky, (2012). Freeling 3.0: Towards wider multilinguality. Proceedings of the Language Resources and Evaluation Conference (LREC) ELRA. Istanbul, Turkey.

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

Redondo, J., I. Fraga, I. Padrón, & M. Comesaña, (2007). The Spanish adaptation of ANEW (affective norms for English words). Behavior research methods, 39(3), 600-605.

Socher, R., A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, & C. Potts, (2013, October). Recursive deep models for semantic compositionality over a sentiment treebank. In Proceedings of on empirical methods in natural language processing (EMNLP) (Vol. 1631, p. 1642).

Villena-Román, J., J. García-Morera, M. A.; García-Cumbreras, E. Martínez-Cámara, M. T. Martín-Valdivia, L. A. Ureña-López, (2015): Overview of TASS 2015, In Proceedings of TASS 2015: Workshop on Sentiment Analysis at SEPLN. vol. 1397.

Whissell, C., M. Fournier, R. Pelland, D. Weir, & K. A. Makarec, (1986) dictionary of affect in language. IV. Reliability, validity, and applications. Perceptual and Motor Skills, 62(3), 875-888.