Aplicación del algoritmo Boosting Adaptativo (ADABOOST) a un problema de clasificación automática de textos

Application of the Adaptive Boosting Algorithm (ADABOOST) to a problem of automatic text classification

Ivana Barbona; Celina Beltrán
Facultad de Ciencias Agrarias, Universidad Nacional de Rosario, Argentina

ivanabarbona@gmail.com

Abstract

Boosting is a method that aims to improve the performance of any supervised learning algorithm by combining the results of several weak or base classifiers to obtain a robust final classifier.
One of the most popular Boosting techniques is the Adaptive Boosting algorithm (AdaBoost). This algorithm, through an iterative training of the weak or base classifiers, gives greater importance to the previously misclassified data, and in this way obtains a new classifier, Achieving better results increasing the accuracy of the algorithm.
In the present article, with the objective of evaluating the performance of the AdaBoost algorithm, Logistic Regression and SMO (Sequential minimal optimization) classification methods are applied, with and without the AdaBoost algorithm to a set of texts. Then, the results obtained from the classification methods are compared with the results considered them as the base algorithm for AdaBoost. The classification criteria used was the gender to which the text belongs (Scientific / Non-Scientific). The characterization of the texts is based on the frequency distribution of the morpho-syntactic categories. The final results of the different classifiers considered are evaluated by percentages of bad classification. It was observed that when applying AdaBoost considering the Logistic Regression method as a basic algorithm, there was no reduction in the percentage of misclassification. In contrast, in the case of the SMO method as a base algorithm, the percentage of bad classification fell by 8.67%.
.
Keywords: Support Vector Machine, Learning Machine, Supervised Classification Methods, Text Classification.

Resumen

Boosting es un método que pretende mejorar el desempeño de cualquier algoritmo de aprendizaje supervisado mediante la combinación de los resultados de varios clasificadores débiles o de base para obtener un clasificador final robusto.
Una de las técnicas más populares de Boosting es el algoritmo Boosting Adaptativo (AdaBoost). Este algoritmo, mediante un entrenamiento iterativo de los clasificadores débiles o de base, le asigna mayor importancia a los datos mal clasificados anteriormente, y de esta manera obtiene un nuevo clasificador. Logra, de esta forma, adaptarse y obtener mejores resultados aumentando la precisión del algoritmo.
En el presente trabajo, con el objetivo de evaluar el desempeño del algoritmo AdaBoost, se aplican los métodos de clasificación Regresión Logística y SMO (Sequential minimal optimization), con y sin el algoritmo AdaBoost a un conjunto de textos. Luego, se comparan los resultados obtenidos de

los métodos de clasificación al considerarse solos, con los resultados al considerarlos como algoritmo de base para AdaBoost. El criterio de clasificación utilizado fue el género al que pertenece el texto (Científico / No Científico). La caracterización de los textos está basada en la distribución de frecuencias de las categorías morfo-sintácticas. Los resultados finales de los distintos clasificadores considerados se evalúan mediante porcentajes de mala clasificación. Se observó que al aplicar AdaBoost teniendo en cuenta como algoritmo de base el método de Regresión Logística no se presentó una reducción en el porcentaje de mala clasificación. En cambio, para el caso del método SMO como algoritmo de base, el porcentaje de mala clasificación bajó un 8,67%.

Palabras clave: Support Vector Machine, Learning Machine, Métodos de Clasificación Supervisada, Clasificación de Textos.

1. INTRODUCCION

Boosting es un algoritmo que se utiliza para mejorar el desempeño de cualquier método de aprendizaje supervisado mediante la combinación de los resultados de varios clasificadores débiles o de base para obtener un clasificador final robusto.

Una de las técnicas más populares de Boosting es el algoritmo Boosting Adaptativo (AdaBoost). Este algoritmo, mediante un entrenamiento iterativo de los clasificadores débiles o de base, le asigna mayor importancia a los datos mal clasificados anteriormente, y de esta manera obtiene un nuevo clasificador, logrando adaptarse y obtener mejores resultados aumentando la precisión del algoritmo.

El presente trabajo tiene como objetivo evaluar el desempeño del algoritmo AdaBoost. Se aplican los métodos de clasificación Regresión Logística y SMO (Sequential minimal optimization), con y sin el algoritmo AdaBoost a un conjunto de textos para luego comparar los resultados obtenidos de los métodos de clasificación al considerarse solos, con los resultados al considerarlos como algoritmo de base para AdaBoost. El criterio de clasificación utilizado fue el género al que pertenece el texto (Científico / No Científico). La caracterización de los textos está basada en la distribución de frecuencias de las categorías morfo-sintácticas.

Los resultados finales de los distintos clasificadores considerados se evalúan mediante porcentajes de mala clasificación y el Índice de Concordancia Kappa, medida de concordancia entre las categorías pronosticadas por el clasificador y las categorías observadas.

2. MATERIAL Y METODOS

2.1. Diseño de la muestra

El marco muestral para la selección de la muestra de los textos científicos está compuesto por textos académicos, resúmenes de trabajos presentados a congresos y revistas, extraídos de internet pertenecientes a distintas disciplinas. Los textos periodísticos fueron seleccionados de un corpus mayor utilizado por el equipo de investigación INFOSUR. Este corpus se construyó con noticias extraídas de las páginas web de periódicos argentinos (noticias de tipo general, no especializadas en español). La unidad de muestreo fue el texto y la selección de la muestra se llevó a cabo empleando un diseño muestral estratificado.

Luego de obtener las muestras de los estratos, fueron evaluadas y comparadas respecto al número medio de palabras por texto. Se requiere esta evaluación para evitar que la comparación entre los géneros se vea afectada por el tamaño de los textos.

La muestra final para este trabajo quedó conformada de la siguiente manera:

Tabla 1. Conformación de la muestra final

Muestra	Nro. de textos	Cantidad de palabras
Científico	90	14.554
No científico	60	8.080

2.2. Etiquetado: Análisis morfológico de los textos

El software Smorph, analizador y generador morfosintáctico desarrollado en el Groupe de Recherche dans les Industries de la Langue (Universidad Blaise-.Pascal, Clermont II) por Salah Aït-Mokhtar (1998) realiza en una sola etapa la tokenización y el análisis morfológico. A partir de un texto de entrada se obtiene un texto lematizado con las formas correspondientes a cada lema (o a un subconjunto de lemas) con los valores correspondientes. Se trata de una herramienta declarativa, la información que utiliza está separada de la maquinaria algorítmica, en consecuencia, puede adaptarse a distintos usos. Con el mismo software se puede tratar cualquier lengua si se modifica la información lingüística declarada en sus archivos.

El módulo post-smorph **MPS** es un analizador que recibe en entrada una salida Smorph (en formato Prolog) y puede modificar las estructuras de datos recibidos. Ejecuta dos funciones principales: la Recomposición y la Correspondencia, que serán útiles para resolver las ambigüedades que resulten del análisis de Smorph.

La información contenida en estos archivos es la presentada en Beltrán (2009) para implementar el etiquetador.

2.3. Diseño y desarrollo de la base de datos

La información resultante del análisis morfológico se dispuso en una matriz de dimensión: tantas filas como cantidad de objetos lingüísticos tenga el texto y tantas columnas como ocurrencia+lema+valores. Luego, a partir de esta base de datos por palabra, se confeccionó la base de datos por documento que es analizada estadísticamente. La información registrada en esta base corresponde a las siguientes variables:

- CORPUS: Corpus al que pertenece el texto
- TEXTO: Identificador del texto dentro del corpus
- Adj: cantidad de adjetivos del texto
- Adv: cantidad de adverbios del texto
- Cl: cantidad de clíticos del texto
- Cop: cantidad de copulativos del texto
- Det: cantidad de determinantes del texto
- Nom: cantidad de nombres (sustantivos) del texto
- Prep: cantidad de preposiciones del texto
- V: cantidad de verbos del texto
- Otro: cantidad de otras etiquetas del texto

- Total_pal: cantidad total de palabras del texto

2.4. Metodología estadística

Las técnicas evaluadas en este trabajo tienen por objetivo construir un sistema que permita clasificar unidades en una de las categorías definidas y conocidas previamente en función de las variables relevadas, como así también otras variables que demuestren un aporte significativo en la predicción del grupo de pertenencia. Luego de la aplicación de cada una de las técnicas definidas en este apartado se debe evaluar la calidad de los resultados, es decir el desempeño para clasificar mediante la validación del mismo. Esto se realiza particionando el conjunto de unidades en dos grupos. Uno es utilizado para la estimación del mismo (entrenamiento del sistema) y el segundo conforma el grupo de validación para la fase de prueba.

Luego, a cada técnica se le aplica el algoritmo Boosting Adaptativo (AdaBoost) para evaluar la mejora en la clasificación de dichas técnicas. Se consideró como medida para la comparación entre métodos el error de mala clasificación calculada sobre una muestra de textos no incluidos en el proceso de construcción de la regla de clasificación. Otra medida que se utilizó fue el Índice de Concordancia Kappa, medida de concordancia entre las categorías pronosticadas por el clasificador y las categorías observadas, que tiene en cuenta las posibles concordancias debidas al azar. Donde:
- Si el valor es 1: Concordancia perfecta.

- Si el valor es 0: Concordancia debida al azar.

- Si el valor es negativo: Concordancia menor que la que cabría esperar por azar.

A continuación se presentan dos de las técnicas multivariadas que tienen por objetivo clasificar unidades en categorías definidas a priori que fueron evaluadas en diferentes aplicaciones por los autores y en este trabajo se las aplica con y sin el algoritmo AdaBoost.

2.4.1. Análisis de regresión logística

Esta técnica es un caso particular de los modelos lineales generalizados, modela la probabilidad de que una unidad experimental pertenezca a un grupo en particular considerando información medida o registrada en dicha unidad.
La regresión logística es utilizada en situaciones en las cuales el objetivo es describir la relación entre una variable respuesta categórica, en este caso dicotómica, y un conjunto de variables explicativas que pueden ser tanto categóricas como cuantitativas.
Sea **x** un vector de p variables independientes, esto es, $\mathbf{x}' = (x_1, x_2, \ldots, x_p)$. La probabilidad condicional de que la variable *y* tome el valor 1 (presencia de la característica estudiada), dado valores de las covariables **x** es:

$$P(y = 1/X) = \pi(x) = \frac{e^{g(x)}}{1 + e^{g(x)}}$$

donde $g(x) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p$

β_0 es la constante del modelo o término independiente
p el número de covariables
βi los coeficientes de las covariables
xi las covariables que forman parte del modelo.

Si alguna de las variables independientes es una variable discreta con k niveles se debe incluir en el modelo como un conjunto de k-1 "variables de diseño" o "variables dummy". El cociente de las probabilidades correspondientes a los dos niveles de la variable respuesta se denomina odds y su expresión es:

$$\frac{P(y=1|X)}{1-P(y=1|X)} = e^{\beta_0+\beta_1 x_1+\ldots+\beta_p x_p}$$

Si se aplica el logaritmo natural, se obtiene el modelo de regresión logística:

$$\log\left(\frac{P(y=1|\,X)}{1-P(y=1|\,X)}\right) = \log(e^{\beta_0+\beta_1 x_1+\ldots+\beta_p x_p}) = \beta_0+\beta_1 x_1+\ldots+\beta_p x_p$$

En la expresión anterior el primer término se denomina logit, esto es, el logaritmo de la razón de proporciones de los niveles de la variable respuesta.
Los coeficientes del modelo se estiman por el método de Máxima Verosimilitud y la significación estadística de cada uno de los coeficientes de regresión en el modelo se puede verificar utilizando, entre otros, el test de Wald y el test de razón de verosimilitudes.

Una cuestión importante en este tipo de análisis es determinar si todas las variables consideradas en la función de discriminante contienen información útil y si solamente algunas de ellas son suficientes para diferenciar los grupos. Dado que las variables utilizadas para explicar la respuesta es probable que estén correlacionadas, es posible también que compartan información. Por lo tanto, se puede buscar un subgrupo de variables mediante algún criterio de modo tal que las variables excluidas no contengan ninguna información adicional. Existen varios algoritmos de selección de variables, entre ellos podemos citar:
Método forward: comienza por seleccionar la variable más importante y continúa seleccionando las variables más importantes una por vez, usando un criterio bien definido. Uno de estos criterios involucra el logro de un nivel de significación deseado pre-establecido. El proceso termina cuando ninguna de las variables restantes encuentra el criterio pre-especificado.
Método backward: comienza con el modelo más grande posible. En cada paso descarta la variable menos importante, una por vez, usando un criterio similar a la selección forward. Continúa hasta que ninguna de las variables pueda ser descartada.
Selección paso a paso: combina los dos procedimientos anteriores. En un paso, una variable puede entrar o salir desde la lista de variables importantes, de acuerdo a algún criterio pre-establecido.
En este trabajo se utilizó como evaluación del ajuste del modelo el test de Hosmer-Lemeshow y, dado que el modelo es utilizado para clasificar unidades, se utilizó también la tasa de mala clasificación calculada sobre la muestra independiente excluida en la etapa de estimación.

2.4.2. Support Vector Machine (SVM) y Sequential Minimal Optimization (SMO)

Support Vector Machine utiliza un algoritmo que se basa en una clase especial de modelo lineal denominado ***hiperplano óptimo de máximo margen.*** Este hiperplano, que pertenece a un espacio de dimensionalidad que puede llegar a ser infinito, es hallado utilizando vectores soporte. Luego, mediante una transformación inversa se obtiene una frontera no necesariamente lineal que separa los grupos en el espacio original.
Los ***vectores soportes*** son las observaciones que están más cerca del hiperplano. Siempre hay como mínimo un vector soporte para cada clase.
La expresión del hiperplano de máximo margen viene dada por:

$$x = b + \sum a_i\, y_i\, (a(i).a)^n$$

dónde y_i es -1 o 1 depende del grupo al que pertenezca la observación; **a**(i) es el vector de valores de atributos correspondientes al i-ésimo vector soporte y **a** otro vector de atributos para una observación; b y α son parámetros calculados por el algoritmo; y *n* se elije según el grado del polinomio kernel con el que se desee trabajar. Algunos de los kernels existentes son Lineal (n=1), Polinomio de segundo grado (n=2), Radial Basis Function (RBF de parámetro γ). Al aplicar el método de SVM hay que tener en cuenta la constante de penalización C que impone una cota máxima al coeficiente α_i

Sequential Minimal Optimization (SMO) es un algoritmo que resuelve un problema, que surge en SVM, de optimización de una función cuadrática de varias variables, pero sujetas a una restricción lineal de esas variables.

2.4.3. Algoritmo Boosting Adaptativo

Adaboost fue el primer algoritmo de boosting *adaptativo* en los métodos de aprendizaje de máquina. Mediante un entrenamiento iterativo de los clasificadores débiles o de base, le asigna mayor importancia a los datos mal clasificados anteriormente, y de esta manera obtiene un nuevo clasificador. Logra, de esta forma, adaptarse y obtener mejores resultados aumentando la precisión del algoritmo.

El funcionamiento del algoritmo se puede presentar de la siguiente manera (Mayr, 2014):

Inicio

(1) Se fija el contador de iteración en $m = 0$ y los pesos individuales w_i para las observaciones con $i = 1, \dots, n$ siendo $w_i^{[0]} = \frac{1}{n}$.

Algoritmo de Base

(2) Hacer $m := m + 1$ y calcular el clasificador de base para los datos ponderados:

Observaciones ponderadas con $w_1^{[m-1]}, \dots, w_n^{[m-1]} \xrightarrow{clasificador\ de\ base} \hat{h}^{[m]}(.)$

Actualizar ponderaciones

(3) Calcular la tasa de error y actualizar el coeficiente específico de iteración $\alpha_m \rightarrow$ valores altos para tasas de error bajas. Actualizar los pesos individuales $w_1^{[m]} \rightarrow$ valores altos si la observación fue mal clasificada.

Iterar

(4) Iterar pasos 2 y 3 hasta que $m = m_{stop}$

3. RESULTADOS

Sequential Minimal Optimization (SMO) vs AdaBoost con SMO como algoritmo de base.

SMO	
Instancias Clasificadas Correctamente	**65,33%**

Instancias Clasificadas Incorrectamente o %MC	**34,67%**
Estadística Kappa	**0,2073**
Número total de instancias	**150**

Para el método de clasificación SMO, el %MC fue del 34,67% y el coeficiente de concordancia Kappa 0,2073 lo cual indicaría un bajo grado de concordancia.

AdaBoost con SMO como base	
Instancias Clasificadas Correctamente	**72,67%**
Instancias Clasificadas Incorrectamente o %MC	**27,33%**
Estadística Kappa	**0,416**
Número total de instancias	**150**

Al aplicar AdaBoost con SMO como clasificador de base, el %MC disminuye, siendo del 27,33% y el coeficiente de concordancia Kappa 0,416.

Regresión Logística (RL) vs AdaBoost con RL como base

Regresión Logística	
Instancias Clasificadas Correctamente	**77,33%**
Instancias Clasificadas Incorrectamente o %MC	**22,67%**
Estadística Kappa	**0,5278**
Número total de instancias	**150**

AdaBoost con Regresión Logística como base	
Instancias Clasificadas Correctamente	**77,33%**
Instancias Clasificadas Incorrectamente o %MC	**22,67%**
Estadística Kappa	**0,5278**

Número total de instancias	**150**

Tanto para el método de Regresión Logística como para el caso de AdaBoost con Regresión Logística como base, el %MC fue 22,67% y el coeficiente de concordancia Kappa fue 0,5278.

4. Conclusiones

Al comparar el método de clasificación SMO con AdaBoos con SMO como base se observó que el %MC disminuyó 7,34%, y el índice de concordancia Kappa aumentó. Por lo tanto, el Boosting Adaptativo mejoró el desempeño del método SMO.
En el caso de Regresión Logística, el desempeño resultó mejor que para SMO, no obstante la aplicación del Boosting Adaptativo no mejoró el rendimiento.

Referencias

Andreas Mayr, Harald Binder, Olaf Gefeller, Matthias Schmid. 2014 The Evolution of Boosting Algorithms. From Machine Learning to Statistical Modelling. Methods Inf Med 2014; 53(6): 419-427. arXiv:1403.1452 [stat.ME]

Barbona, I. 2015 Comparación de métodos de clasificación aplicados a textos Científicos y No Científicos. Revista INFOSUR. Grupo INFOSUR. Rosario.

Beltrán, C. 2009 Modelización lingüística y análisis estadístico en el análisis automático de textos. Ediciones Juglaría. Rosario.

Bès,Gabriel, Solana, Z y Beltrán, C. 2005 Conocimiento de la lengua y técnicas estadísticas en el análisis lingüístico en Desarrollo, implementación y uso de modelos para el procesamiento automático de textos (ed. Víctor Castel) Facultad de Filosofía y Letras, UNCUYO

Cuadras, C.M. 2014 NUEVOS MÉTODOS DE ANÁLISIS MULTIVARIANTE. CMC Editions. Barcelona, España.

Freund, Y., Shapire, R., Experiments with a New Boosting Algorithm, International Conference on Machine Learning, 1996.

Hosmer, D., Lemeshow, S., Sturdivant, R.2013.Applied Logistic Regression.John Wiley & Sons.

Witten, I., Frank, E. 2005. Data Mining. Practical Machine Learning Tools and Techniques. Elsevier.