

La Combinación de Sistemas y el PLN *

Fernando Enríquez, José A. Troyano, Fermín Cruz y F. Javier Ortega

Dep. de Lenguajes y Sistemas Informáticos

Universidad de Sevilla

Avda. Reina Mercedes s/n

41012 Sevilla

fenros@us.es

Resumen: La combinación de sistemas constituye un área de investigación ampliamente estudiada en el ámbito del Reconocimiento de Patrones, en donde se han desarrollado múltiples técnicas para aprovechar la diversidad de métodos de clasificación de los que se dispone actualmente gracias al Aprendizaje Automático. Tras un análisis de los trabajos que emplean alguna de estas técnicas para resolver tareas del PLN, vemos que aunque se le haya prestado atención en múltiples trabajos a esta línea de investigación, se suelen emplear los métodos más populares debido quizás al desconocimiento general sobre la existencia de un gran número de opciones disponibles para llevar a cabo la combinación. En este trabajo pretendemos dar a conocer los aspectos teóricos en los que se apoya el uso de estas técnicas, así como mostrar el amplio abanico de opciones existentes hoy en día, además de repasar los trabajos más importantes de PLN que han hecho uso de alguna de ellas.

Palabras clave: Aprendizaje Automático, combinación de sistemas

Abstract: The combination of systems is an area of widely studied research in the field of Recognition Patterns, where there have been developed many techniques for taking advantage of the diversity of classification methods that are currently available thanks to Machine Learning. After an analysis of the works that employ some of these techniques to solve NLP tasks, we see that although this research has been considered in many cases, only the most popular methods are often used perhaps due to widespread ignorance about the existence of a large number of options available for carrying out the combination. In this paper we intend to present the theoretical aspects that support the use of these techniques, to show the wide range of options existing today, and review the most important NLP works where any of them has been used.

Keywords: Machine Learning, system combination

1. Introducción

Durante muchos años han ido surgiendo un número considerable de métodos de clasificación, y aunque en algunos casos podamos reconocer uno de ellos como el mejor dotado para la tarea que pretendemos afrontar, no es lo habitual disponer de un clasificador perfecto que alcance una precisión del 100%. En otros muchos casos no podemos ni tan siquiera disponer de una lista ordenada que nos facilite el nombre del mejor sistema de clasificación. En el primer caso, donde te-

nemos un candidato destacado del resto, no podemos asegurar que éste no carezca de alguna característica deseable que sí se encuentra presente en el resto de clasificadores. En el segundo caso expuesto, parece lógico pensar que sería mejor poder trabajar con todos los clasificadores, combinando sus resultados para extraer lo mejor de cada uno, en lugar de decantarnos por uno de ellos, descartando al resto. Un gran número de tareas del Procesamiento del Lenguaje Natural (PLN) pueden considerarse tareas de clasificación, donde debemos resolver algún tipo de ambigüedad relacionada con el contenido textual de una colección de documentos, pudiendo

* Parcialmente financiado por el proyecto I+D HUM2007-66607-C04-04.

do aplicarse múltiples algoritmos de clasificación, así como técnicas y estrategias de combinación de sistemas. En la siguiente sección comentaremos las razones que justifican la combinación y los diferentes puntos de vista desde los que podemos aplicarla. A continuación comentaremos las familias más influyentes de técnicas y algoritmos. En la sección 4 abordaremos la combinación de sistemas desde el punto de vista de su aplicación al PLN para pasar finalmente a exponer las conclusiones.

2. La Combinación de Sistemas

En Hansen y Salamon (Hansen y Salamon, 1990) se establecen la *precisión* y la *diversidad* como requisitos necesarios y suficientes para llevar a cabo con éxito la combinación de dos o más sistemas de clasificación. Por su parte Dietterich (Dietterich, 2000) justifica la combinación desde tres puntos de vista. Considerando el punto de vista *estadístico*, escoger un clasificador para resolver nuestro problema entraña un riesgo, ya que éste puede no ser el más capacitado para el caso particular que estamos tratando. Debido a esto, puede que la combinación no nos proporcione mejores resultados que el mejor clasificador individual de que disponemos, pero sí que elimina o mitiga el riesgo de equivocarnos en la selección. Desde el punto de vista *computacional*, aún teniendo un volumen de datos tal que haga desaparecer el problema estadístico mencionado anteriormente, existe el problema de los máximos locales, en los que pueden incurrir muchos algoritmos en su búsqueda del objetivo. Una vez más, la combinación de los clasificadores puede realizar la búsqueda partiendo de muchos puntos diferentes del espacio de hipótesis y acercarse más a la solución ideal de lo que consiguen acercarse estos clasificadores individualmente. Por último, desde el punto de vista de la *representación* hay que considerar la posibilidad de que el espacio de búsqueda de los clasificadores no contenga la solución que estamos buscando, mientras que la combinación puede darnos la oportunidad de expandir dicho espacio con la posibilidad que esto acarrea de que consigamos incluir el objetivo en dicho espacio y tengamos opciones de aproximarnos más a él.

Apoyándose en estas afirmaciones se han desarrollado múltiples métodos de combinación, muchos de ellos aportando enfoques

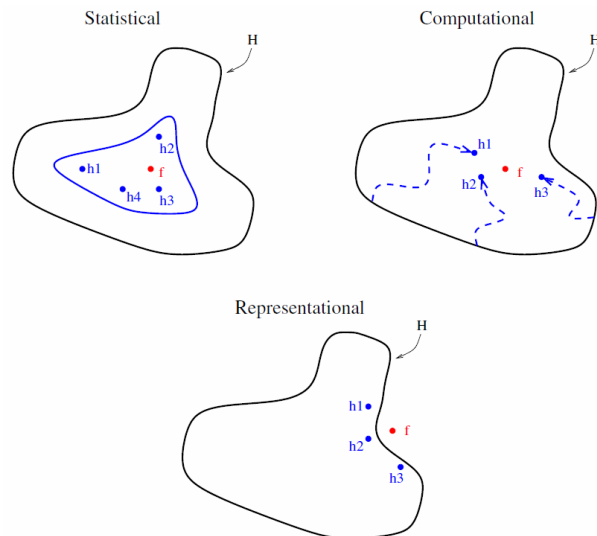


Figura 1: Justificación para la combinación según Dietterich.

muy dispares que en (Kuncheva y Whitaker, 2003) son organizados en base a cuatro niveles como se aprecia en la figura 2.

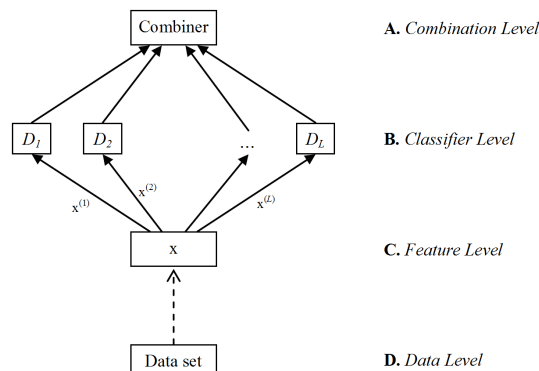


Figura 2: Enfoques para la combinación de clasificadores.

Estos niveles representan el punto del proceso donde recae el peso de la combinación, pudiendo hacer uso de diferentes colecciones de datos (*data level*), diferentes subconjuntos de características empleadas para representar los ejemplos (*feature level*), diferentes clasificadores (*classifier level*) o distintas técnicas de combinación (*combiner level*). Prácticamente la totalidad de métodos existentes puede categorizarse en función del nivel en el que se ejerce mayor énfasis.

Existen otros criterios para agrupar los métodos de combinación, como puede ser la distinción entre métodos de fusión y métodos de selección. En la fusión todas las opiniones gozan de la misma consideración y se busca

una opinión común, mientras que en la selección se elige un participante responsable de clasificar un determinado grupo de ejemplos.

No todos los métodos de combinación existentes son aplicables a cualquier conjunto de clasificadores. Es importante considerar el tipo de información que estos producen como salida. En (Xu, Krzyzak, y Suen, 1992) se describen tres posibilidades: el *abstract level* (la salida es una única etiqueta o un subconjunto de las etiquetas posibles), el *rank level* (se devuelven las etiquetas o un subconjunto de ellas ordenadas según el orden de preferencia) y el *measurement level* (el clasificador atribuye a cada etiqueta un valor indicativo de la confianza que se tiene en ella).

3. Métodos de Combinación

Algunos de los métodos de combinación más importantes son los que pasamos a comentar a continuación.

Votación

Uno de los métodos de combinación más populares es el basado en la llamada Teoría de Elección Social, que relaciona el concepto de *elecciones* con el problema de clasificación, el de *votes* con los clasificadores implicados y el de *votos* con las etiquetas propuestas por cada uno de ellos. La técnica de *votación por mayoría* decreta como vencedor a aquel que consiga superar en votos a todos sus rivales, y aunque el criterio más popular es el de *mayoría plural* (el candidato con más votos gana), existen otros como el de mayoría simple (el candidato que supera el 50 % de los votos gana) o por unanimidad (el candidato debe recibir todos los votos para ganar). Estos métodos tienen la ventaja de ser independientes de los datos ya que no requieren entrenamiento previo para su posterior ejecución sobre los ejemplos a clasificar. También existe la posibilidad de ponderar los votos para darle más importancia a unos clasificadores que a otros basándonos en criterios subjetivos o en las respectivas tasas de acierto. Estos métodos también pueden agruparse en mayoritarios y posicionales en función de como se valoran los votos. En el primer caso se realizan comparaciones por pares, y en el segundo la valoración de los candidatos se cuantifica en base a su posición relativa al resto. Los métodos *Condorcet* y *Borda Count* son los métodos más representativos de estos dos grupos respectivamente.

Operaciones sobre valores continuos

Probablemente el método más sencillo es el que podemos aplicar siempre que obtenemos salidas catalogadas en el *measurement level*, a través de la aplicación de una función combinatoria (i.e. media, máximo, media armónica o producto). Son métodos sencillos que no requieren entrenamiento previo, aunque también podemos hacer uso de otras funciones que sí lo requieran como puede ser la media ponderada.

Teorema de Bayes

La aplicación del Teorema de Bayes ha dado muy buenos resultados en múltiples tareas, incluso superando a métodos más complejos. En este caso, se trata de calcular la probabilidad a posteriori de que dadas c etiquetas posibles y dada una serie de observaciones $s = s_1, \dots, s_L$, la etiqueta real sea ω_k . Las observaciones se corresponden con las etiquetas propuestas por los L clasificadores, representando los valores μ obtenidos para $k = 1, \dots, c$ el grado de apoyo que recibe cada etiqueta:

$$\mu_k(x) \propto P(\omega_k) \prod_{i=1}^L P(s_i|\omega_k)$$

Basados en la memorización

Podemos encontrar diversos algoritmos que comparten una idea fundamental, que consiste en memorizar el comportamiento demostrado por los clasificadores en los ejemplos de entrenamiento, para luego tomar decisiones sobre los nuevos elementos basándose en el pasado. Es el caso del *Behavior Knowledge Space* (BKS) (Huang y Suen, 1995), perteneciente a la familia de métodos multinomiales y que se basa en estimar la probabilidad $P(\omega_k|s)$ para todo $k = 1, \dots, c$ y para toda combinación de votos posible s . Estas probabilidades se almacenan en una tabla indexada junto a los contadores donde se van acumulando las veces en que cada combinación dio lugar a cada una de las etiquetas posibles. Para clasificar un nuevo elemento, se accede a la tabla indexada, recuperándose la etiqueta más utilizada entre los ejemplos que ocasionaron la misma combinación de salidas de los clasificadores participantes. La misma filosofía comparte el método *Decision Template* (Kuncheva, Bezdek, y Duin, 2001), que genera una plantilla $DT(j)$ para cada clase ω_j , que representa la matriz de salidas más representativas de los clasificadores para

dicha categoría. Posteriormente se busca la mínima distancia entre las salidas asociadas al ejemplo que se desea clasificar respecto a cada $DT(j)$.

Basados en características

Las características utilizadas para describir los ejemplos a clasificar suponen otro foco de atención en el que podemos aplicar la combinación. El análisis de características puede dividirse en dos aproximaciones basadas en la selección y la extracción. En el primer caso se aplican algoritmos para obtener subconjuntos de características de forma que se maximice la diversidad, pudiendo tratarse de algoritmos denominados *wrappers*, *filters* o *embedded*. Los *wrappers* utilizan técnicas de aprendizaje para valorar los posibles subconjuntos de características pudiendo aplicar diferentes algoritmos de búsqueda para hallar el subconjunto deseado. Los *filters* son métodos genéricos y suelen formar parte del preprocesado mientras que los *embedded* son métodos incrustados en el propio proceso de entrenamiento del clasificador. La combinación se puede aplicar sobre diferentes subconjuntos obtenidos mediante alguna de estas técnicas, aunque otros métodos como el *Random Subspace Method* (Ho, 1998) directamente combinan subconjuntos obtenidos de forma aleatoria. En cuanto a la extracción, puede llevarse a cabo mediante métodos que se encargan de la construcción de nuevas características a partir de las ya existentes, así como de la reducción de la dimensión del espacio original. Este es el caso del *Singular Value Decomposition*, método por el cual se obtiene una aproximación de la matriz original utilizando los autovalores y autovectores en un proceso de factorización.

Explorando agrupaciones de clasificadores

Los métodos que exploran diversos subconjuntos de clasificadores buscan el “equipo” de participantes mejor dotado para resolver la tarea. Uno de ellos es el basado en las *integrales difusas*, que proviene del análisis de decisiones multicriterio, y consiste en valorar la “bondad” de cada subconjunto de clasificadores atendiendo al concepto de *medida difusa*. Según la descripción, si tomamos X como el conjunto de criterios y $\wp(X)$ como el conjunto de las partes de X , podemos definir como medida difusa la

función que cumple:

$$g : \wp(X) \rightarrow [0, 1], \quad g(\emptyset) = 0, \quad g(X) = 1, \\ A \subset B \Rightarrow g(A) \leq g(B) \quad A, B \in \wp(X)$$

En este caso, los criterios se corresponderían a las salidas aportadas por los diferentes clasificadores, y una vez obtenidas las medidas difusas para los subconjuntos se aplicará la integral, cuyas versiones más populares son las de Sugeno, que puede verse como una media ponderada, y la de Choquet, que se asemeja más a una generalización del concepto de mediana. El algoritmo adaptado a la combinación de clasificadores se muestra en (Pal, 2001). Otro algoritmo que explora los subconjuntos de clasificadores posibles es el basado en la Teoría de Dempster-Shafer, en el que se evalúa el grado de creencia (belief) en una hipótesis cuando disponemos de varias fuentes de evidencia (Rogova, 1994).

Bagging y Boosting

Centrados en el Data level según los enfoques mostrados en 2 nos encontramos las técnicas de *Bagging* y *Boosting* (Witten y Frank, 2000). La primera genera la variabilidad necesaria para la combinación creando diferentes versiones de los datos de entrenamiento mediante un remuestreo con reemplazamiento, procediendo finalmente a un proceso de votación para combinar los clasificadores obtenidos a través de los distintos conjuntos de datos. Boosting por su parte asigna pesos a los ejemplos, reflejando en ellos la dificultad que entraña su clasificación y creando versiones progresivas del corpus dotando de mayor relevancia a los ejemplos más difíciles de clasificar.

ECOC

El método denominado *Error Correcting Output Codes* (ECOC) (Dietterich y Bakiri, 1991) representa un caso particular de combinación de sistemas, ya que no encaja en la mayoría de las taxonomías empleadas habitualmente para agrupar los métodos existentes. Se trata de un método que realiza la combinación en base a las salidas o clases asignadas a los ejemplos, generando la variabilidad en el propio conjunto de etiquetas $C = \omega_1, \dots, \omega_c$, creando particiones binarias de este conjunto, a las que se le asigna un clasificador binario especializado. Para c clases podemos representar cada una de estas particiones a través de un vector binario de

c elementos, donde se indica para cada etiqueta en qué lado de la partición ha caído. Si colocamos todas las particiones formando las columnas de una *matriz de códigos*, las filas resultantes representarán a cada etiqueta en todas las particiones consideradas, recibiendo el nombre de *codeword*, mientras que a cada columna se le asignará un clasificador. Al recibir un nuevo ejemplo se ejecutarán los clasificadores generando un *codeword*, al que se le aplicará alguna medida de similitud sobre la matriz de códigos para determinar la etiqueta más probable.

Meta-aprendizaje

El *Meta-aprendizaje* es otra de las vías que más se han explorado a la hora de combinar clasificadores, entendiendo como tal la utilización de las salidas proporcionadas por un método de aprendizaje automático como entrada para otro. A estos métodos se les suele denominar *Stacking* ya que se puede interpretar este proceso como una pila, en la que un algoritmo se nutre de los resultados del algoritmo en el que se apoya (sean uno o varios). Hay que distinguir algunas variantes entre las que destaca el *Cascading* y el bautizado como *Stacked Generalization* por (Wolpert, 1992). En el primero un clasificador es entrenado con la totalidad del conjunto de datos de entrenamiento, y su salida es empleada como entrada para otro clasificador, que a su vez puede enlazarse con otro y así sucesivamente, mientras que el segundo queda reflejado en la figura 3, donde Z es el corpus de entrenamiento, D_i son los clasificadores y r_i las etiquetas que proponen.

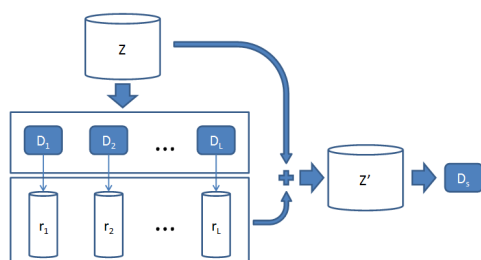


Figura 3: Esquema general del *Stacking*

4. La Combinación y el PLN

Para valorar el grado de presencia de estas técnicas en el área del PLN, hemos analizado un gran número de trabajos en los que se aplica alguna de ellas, extrayendo los de mayor relevancia, principalmente en base al

número de citas recibidas. Un primer análisis de estos trabajos nos muestra la línea temporal que comenzó en los años noventa y tuvo su punto álgido a principios de la pasada década como vemos en la figura 4.

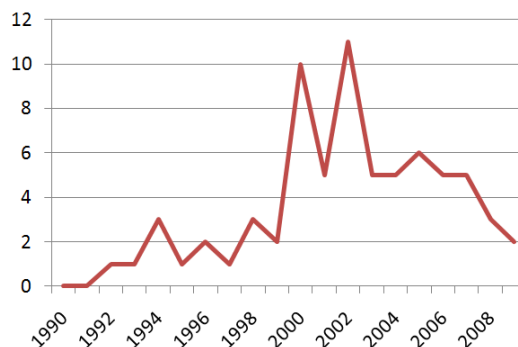


Figura 4: Referencias seleccionadas por año.

4.1. Algunas propuestas

En las próximas secciones pasamos a presentar algunos de los trabajos más representativos en cuanto a la aplicación de la combinación de sistemas al PLN.

Orígenes

En (Belkin y Callan, 1993) se afrontaba la tarea de la Recuperación de Información (IR) combinando diferentes formulaciones booleanas de queries sobre una base de datos de documentos. Un año más tarde, en (Frederking y Nirenburg, 1994) también se combinó un sistema basado en conocimiento con otro basado en ejemplos y un sistema de transferencia léxica para resolver la tarea de la Traducción Automática. A partir de 1998, con la publicación de los trabajos (Halteren, Zavrel, y Daelemans, 1998) y (Brill y Wu, 1998), fue cuando un mayor número de investigadores desarrollaron sus trabajos aplicando las técnicas de combinación a tareas del PLN. Ambos artículos se dedicaban al etiquetado POS, aunque les sucedieron trabajos centrados en otras tareas.

Part-Of-Speech Tagging (POS)

En los trabajos de Halteren y Brill se utilizaron clasificadores basados en modelos de Markov, reglas de transformación y en el algoritmo de máxima entropía, añadiendo el primero un etiquetador basado en ejemplos mientras que el segundo agrega un sencillo algoritmo basado en unigramas a la lista de clasificadores empleados. Ambos muestran mejoras sobre todos los clasi-

ficadores base mediante votación y Stacking aunque este último método obtiene los mejores resultados. En (Borin, 2000) el problema se afronta desde otra perspectiva, mediante la utilización de un sistema de combinación basado en reglas motivadas lingüísticamente. Esto hace de él un sistema pionero al ser un sistema de combinación basado en conocimiento, intentando profundizar en cómo deben utilizarse las diferencias existentes entre los diferentes clasificadores para conseguir mejores resultados en la asignación final de las etiquetas. En (Kuta et al., 2008) se afronta la tarea una vez más con un idioma menos prolífico que el inglés en cuanto a disponibilidad de recursos como es el polaco. Aquí de nuevo se emplean métodos de votación sobre diversos etiquetadores conocidos.

Word Sense Disambiguation (WSD)

En la tarea de WSD sorprenden las mejoras obtenidas por (Pedersen, 2000) con un sistema de combinación bastante sencillo, en el que básicamente se varía el tamaño de las ventanas de contexto para obtener la variabilidad necesaria partiendo de un clasificador basado en el algoritmo Naïve Bayes. Los experimentos se llevaron a cabo con los sustantivos *line* e *interest* y con un sencillo sistema de votación por mayoría aplicado sobre nueve versiones del clasificador bayesiano lograron sobrepasar a otros sistemas anteriores. En (Florian et al., 2002) se completa el trabajo que los autores presentaron con éxito al certamen SENSEVAL-2 combinando cuatro clasificadores mediante técnicas de votación. Concretamente se agregan nuevos clasificadores consiguiendo un conjunto diverso de métodos muy populares y otros más novedosos, además de realizarse nuevos experimentos basados en técnicas de stacking, con los que se obtuvieron resultados por encima de las mejores cifras conocidas hasta la fecha. En otro trabajo interesante, (Klein et al., 2002) muestra como la combinación de sistemas que alcanzan resultados mediocres puede alcanzar medidas de precisión a la altura de los mejores sistemas publicados para esta tarea. En este caso se recopilaron 23 trabajos de alumnos matriculados en el curso de PLN ofrecido por la Universidad de Stanford, generando un conjunto de clasificadores muy heterogéneo. Más recientemente vemos como Cuong introduce en su Tesis Doctoral (Cuong, 2007) dedicada al WSD el con-

cepto de *Meta-Stacking*, que se diferencia del modelo de stacking habitual por utilizar como entradas para el segundo nivel de aprendizaje los resultados obtenidos por diferentes métodos de combinación aplicados sobre los clasificadores base, en lugar de usar directamente las etiquetas proporcionadas por éstos.

Named Entity Recognition (NER)

El grupo de Florian vuelve a dedicar esfuerzos a la combinación, en este caso sobre la tarea del NER en (Florian, 2002), empleando Cascading sobre los clasificadores fnTBL y Snow. Más tarde en (Jing et al., 2003) y (Florian et al., 2003) estudian el impacto que tiene en los resultados de la combinación la inclusión de recursos adicionales como gazetteers y las salidas de otros clasificadores en el proceso. Otro experimento interesante es el realizado por (Wu, Ngai, y Carpuat, 2003), en el que se combinan fases de votación y stacking para dar lugar a un esquema *stacked-voted-stacked*, con el que se consiguió un valor $F_{\beta=1} = 89,51$ que supera en casi 5 puntos al mejor sistema del CoNLL-2002 y supone una reducción del error del 19.7%. En (Kozareva et al., 2007) por otro lado se muestran, además de la aplicación de diversos métodos de votación a un conjunto heterogéneo de clasificadores, la aportación del sistema resultante a diversas tareas finales, como la recuperación de información geográfica o el modelado conceptual. Utilizando modelos de Markov, máxima entropía y algoritmos basados en ejemplos, se obtiene un reconocedor de entidades, que tras ser evaluado sobre los datos de las competiciones CoNLL-2002 (español y holandés) y HAREM-2005 (portugués) se situaría en tercera posición (de 15 participantes) y quinta (de 13 participantes) respectivamente. El reconocimiento de entidades en textos biomédicos es otra área en la que se ha experimentado con técnicas de combinación. Una muestra es el trabajo (Dimililer, Varoğlu, y Altınçay, 2007), donde se mezclan clasificadores con buena precisión con otros que alcanzan buenos porcentajes de cobertura. En este último se hace uso de los algoritmos genéticos para medir la diversidad en base al rendimiento combinado obtenido con los datos de entrenamiento.

Otras tareas

Existen otras tareas que se llevan beneficiando de las virtudes de la combinación des-

de hace tiempo, como demuestran los trabajos (Henderson y Brill, 1999) para el Parsing (PAR) o (Hull, Pedersen, y Schutze, 1996) y (Larkey y Croft, 1996) para la clasificación de documentos (DC). Otros continúan aprovechándolas actualmente gracias por ejemplo a la aparición de nuevos algoritmos de clasificación, como ocurre en (Silva y Ribeiro, 2007) con el uso de los *RVM* (Relevance Vector Machine) en donde se aprovechan las ventajas que aportan frente a los habituales *SVM*.

4.2. Análisis de la bibliografía

Tras recopilar un gran número de trabajos en los que se aplican técnicas de combinación a alguna tarea del PLN, seleccionamos los 70 trabajos más relevantes basándonos en el número de citas o el lugar de publicación entre otros factores. En la figura 5 se aprecia el reparto en cuanto a las tareas que han recibido más atención en los trabajos seleccionados. En cuanto a las técnicas de clasificación y de combinación apreciamos un uso dispar, ya que en lo referente a los algoritmos de clasificación, si bien hay algunos métodos que destacan ligeramente del resto, existe un mayor equilibrio en cuanto a la frecuencia de uso. En los métodos de combinación sin embargo, los métodos de votación y Stacking acaparan la mayor parte de los trabajos, dejando entrever una posible falta de experimentación con el resto de métodos que hemos comentado y que podrían ofrecer mejoras en múltiples tareas de interés para el ámbito del PLN.

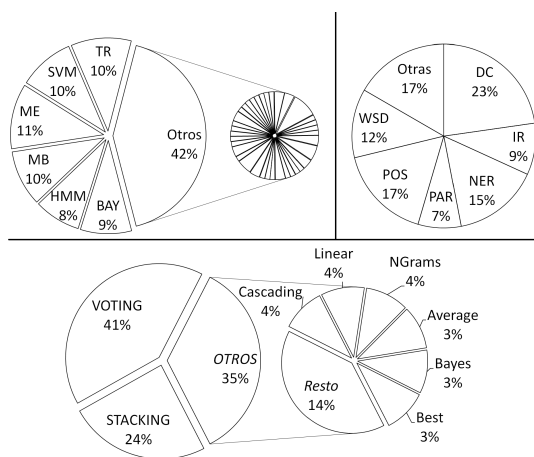


Figura 5: Clasificadores empleados, tareas afrontadas y métodos de combinación aplicados en las referencias seleccionadas.

En cuanto a los resultados presentados en

los trabajos, resulta difícil establecer comparaciones debido a la gran variedad de métodos de clasificación y combinación, además de las tareas y los datos, contabilizándose cerca de 50 corpus distintos. Aún así hemos confeccionado una tabla donde se reflejan las mejoras mínima, máxima y media alcanzadas en los trabajos que aplican combinación de sistemas a alguna tarea del PLN. Además, mostramos estos mismos valores para los casos en los que se ha aplicado un método de votación o Stacking (las técnicas de combinación más utilizadas según se aprecia en la figura 5) así como para el grupo formado por el resto de técnicas aquí comentadas. Lejos de querer darle importancia a los valores concretos, pretendemos interpretar estos resultados como claros indicios sobre las capacidades de mejora que pueden lograrse en muchas tareas del PLN, así como mostrar la valía de muchos métodos ya existentes que gozan de mucha menos popularidad que los métodos de votación y Stacking. Respecto a las tareas, podemos apreciar diferencias significativas entre los valores de mejora, lo cual es fruto, desde nuestro punto de vista, de la dificultad de la tarea y del margen de mejora de los clasificadores individuales. Resulta más difícil mejorar resultados en tareas como el etiquetado POS, donde partimos de resultados muy buenos antes de aplicar la combinación.

	mínimo	máximo	media
DC	0,01	8,10	2,02
NER	1,30	6,41	3,52
PAR	0,03	2,30	1,12
POS	-0,58	1,75	0,75
WSD	1,70	7,00	3,34
Voting	-0,58	6,20	1,53
Stacking	0,03	9,02	2,45
Otros	0,02	8,10	2,62

Tabla 1: Resumen de los resultados de las referencias seleccionadas.

5. Conclusiones

Las técnicas de combinación intentan aprovechar los puntos fuertes de los sistemas individuales minimizando los efectos negativos de sus puntos débiles y obteniendo clasificadores más robustos y precisos. Si bien existen bastantes trabajos que usan algunas de estas técnicas, existen técnicas desarro-

lladas fundamentalmente en el área del Reconocimiento de Patrones con muy buenos resultados que apenas tienen cobertura en el PLN. Hemos mostrado los aspectos principales de las técnicas más importantes, no sin antes recuperar los principios teóricos en los que se sustenta esta línea de investigación que desde la pasada década ha ido ganando protagonismo. Como resultado del estudio aquí realizado, concluimos en primer lugar que el margen de mejora depende en gran parte de la precisión de los clasificadores individuales, siendo más difícil mejorarlos cuando parten de valores altos. Además, apreciamos una mayor diversificación en cuanto a los algoritmos de clasificación que utilizamos habitualmente, frente a las técnicas de combinación, entre las que destacan claramente los algoritmos de Votación y Stacking, quizás por el desconocimiento en lo referente a la existencia de otras muchas técnicas como las expuestas en la sección 3. Dadas las capacidades de mejora que muestran los resultados de este estudio, creemos en la necesidad de contemplar el uso de estas técnicas para la mejora de cualquier módulo que lleve a cabo una labor de clasificación.

Bibliografía

- Belkin, N.J. y J.P. Callan. 1993. The effect of multiple query representations on information retrieval system performance. *Proceedings of the ACM SIGIR*, páginas 339–346.
- Borin, L. 2000. Something borrowed, something blue: Rule-based combination of pos taggers. *Proceedings of the 2nd International Conference on Language Resources and Evaluation*.
- Brill, E. y J. Wu. 1998. Classifier combination for improved lexical disambiguation. *Proceedings of the ACL Conference*, páginas 191–195.
- Cuong, Le Anh. 2007. *A Study of Classifier Combination and Semi-Supervised Learning for Word Sense Disambiguation*. Ph.D. tesis, Japan Advanced Institute of Science and Technology.
- Dietterich, T. G. 2000. Ensemble methods in machine learning. *Multiple Classifier Systems, LNCS*, 1857:1–15.
- Dietterich, T.G. y G. Bakiri. 1991. Error-correcting output codes: A general method for improving multiclass inductive learning programs. *In Proceedings of AAAI-91*, páginas 572–577.
- Dimililer, N., E. Varoğlu, y H. Altınçay. 2007. Vote-based classifier selection for biomedical ner using genetic algorithms. *IbPRIA 2007, Part II, LNCS*, 4478:202–209.
- Florian, R. 2002. Named entity recognition as a house of cards: classifier stacking. *COLING-02: proceedings of the 6th conference on Natural language learning*, páginas 1–4.
- Florian, R., S. Cucerzan, C. Schafer, y D. Yarowsky. 2002. Combining classifiers for word sense disambiguation. *Natural Language Engineering*, 4(8):327–341.
- Florian, R., A. Ittycheriah, H. Jing, y T. Zhang. 2003. Named entity recognition through classifier combination. *Proceedings of HLT-NAACL*, páginas 168–171.
- Frederking, R. y S. Nirenburg. 1994. Three heads are better than one. *In Proceedings of ANLP*, páginas 95–100.
- Halteren, H.V., J. Zavrel, y W. Daelemans. 1998. Improving data driven wordclass tagging by system combination. *Proceedings of the ACL Conference*, 1:491–497.
- Hansen, L. y P. Salamon. 1990. Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(10):993–1001.
- Henderson, J.C. y E. Brill. 1999. Exploiting diversity in natural language processing: Combining parsers. *In Joint Sigdat Conference on EMNLP and VLC (ACL)*, páginas 187–194.
- Ho, T.K. 1998. The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):832–844.
- Huang, Y. S. y C. Y. Suen. 1995. A method of combining multiple experts for the recognition of unconstrained handwritten numerals. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17:90–93.
- Hull, D.A., J.O. Pedersen, y H. Schutze. 1996. Method combination for document filtering. *SIGIR Forum*, páginas 279–288.

- Jing, H., R. Florian, X. Luo, T. Zhang, y A. Ittycheriah. 2003. Howtogetachinese-name(entity): segmentation and combination issues. *Proceedings of EMNLP*, páginas 200–207.
- Klein, D., K. Toutanova, H.T. Ilhan, S.D. Kamvar, y C.D. Manning. 2002. Combining heterogeneous classifiers for word-sense disambiguation. *Proceedings of the ACL-02 workshop on Word sense disambiguation*, páginas 74–80.
- Kozareva, Z., O. Ferrández, A. Montoyo, R. Muñoz, A. Suárez, y J. Gómez. 2007. Combining data-driven systems for improving named entity recognition. *Data & Knowledge Engineering*, 61(3):449–466.
- Kuncheva, L.I., J.C. Bezdek, y R.P.W. Duin. 2001. Decision templates for multiple classifier fusion: An experimental comparison. *Pattern Recognition*, 34(2):299–314.
- Kuncheva, L.I. y C.J. Whitaker. 2003. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, 51:181–207.
- Kuta, M., M. Wrzeszcz, P. Chrzęszcz, y J. Kitowski. 2008. Accuracy of baseline and complex methods applied to morphosyntactic tagging of polish. *ICCS '08: Proceedings of the 8th international conference on Computational Science, Part I*, páginas 903–912.
- Larkey, L.S. y W.B. Croft. 1996. Combining classifiers in text categorization. *SIGIR Forum*, páginas 289–297.
- Pal, Sankar K. 2001. *Pattern recognition: from classical to modern approaches*. World Scientific.
- Pedersen, T. 2000. A simple approach to building ensembles of naive bayesian classifiers for word sense disambiguation. *Proceedings of the ACL Conference*, páginas 63–69.
- Rogova, G. 1994. Combining the results of several neural network classifiers. *Neural Networks*, 7:777–781.
- Silva, C. y B. Ribeiro. 2007. Rvm ensemble for text classification. *International Journal of Computational Intelligence Research*, 3(1):31–35.
- Witten, Ian H. y Eibe Frank. 2000. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann.
- Wolpert, D.H. 1992. Stacked generalization. *Neural Networks*, 5:241–259.
- Wu, D., G. Ngai, y M. Carpuat. 2003. A stacked, voted, stacked model for named entity recognition. *Proceedings of HLT-NAACL*, páginas 200–203.
- Xu, L., A. Krzyzak, y C. Y. Suen. 1992. Methods of combining multiple classifiers and their application to handwriting recognition. *IEEE Transactions on Systems, Man, and Cybernetics*, 22:418–435.