

ESQUEMAS DE VOTACIÓN BORDA  
APLICADOS AL ETIQUETADO DE ROLES  
SEMÁNTICOS

REALIZADO POR: VLADÍMIR E. ROBLES BYKBAEV



Directores: Antonio Molina Marco, Paolo Rosso

Tesis desarrollada dentro del Master de Inteligencia  
Artificial, Reconocimiento de Formas e Imagen Digital

Departamento de Sistemas Informáticos y Computación

Universidad Politécnica de Valencia

Junio 2010

Vladimir E. Robles Bykbaev: *Esquemas de Votación Borda aplicados al Etiquetado de Roles Semánticos*, Tesis desarrollada dentro del Master de Inteligencia Artificial, Reconocimiento de Formas e Imagen Digital, © Junio 2010





## ABSTRACT

---

In this work, we have studied the possibility of applying Borda and Fuzzy Borda voting schemes to combine semantic role labeling systems. To better select the correct semantic role, among those provided by different experts, we have introduced two measures: the first one calculates the overlap between labeled sentences, whereas the second one adds different scoring levels depending on the verbs that have been parsed.

## RESUMEN

---

En este trabajo se ha estudiado la posibilidad de aplicar esquemas de votación Borda y Fuzzy-Borda para combinar sistemas de etiquetado de roles semánticos. Para seleccionar los roles semánticos propuestos por los expertos, hemos introducido dos medidas: la primera calcula el nivel de solape que existe entre las oraciones etiquetadas, mientras que la segunda agrega niveles de puntuación, de acuerdo a los verbos que se deben analizar.



## PUBLICACIONES

---

Algunas ideas y figuras aparecerán en la siguiente publicación:

*Borda-based voting schemes for Semantic Role Labeling. 13th International Conference on Text, Speech and Dialogue. Lecture Notes in Artificial Intelligence, Vol. 6231, Springer-Verlag. Brno, Czech Republic, September 6–10 2010.*





## AGRADECIMIENTOS

---

En primer lugar quisiera agradecer al Dr. Luciano Bellini, al Dr. Javier Herrán y al Eco. Luis Tobar, y a la Universidad Politécnica Salesiana en conjunto, por brindarme su apoyo para la realización de este posgrado y por impulsar día a día la investigación y el desarrollo de la educación. De igual forma, manifiesto mi eterna gratitud a quienes otrora fueron mis mentores y hoy en día son mis amigos y compañeros en la docencia: Diego Quinde, Germán Parra, Diego Peñaloza, Diego Chacón, Eduardo Pinos y Jhon Calle.

No puedo dejar de dar las gracias a mis tutores Paolo Rosso y Antonio Molina, no sólo por el gran soporte brindado en la realización de este trabajo, sino por la paciencia que han tenido con mi persona y por sus ilustrados consejos.

A mi amigo Ihab Al Khoury, por su apoyo constante para redacción del artículo que fundamenta este trabajo. A Aremy, mi musa, mi compañera y amor, tqm.

Esta tesis ha sido desarrollada en el marco del proyecto MICINN (Plan I+D+i): TEXTENTERPRISE 2.0, Técnicas de Comprensión de textos aplicadas a las necesidades de la Empresa 2.0. (TIN2009-13391-C04-03).



# ÍNDICE GENERAL

---

|                                             |                                                        |    |
|---------------------------------------------|--------------------------------------------------------|----|
| 1                                           | INTRODUCCIÓN                                           | 1  |
| <b>I ESTADO DEL ARTE 5</b>                  |                                                        |    |
| 2                                           | PROCESAMIENTO DEL LENGUAJE NATURAL                     | 7  |
| 3                                           | LOS ROLES SEMÁNTICOS                                   | 11 |
| 3.1                                         | Aplicaciones de los roles semánticos.                  | 13 |
| 4                                           | MODELOS BASADOS EN MAXIMA ENTROPIA                     | 17 |
| 4.1                                         | Entropía y ME                                          | 17 |
| 4.1.1                                       | Entropía                                               | 18 |
| 4.1.2                                       | Máxima Entropía                                        | 19 |
| 5                                           | MODELOS BASADOS EN CAMPOS ALEATORIOS CONDICIONALES     | 21 |
| 5.1                                         | Campos Aleatorios Condicionales                        | 21 |
| 5.1.1                                       | El Problema de la Etiqueta Bias                        | 21 |
| 5.1.2                                       | Campos Aleatorios Condicionales                        | 22 |
| 6                                           | MODELOS BASADOS EN MAQUINAS DE SOPORTE VECTORIAL       | 23 |
| 6.1                                         | Maquinas de Soporte Vectorial                          | 23 |
| 7                                           | ADABOOST                                               | 27 |
| <b>II ETIQUETADO DE ROLES SEMÁNTICOS 29</b> |                                                        |    |
| 8                                           | COMPETICIÓN CONLL                                      | 31 |
| 8.1                                         | Competición CoNLL 2005: Etiquetado de Roles Semánticos | 31 |
| 8.2                                         | Corpus CoNLL 2005                                      | 32 |
| 8.3                                         | Etiquetado de Roles Semánticos                         | 34 |
| 8.3.1                                       | Etiquetado basado en modelos de ME                     | 34 |
| 8.3.2                                       | Etiquetado basado en modelos de CAC                    | 37 |
| 8.3.3                                       | Etiquetado de roles semánticos usando MSV              | 39 |
| 8.4                                         | Mejores sistemas de la competición CoNLL 2005          | 43 |
| 8.4.1                                       | Sistema Punyakanok                                     | 44 |
| 8.4.2                                       | Sistema Haghghi                                        | 46 |
| 8.4.3                                       | Sistema Marquez                                        | 48 |
| 8.4.4                                       | Sistema Pradhan                                        | 49 |
| 8.4.5                                       | Sistema Surdeanu                                       | 50 |
| <b>III ESQUEMAS DE VOTACIÓN 53</b>          |                                                        |    |
| 9                                           | ESQUEMAS DE VOTACIÓN BORDA                             | 55 |
| 9.1                                         | Regla de Borda                                         | 55 |
| 9.1.1                                       | Borda clásico aplicada al ERS                          | 57 |
| 9.2                                         | Fuzzy-Borda aplicada al ERS                            | 60 |
| <b>IV DESARROLLO Y EXPERIMENTACIÓN 63</b>   |                                                        |    |
| 10                                          | DESARROLLO DEL SISTEMA                                 | 65 |
| 10.1                                        | Metodologías de analisis                               | 65 |
| 10.1.1                                      | Nivel de solapamiento entre roles candidatos           | 65 |
| 10.1.2                                      | Análisis puntuado de verbos                            | 68 |
| 10.2                                        | Diseño del Sistema                                     | 70 |
| 10.2.1                                      | Estructura general                                     | 70 |
| 10.2.2                                      | Estructura modular                                     | 70 |
| 11                                          | EXPERIMENTACIÓN Y RESULTADOS                           | 73 |

|      |                                                             |    |
|------|-------------------------------------------------------------|----|
| 11.1 | Resultados corpus tWSJ                                      | 74 |
| 11.2 | Resultados corpus tBrown                                    | 75 |
| 11.3 | Resultados corpus tBrown para parámetros $\alpha$ y $\beta$ | 77 |
| V    | CONCLUSIONES Y TRABAJO FUTURO                               | 85 |
| 12   | CONCLUSIONES Y TRABAJO FUTURO                               | 87 |
| VI   | REFERENCIAS Y ANEXOS                                        | 89 |
|      | BIBLIOGRAFÍA                                                | 91 |

## ÍNDICE DE FIGURAS

---

|           |                                                                                                                                                                                                                                                                                                                                                                                                            |    |
|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Figura 1  | Ejemplo de etiquetado léxico. Se puede apreciar que para la frase “Este río está seco” existen varias palabras que admiten diferentes categorías léxicas: “Este” puede ser Determinante (D), Pronombre (P) o Nombre (N). De igual forma ocurre con las demás palabras. A la salida se observa las categorías correctas. Fuente: [58]                                                                       | 8  |
| Figura 2  | Arquitectura de un sistema de recuperación de información. Como se aprecia, la consulta se representa de forma que se pueda comparar con la descripción de los documentos contenidos en el repositorio de datos. Fuente: [75]                                                                                                                                                                              | 9  |
| Figura 3  | Ejemplo del proceso de extracción de opiniones. Se puede apreciar que se llevan a cabo tres sub-tareas: 1) recolección de las palabras de opinión y los marcos semánticos relacionados con las opiniones, 2) etiquetado de roles semánticos para esos marcos y 3) mapeo de roles semánticos a fuentes de opinión y tema. Fuente: [37]                                                                      | 15 |
| Figura 4  | Autómata ejemplo, problema de la etiqueta bias [41].                                                                                                                                                                                                                                                                                                                                                       | 21 |
| Figura 5  | Frontera de decisión $m$ en las MSV.                                                                                                                                                                                                                                                                                                                                                                       | 23 |
| Figura 6  | Casos lineal y no linealmente separables. Medidas de violación de restricciones $\xi_i$ .                                                                                                                                                                                                                                                                                                                  | 26 |
| Figura 7  | Ejemplo en pseudocódigo del algoritmo Ada-Boost para tareas de clasificación binarias. Se puede apreciar que después de seleccionar el clasificador óptimo $h_t$ para la distribución $D_t$ , los ejemplos $x_i$ que el clasificador $h_t$ identificó correctamente se pesan con un menor valor. En el caso contrario, a los ejemplos que se han identificado incorrectamente se les asigna un peso mayor. | 28 |
| Figura 8  | Ejemplo de una oración anotada del corpus WSJ.                                                                                                                                                                                                                                                                                                                                                             | 34 |
| Figura 9  | Árbol de análisis sintáctico etiquetado con roles semánticos respecto al predicado <i>sell</i> [22].                                                                                                                                                                                                                                                                                                       | 37 |
| Figura 10 | Ejemplo de una oración anotada [50].                                                                                                                                                                                                                                                                                                                                                                       | 42 |
| Figura 11 | Ejemplo de error producido por el desplazamiento de argumentos [74].                                                                                                                                                                                                                                                                                                                                       | 47 |
| Figura 12 | Ejemplo de un árbol sintáctico con anotaciones de roles [74].                                                                                                                                                                                                                                                                                                                                              | 48 |
| Figura 13 | Ejemplo de mapeo de argumentos semánticos a constituyentes sintácticos [71].                                                                                                                                                                                                                                                                                                                               | 50 |

- Figura 14 Ejemplo de los valores de solapamiento calculados para la combinación de sistemas  $S_1 - S_2 - S_3$ . Se puede apreciar que los 3 sistemas coinciden en los roles candidatos para el caso del verbo *screwed*, mientras que para el verbo *said* existe un mayor valor de solapamiento entre los sistemas  $S_1$  y  $S_3$ . 68
- Figura 15 Esquema general de funcionamiento de la propuesta de combinación realizada. Se pueden observar 4 elementos fundamentales: a) sistemas participantes ( $S_1$  a  $S_5$ ), b) opciones de combinación, c) sistema que realiza la combinación y d) *script* de evaluación de la competición CoNLL 2005. 70
- Figura 16 Esquema modular de la propuesta de combinación de los sistemas de ERS. Como se aprecia, para combinación y conjunto de parámetros se generará un resultado diferente. 72
- Figura 17 Roles anotados correctamente. Se puede apreciar una comparativa entre los valores promedio de los mejores sistemas originales, todas las combinaciones, el mejor sistema original y la mejor combinación. 74
- Figura 18 Roles sobreclasificados y omitidos. Se observa una comparativa entre los valores promedio de los mejores sistemas originales, todas las combinaciones, el mejor sistema original y la mejor combinación. 75
- Figura 19 Precisión, *recall* y *F-Measure* de los mejores sistemas originales y combinaciones. Se aprecia que el mejor sistema original posee el mayor valor de precisión. 76
- Figura 20 Roles anotados correctamente, valores puntuales (corpus tWSJ). Se puede apreciar que la combinación  $S_1 - S_2 - S_3 - S_4 - S_5$  posee el índice más alto de roles anotados correctamente. El significado de las abreviaturas es el siguiente: SF-SP=Sin Factor de promedio ponderado con Solape Parcial, SF-SC=Sin Factor de promedio ponderado con Solape Completo y CF-SP=Con Factor de promedio ponderado con Solape Parcial. 77
- Figura 21 Roles sobreclasificados y omitidos, valores puntuales (corpus tWSJ). Como se aprecia, la combinación que posee los dos valores más bajos es la que se conforma de los sistemas  $S_1 - S_3$  sin factor de promedio ponderado. 78
- Figura 22 Precisión, *recall* y *F-Measure* de los mejores sistemas originales y combinaciones, valores puntuales (corpus tWSJ). La combinación que logra obtener valores más altos de precisión, *recall* y por ende, *F-Measure*, es  $S_1 - S_3$  sin factor de promedio ponderado. 78

- Figura 23 Roles anotados correctamente, valores promedio generales (corpus tBrown). Se puede apreciar una comparativa entre los valores promedio de los mejores sistemas originales, todas las combinaciones, el mejor sistema original y la mejor combinación. 79
- Figura 24 Roles sobreclasificados y omitidos, valores promedio generales (corpus tBrown). Se observa una comparativa entre los valores promedio de los mejores sistemas originales, todas las combinaciones, el mejor sistema original y la mejor combinación. 79
- Figura 25 Precisión, *recall* y *F-Measure* de los mejores sistemas originales y combinaciones, valores promedio generales (corpus tBrown). Se puede notar que tanto el valor medio de las combinaciones, como la mejor combinación son superiores a los sistemas originales. 80
- Figura 26 Roles anotados correctamente, valores puntuales (corpus tBrown). Como se aprecia, la combinación que alcanza el número de roles anotados correctamente es la que se conforma de los sistemas  $S_1 - S_2 - S_3 - S_4 - S_5$  y utiliza solape completo con factor de promedio ponderado. El significado de las abreviaturas es el siguiente: SF-SP=Sin Factor de promedio ponderado con Solape Parcial y CF-SC=Con Factor de promedio ponderado con Solape Completo. 80
- Figura 27 Roles sobreclasificados y omitidos, valores puntuales (corpus tBrown). Se puede apreciar que los valores conjuntos más bajos pertenecen a la combinación  $S_1 - S_4$  con solape parcial y sin factor de promedio ponderado. 81
- Figura 28 Precisión, *recall* y *F-Measure* de los mejores sistemas originales y combinaciones, valores específicos (corpus tBrown). Se puede observar que las combinaciones obtienen mejores valores que los sistemas originales. Para el caso del *F-Measure* se alcanzan diferencias de 2 puntos. 81
- Figura 29 Variación de la precisión en el combinado de sistemas, de acuerdo a los parámetros de ajuste. La ilustración corresponde a la combinación de los sistemas  $S_1 - S_3 - S_4$  sin factor de promedio ponderado y con solape parcial (corpus tBrown). 82
- Figura 30 Variación del *recall* en el combinado de sistemas, de acuerdo a los parámetros de ajuste. La ilustración corresponde a la combinación de los sistemas  $S_1 - S_3 - S_4$  sin factor de promedio ponderado y con solape parcial (corpus tBrown). 82

|           |                                                                                                                                                                                                                                                            |
|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Figura 31 | Variación del <i>F-Measure</i> en el combinado de sistemas, de acuerdo a los parámetros de ajuste. La ilustración corresponde a la combinación de los sistemas $S_1 - S_3 - S_4$ sin factor de promedio ponderado y con solape parcial (corpus tBrown). 83 |
| Figura 32 | Roles Anotados Correctamente usando Solape Parcial, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tWSJ. 99                                                                                                                                          |
| Figura 33 | Roles Anotados Correctamente usando Solape Completo, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tWSJ. 100                                                                                                                                        |
| Figura 34 | Roles Sobrecalificados usando Solape Parcial, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tWSJ. 100                                                                                                                                               |
| Figura 35 | Roles Sobrecalificados usando Solape Completo, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tWSJ. 101                                                                                                                                              |
| Figura 36 | Roles Omitidos usando Solape Parcial, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tWSJ. 101                                                                                                                                                       |
| Figura 37 | Roles Omitidos usando Solape Completo, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tWSJ. 102                                                                                                                                                      |
| Figura 38 | Precisión de las Combinaciones usando Solape Parcial, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tWSJ. 102                                                                                                                                       |
| Figura 39 | Precisión de las Combinaciones usando Solape Completo, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tWSJ. 103                                                                                                                                      |
| Figura 40 | Recall de las Combinaciones usando Solape Parcial, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tWSJ. 103                                                                                                                                          |
| Figura 41 | Recall de las Combinaciones usando Solape Completo, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tWSJ. 104                                                                                                                                         |
| Figura 42 | <i>F-Measure</i> de las Combinaciones usando Solape Parcial, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tWSJ. 104                                                                                                                                |
| Figura 43 | <i>F-Measure</i> de las Combinaciones usando Solape Completo, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tWSJ. 105                                                                                                                               |
| Figura 44 | Roles Anotados Correctamente usando Solape Parcial, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tBrown. 105                                                                                                                                       |
| Figura 45 | Roles Anotados Correctamente usando Solape Completo, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tBrown. 106                                                                                                                                      |
| Figura 46 | Roles Sobrecalificados usando Solape Parcial, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tBrown. 106                                                                                                                                             |
| Figura 47 | Roles Sobrecalificados usando Solape Completo, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tBrown. 107                                                                                                                                            |
| Figura 48 | Roles Omitidos usando Solape Parcial, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tBrown. 107                                                                                                                                                     |
| Figura 49 | Roles Omitidos usando Solape Completo, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tBrown. 108                                                                                                                                                    |
| Figura 50 | Precisión de las Combinaciones usando Solape Completo, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tBrown. 108                                                                                                                                    |



|           |                                                                                                                                         |
|-----------|-----------------------------------------------------------------------------------------------------------------------------------------|
| Figura 51 | Precisión de las Combinaciones usando Solape Parcial, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tBrown. <a href="#">109</a>  |
| Figura 52 | Recall de las Combinaciones usando Solape Parcial, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tBrown. <a href="#">109</a>     |
| Figura 53 | Recall de las Combinaciones usando Solape Completo, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tBrown. <a href="#">110</a>    |
| Figura 54 | F-Measure de las Combinaciones usando Solape Parcial, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tBrown. <a href="#">110</a>  |
| Figura 55 | F-Measure de las Combinaciones usando Solape Completo, con factores $\alpha = 1,0$ y $\beta = 1,0$ , corpus tBrown. <a href="#">111</a> |
| Figura 56 | Niveles de precisión, recall y F-Measure del sistema Punyakanok, corpus tWSJ. <a href="#">112</a>                                       |
| Figura 57 | Niveles de precisión, recall y F-Measure del sistema Haghighi, corpus tWSJ. <a href="#">113</a>                                         |
| Figura 58 | Niveles de precisión, recall y F-Measure del sistema Marquez, corpus tWSJ. <a href="#">114</a>                                          |
| Figura 59 | Niveles de precisión, recall y F-Measure del sistema Pradhan, corpus tWSJ. <a href="#">115</a>                                          |
| Figura 60 | Niveles de precisión, recall y F-Measure del sistema Surdeanu, corpus tWSJ. <a href="#">116</a>                                         |
| Figura 61 | Niveles de precisión, recall y F-Measure del sistema Punyakanok, corpus tBrown. <a href="#">117</a>                                     |
| Figura 62 | Niveles de precisión, recall y F-Measure del sistema Haghighi, corpus tBrown. <a href="#">118</a>                                       |
| Figura 63 | Niveles de precisión, recall y F-Measure del sistema Marquez, corpus tBrown. <a href="#">119</a>                                        |
| Figura 64 | Niveles de precisión, recall y F-Measure del sistema Pradhan, corpus tBrown. <a href="#">120</a>                                        |
| Figura 65 | Niveles de precisión, recall y F-Measure del sistema Surdeanu, corpus tBrown. <a href="#">121</a>                                       |

## INDICE DE TABLAS

---

|          |                                                                                                      |
|----------|------------------------------------------------------------------------------------------------------|
| Table 1  | Relación información e incertidumbre. <a href="#">19</a>                                             |
| Cuadro 2 | Características del corpus CoNLL 2005 [18]. <a href="#">33</a>                                       |
| Cuadro 3 | Ejemplo del formato de almacenamiento y etiquetado de un oración del corpus tWSJ. <a href="#">33</a> |
| Cuadro 4 | Características empleadas por [42]. <a href="#">35</a>                                               |
| Cuadro 5 | Resultados obtenidos usando del enfoque propuesto por [42]. <a href="#">36</a>                       |
| Cuadro 6 | Características empleadas por [43]. <a href="#">36</a>                                               |
| Cuadro 7 | Resultados obtenidos usando del enfoque propuesto por [43]. <a href="#">37</a>                       |
| Cuadro 8 | Resultados obtenidos usando el enfoque basado en CAC [22]. <a href="#">39</a>                        |

|           |                                                                                                                                                                                         |    |
|-----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Cuadro 9  | Características empleadas por [55].                                                                                                                                                     | 40 |
| Cuadro 10 | Resultados obtenidos usando del enfoque propuesto por [55].                                                                                                                             | 41 |
| Cuadro 11 | Representación de esquemas IOB [73].                                                                                                                                                    | 41 |
| Cuadro 12 | Comparación de los métodos frase a frase y palabra a palabra [32].                                                                                                                      | 41 |
| Cuadro 13 | Resultados obtenidos usando del enfoque propuesto por [50].                                                                                                                             | 43 |
| Cuadro 14 | Listado de los 5 mejores sistemas de la competición CoNLL 2005.                                                                                                                         | 44 |
| Cuadro 15 | Comparativa de etiquetado realizado por 3 mejores sistemas de una oración del corpus tWSJ.                                                                                              | 58 |
| Cuadro 16 | Orden de preferencia de los roles candidatos etiquetados por cada sistema.                                                                                                              | 58 |
| Cuadro 17 | Orden de preferencia de los roles candidatos etiquetados por cada sistema, luego de aplicar la regla Borda.                                                                             | 59 |
| Cuadro 18 | Orden de preferencia de los roles candidatos etiquetados por cada sistema, se agregan pesos o niveles de certeza.                                                                       | 60 |
| Cuadro 19 | Orden de preferencia de los roles candidatos etiquetados por cada sistema, luego de aplicar la regla Fuzzy-Borda.                                                                       | 61 |
| Cuadro 20 | Oración del corpus tWSJ, etiquetada por los sistemas Punyakanok y Haghghi.                                                                                                              | 66 |
| Cuadro 21 | Oración del corpus tWSJ, etiquetada por los sistemas Punyakanok y Haghghi. Se incluyen los valores que genera el análisis puntuado de verbos.                                           | 69 |
| Cuadro 22 | Oración del corpus tWSJ, etiquetada por los sistemas Punyakanok y Haghghi.                                                                                                              | 73 |
| Cuadro 23 | Ejemplo de combinaciones ejecutadas con variación de los parámetros $\alpha$ y $\beta$ . De igual forma se procede para las combinaciones de sistemas $S_1 - S_5$ y $S_1 - S_3 - S_4$ . | 73 |

## ACRÓNIMOS

---

CAC Campos Aleatorios Condicionales

CoNLL Conference on Computational Natural Language Learning

SN Sintagma Nominal

ME Máxima Entropía

MSV Máquinas de Soporte Vectorial

PLN Procesamiento del Lenguaje Natural

## INTRODUCCIÓN

---

Uno de los bienes más preciados que posee la humanidad, es el conocimiento. Por ello, desde tiempos inmemoriales el hombre ha buscado la forma de preservar este valioso tesoro. Los antiguos egipcios plasmaron la historia, religión y costumbres de su pueblo en las paredes de sus hermosos templos. Los chinos crearon el primer libro impreso en el año 868 d. n. e., donde recogían las sabias enseñanzas de Siddharta Gautama, El Buda. En el año 1449 Johannes Gutenberg sorprendería a sus contemporáneos con la publicación del primer libro tipográfico del mundo.

Sin embargo, el progreso continuo de las ciencias aplicadas ha permitido idear nuevos métodos y herramientas para almacenar el conocimiento, que otrora fuera contenido únicamente en libros. Si antes se requería del uso de complejos sistemas de impresión para transmitir dicho conocimiento, en la actualidad podemos guardar cientos de libros, miles de fotografías, cientos de horas de música o video en memorias que no superan el centímetro cuadrado de tamaño. De igual forma, realizar búsquedas por cientos de miles de documentos, recuperar información de las ramas y contenidos más variados de la ciencia, expresar ideas o publicar contenidos y en general, realizar diversas tareas de intercambio de información, son actividades que se efectúan en pocos segundos gracias a Internet.

No cabe duda que Internet es una de las plataformas que ha potenciado de forma increíble los diferentes procesos de intercambio de información. Según [27], en el año 2006, la cantidad de información digitalizada fue 3 millones de veces mayor que la de todos los libros que se han escrito hasta el momento.

Por todo esto, el PLN (Procesamiento del Lenguaje Natural) adquiere gran importancia en el contexto actual, ya que permite procesar y manipular la información contenida en diferentes medios y formatos de almacenamiento. Para llevar a cabo estos procesos, el PLN se plantea como objetivo principal el diseño de mecanismos de comunicación eficientes a nivel computacional. Justamente estos mecanismos son los que dan posibilidad a realizar interesantes procesos, como la traducción de texto escrito o habla de un lenguaje a otro de forma automática [79], identificación del sentido de palabras polisémicas de acuerdo al contexto de la oración [54], búsqueda y recuperación de información [52], búsqueda de respuestas a preguntas planteadas en lenguaje natural [67], análisis de textos de tipo subjetivo y de opinión [21, 37] y otras varias aplicaciones no menos importantes.

De este modo, consideramos que es muy importante conocer una metodología que permite desarrollar de forma más sencilla varias de las tareas antes mencionadas: el etiquetado de roles semánticos o papeles temáticos. Por ello, si se cuenta con un corpus anotado semánticamente, se tiene la posibilidad de mejorar el rendimiento general de los sistemas de PLN que realizan análisis de textos. Esta idea se ha constatado ampliamente en trabajos previos, entre los que citamos a [36, 70].

El análisis semántico, al ser parte importante de la lingüística general, define un conjunto de reglas y principios que explican como se asignan

*La Teoría – ∅ define un conjunto de reglas y principios que explican como se asignan roles semánticos a los predicados de una oración...*

estos roles a un determinado predicado en una oración. Este conjunto de reglas se encuentra contenido en la Teoría –  $\theta$  [68]. Dentro de esta teoría se explica que un rol semántico es la relación subyacente que existe entre un constituyente sintáctico (conformado por una palabra o secuencia de palabras) y el verbo principal de una oración. El rol es el papel que el predicado asigna a sus argumentos [51]. Un ejemplo claro de lo que se ha mencionado se representa en la siguiente oración: “*Leónidas escribió una carta*”, tendríamos los siguientes roles: “[*Leónidas*]*agente* *escribió* [*una carta*]*tema*”. El constituyente sintáctico “*Leónidas*” es el quién produce una acción, mientras que el constituyente “*una carta*” representa el argumento que sufre un cambio de estado. Los principales roles temáticos son:

- Agente . - argumentos que producen la acción.
- Experimentador . - argumentos que se someten a una experiencia sensorial, cognitiva o emocional.
- Beneficiario . - argumentos que reciben algo bueno o malo en una situación.
- Fuerza . - argumentos que determinan el causante involuntario de un evento.
- Instrumento . - argumentos que indican el instrumento utilizado en un evento.
- Tema . - argumentos que sufren un cambio de estado.

En virtud de lo expuesto, en este trabajo de tesis revisamos la posibilidad de aplicar una propuesta de votación basada en esquemas Borda y Fuzzy – Borda [28], para combinar diversos sistemas de etiquetado de roles semánticos. Para realizar esta tarea se ha utilizado el conjunto de datos publicado en la *shared task* desarrollada en el marco de la conferencia CoNLL 2005<sup>1</sup>. De estos datos tomamos los corpus etiquetados por los mejores sistemas y definimos dos medidas de análisis combinadas: **el nivel de solapamiento** y **las tablas de puntuación de roles** contenidos en cada oración. El nivel de solapamiento es una medida sencilla que nos permite verificar en que proporción coinciden los roles asignados por cada sistema a los constituyentes de una oración. Las tablas de puntuación se inspiran en el esquema de votación Borda y asignan un valor numérico a los roles etiquetados dentro de una oración, en base al verbo del que dependen y los valores de precisión de cada sistema (también es posible usar *recall* o F – Measure). Tomando en consideración lo que se ha mencionado, podemos establecer los objetivos que se desean alcanzar con este trabajo:

- Combinar varios sistemas de etiquetado de roles semánticos.
- Establecer una medida alternativa para seleccionar los roles asignados a los argumentos de una oración.
- Evaluar la propuesta planteada en este trabajo a través del corpus empleado en la competición CoNLL, de modo que se pueda contrastar correctamente los resultados obtenidos.

<sup>1</sup> El objetivo de la *shared task* de CoNLL 2005 fue la evaluación de sistemas de clasificación de roles semánticos, basados en técnicas de aprendizaje. Los conjuntos de entrenamiento y prueba de cada sistema participante pueden descargarse en la url: <http://www.lsi.upc.edu/~srllconll/st05/st05.html>.

Finalmente, pasamos a detallar cómo se organiza el presente trabajo: en la sección 1 revisamos cuatro de las metodologías más importantes para etiquetado de roles semánticos, esto es, Máxima Entropía (ME), Campos Aleatorios Condicionales (CAC), Máquinas de Soporte Vectorial (MSV) y el algoritmo AdaBoost; en la sección 2 describimos las características más importantes del corpus empleado; en la sección 3 se revisa el esquema de votación Borda y su variante Fuzzy, así como la posibilidad de emplearlo para combinar dos o más sistemas de etiquetado de roles; en la sección 4 detallamos las medidas que hemos empleado para combinar los resultados generados por los sistemas de la *shared task*; en la sección 5 recopilamos los resultados obtenidos y los analizamos; en la sección 6 recogemos las conclusiones e ideas que se puedan desarrollar como trabajo futuro.



Parte I

ESTADO DEL ARTE





El PLN es una subdisciplina de la Inteligencia Artificial y la rama ingenieril de la lingüística computacional. El PLN se ocupa de la formulación e investigación de mecanismos eficaces computacionalmente para la comunicación entre personas o entre personas y máquinas por medio de lenguajes naturales. El PLN no trata de la comunicación por medio de lenguajes naturales de una forma abstracta, sino de diseñar mecanismos para comunicarse que sean eficaces computacionalmente (que se puedan realizar por medio de programas que ejecuten o simulen la comunicación). Los modelos aplicados se enfocan no sólo a la comprensión del lenguaje de por sí, sino a aspectos generales cognitivos humanos y a la organización de la memoria<sup>1</sup>. Los ámbitos de aplicación del PLN son muy variados, entre los más importantes podemos citar los siguientes:

- **Análisis Sintáctico** . - Se encarga de recuperar la estructura o árbol sintáctico que posee una oración. Para realizar esta tarea se requiere establecer una gramática que permita definir las estructuras del lenguaje y en base a ella inferir el árbol sintáctico. Definir una gramática es una tarea costosa, que presenta diversos problemas como: obtener una correcta segmentación, la subgeneración que supone que la gramática no da cobertura a todo el lenguaje y la ambigüedad sintáctica [14]. Debido a esto, se han propuesto las técnicas de análisis parcial (*partial parsing*) y el análisis superficial (*shallow parsing* o *chunking*). El análisis parcial tiene como objetivo recuperar información sintáctica de forma eficiente y fiable, partiendo de textos no restringidos (se sacrifica la completitud y la profundidad del análisis global) [1]. Por su parte, el análisis superficial consiste en dividir al texto en segmentos no solapados que se corresponden con ciertas estructuras sintácticas no recursivas, llamadas *chunks*. Por ejemplo, se identifican sintagmas nominales (SN), sintagmas adjetivos (SADJ), núcleos verbales (SV), sintagmas adverbiales básicos (SADV), etc [58]. Por ejemplo, para la oración “El cartero da al hombre una carta”, se tendrían los siguientes sintagmas nominales básicos: “[SN El cartero] da al [SN hombre] [SN una carta]”.
- **Desambigüación del Sentido de las Palabras (Word Sense Disambiguation)** . - Cuando una palabra posee más de un significado o sentido (fenómeno lingüístico conocido como polisemia), se debe analizar el contexto de dicha palabra y asignarle el sentido correcto. Según se detalla en [76], el contexto de una palabra es considerado como el conjunto de palabras que la acompañan, junto con las relaciones sintácticas y categorías semánticas. Por ejemplo, dentro de la oración: “Juan dejó el periódico en el banco”, puede interpretarse dos casos: a) Juan ha dejado el periódico en un asiento o mesa y b) Juan ha dejado el periódico en alguna institución financiera. Esto se debe a que la palabra banco tiene

<sup>1</sup> Fuente: Wikipedia ([http://es.wikipedia.org/wiki/Procesamiento\\_de\\_lenguajes\\_naturales](http://es.wikipedia.org/wiki/Procesamiento_de_lenguajes_naturales))

varios sentidos: institución financiera, edificio u oficina perteneciente a una institución, grupo numeroso de peces, asiento largo y estrecho y mesa fuerte y robusta.

- Etiquetado Léxico (Pos Tagging) . - Esta es una tarea en la que se toma como entrada frases escritas en un lenguaje determinado y proporciona como salida la categoría léxica más adecuada para cada palabra, tomando en consideración el contexto en el que aparece. El proceso de etiquetado léxico tiene en cuenta dos fuentes de conocimiento: el Modelo del Lenguaje, que describe las probables secuencias de categorías léxicas, y el Modelo Léxico, que representa la relación entre el vocabulario de la aplicación y el conjunto de categorías [58]. En la Figura 1 podemos observar un ejemplo del proceso de etiquetado léxico.

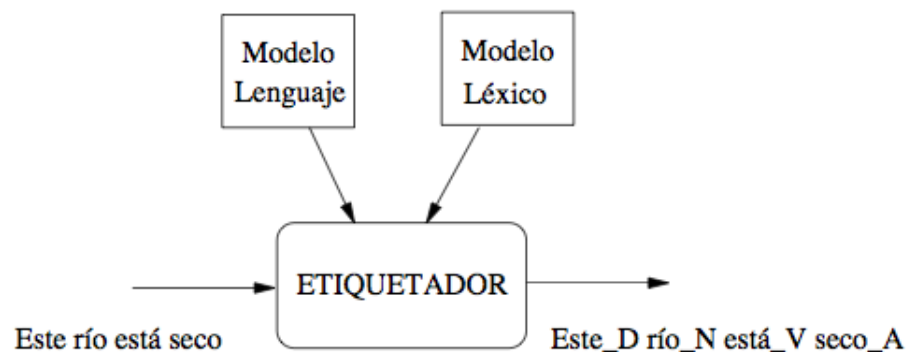


Figura 1: Ejemplo de etiquetado léxico. Se puede apreciar que para la frase “Este río está seco” existen varias palabras que admiten diferentes categorías léxicas: “Este” puede ser Determinante (D), Pronombre (P) o Nombre (N). De igual forma ocurre con las demás palabras. A la salida se observa las categorías correctas. Fuente: [58]

- Etiquetado de Roles Semánticos . - La idea que fundamenta la concepción de los roles semánticos, es que el lenguaje se puede usar para referirse a los participantes de una acción o acontecimiento [5]. Tomando esto en consideración, la anotación o etiquetado de roles semánticos trata de solucionar la ambigüedad que se produce al determinar la agrupación y relaciones entre palabras de una oración . Ejemplos de roles semánticos son: agente, paciente, beneficiario, etc., o también adjuntos, como causa, manera o temporal [51]. Por ejemplo, si consideramos la oración “*María golpeó a Juan con una pelota, ayer en el parque*”, podremos observar los roles que se asignan a los argumentos del verbo “*golpeó*”: “[*María*]<sub>agente</sub> golpeó [*a Juan*]<sub>objeto\_golpeado</sub> [*con una pelota,*]<sub>manera</sub> [*ayer*]<sub>temporal</sub> [*en el parque*]<sub>ubicación</sub>”.

Las aplicaciones antes mencionadas permiten resolver diversos problemas en el tratamiento del lenguaje, y asimismo, se constituyen en tareas base para realizar procesos más complejos como son:

- Traducción automática . - Consiste en la obtención de un texto equivalente (esto es, que preserve el contenido) en una lengua destino a partir de un texto en una lengua origen [2].
- Recuperación de información . - Esta tarea permite responder a las consultas que realiza el usuario. Para realizar esta tarea

se efectúan las siguientes etapas: indexación de la colección de documentos, análisis de la consulta realizada y transformación (en caso de ser necesario), comparación de la descripción de cada documento y presentación de aquellos documentos con la descripción mas parecida a la consulta [75]. En la [Figura 2](#) podemos observar un ejemplo del proceso seguido para recuperar información.

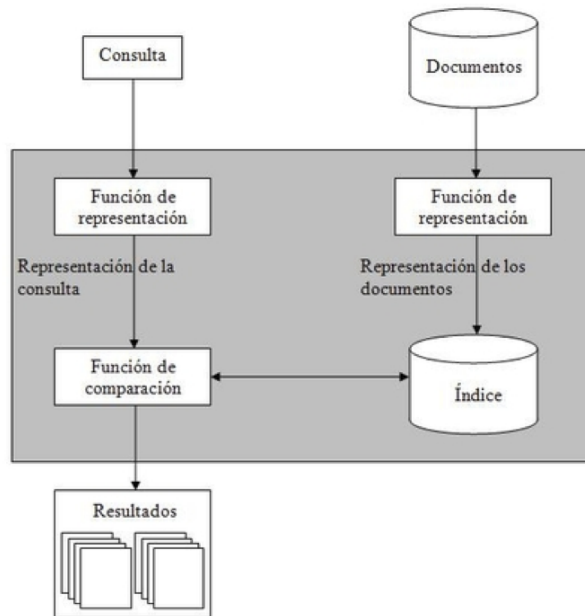


Figura 2: Arquitectura de un sistema de recuperación de información. Como se aprecia, la consulta se representa de forma que se pueda comparar con la descripción de los documentos contenidos en el repositorio de datos. Fuente: [75]

- Extracción de información . - Como sabemos, un sistema de recuperación de información recoge material útil de grandes cantidades de textos, con la finalidad de seleccionar aquellos que sean relevantes. Por otra parte, un sistema que extrae información parte de estos textos y busca obtener información relevante que pueda ser utilizada electrónicamente [24]. Según lo que describe [44], el objetivo de la extracción de información es construir sistemas que encuentren y relacionen información relevante mientras ignoran otras informaciones que no lo sean.

El último aspecto a tener en cuenta es el tipo de aprendizaje que deben llevar los sistemas para poder realizar todas estas tareas. Existen básicamente dos enfoques [51]:

- Enfoques basados en corpus . - También denominado aprendizaje automático (*machine learning*), es una rama de la inteligencia artificial relacionada con el desarrollo de algoritmos y técnicas que permitan que los ordenadores sean capaces de aprender. Los enfoques basados en corpus se pueden clasificar en:
  - Aprendizaje automático supervisado . - Su tarea es predecir el valor de la función que asigna la etiqueta de clase para cualquier objeto de entrada válido, después de haber un número de ejemplos de entrenamiento. Entre los principales

algoritmos de aprendizaje automático supervisado, podemos citar los siguientes: basados en máxima entropía, basados en memoria, máquinas de soporte vectorial, separadores o clasificadores lineales (Winnow), etc.

- Aprendizaje automático semi-supervisado . - Hace uso tanto de datos de entrenamiento etiquetados como no etiquetados para generar una función o clasificador apropiado. Normalmente la cantidad de datos etiquetados es pequeña frente a los no etiquetados. Dentro esta categoría están los algoritmos *bootstrapping* y *co-training*.
- Aprendizaje automático no supervisado . - Todo el proceso de modelado se lleva a cabo sobre un conjunto de ejemplos formado únicamente por las entradas del sistema. No se tiene información sobre las categorías de esos ejemplos. El algoritmo debe ser capaz de descubrir las clases en base a agrupación de ejemplos en categorías similares. Los algoritmos que están en esta categoría se conocen con el nombre de algoritmos de agrupación (*clustering*).
- Enfoques basados en conocimiento . - Los sistemas que se basan en este enfoque buscan resolver problemas usando representación simbólica del conocimiento humano.

## LOS ROLES SEMÁNTICOS

---

Un rol semántico es una relación que existe entre un constituyente sintáctico (generalmente el argumento del verbo, aunque no siempre) y un predicado (generalmente el verbo, aunque no siempre) [51]. A diferencia del nivel sintáctico, donde se ha llegado a un cierto consenso sobre los constituyentes sintácticos y su definición, con los roles semánticos no se tiene un acuerdo sobre que roles existen, ni cuáles son las características de cada uno de ellos. Los motivos que han impedido que se llegue a dicho acuerdo son los siguientes [31]:

- Límites . - Cómo y donde establecer los límites entre tipos de roles dentro de un mismo conjunto.
- Granularidad . - No existe un consenso sobre el número de roles que se necesitan, ni acerca de su cantidad.
- Organización . - Falta de organización interna, en virtud de que el conjunto de roles considerado tiene la forma de lista no estructurada.

Debido a esto, varios investigadores han desarrollado diversas propuestas de roles semánticos. Como ejemplo, podemos citar algunas de las mas importantes :

- *Propuesta de Gruber* . - Conjunto de roles para localización espacial y movimiento: *Theme* (objeto en movimiento o a localizar), *Agent* (produce una acción o estado), *Location* (lugar), *Source* (objeto desde el cual se produce el movimiento), *Path* (camino) y *Goal* (objetivo al que se dirige el movimiento) [64].
- *Propuesta de Fillmore (Gramática de Casos)* . - Plantea que la oración tiene un verbo y un conjunto de casos o roles semánticos, los cuales establecen una relación entre el verbo y los sintagmas nominales de la oración. Estas relaciones ocurren una sola vez en una oración simple. Los roles que define Fillmore son: *Agent* (causante del evento), *Experiencer* (que experimenta el evento), *Force* (causante involuntario de un evento), *Theme* (participante en un evento afectado por el mismo de forma mas directa), *Result* (el producto final de un evento), *Content* (proposición o contenido de un evento proposicional), *Instrument* (instrumento utilizado en un evento), *Beneficiary* (beneficiario del evento), *Source* (origen del objeto en un evento de traslado) y *Goal* (destino de un objeto en un evento de traslado) [64].
- *Propuesta de Sgall (Descripción Generativa Funcional)*. [51] - Consiste en analizar las oraciones en base a dependencias. Trabaja con la representación tectogramática o subyacente de las oraciones. Dicha representación tiene la forma de un árbol de dependencias, de modo que a cada oración se le asigna una estructura de árbol con nodos y arcos etiquetados. Los nodos representan las palabras de la oración y tienen asignado un marco de valencia que incluye información sobre sus valores morfológicos y léxicos. Los arcos

del árbol indican las relaciones de dependencia entre las palabras de la oración y se denominan funtores. Existen dos tipos de dependencias:

- Participantes internos . - Son de dos tipos: sintácticos (actor y paciente, que son primer y segundo participante, respectivamente) y semánticos (dirección, efecto y origen).
  - Modificaciones libres . - Adjuntos (ubicación, tiempo, manera o intención).
- *Propuesta de Schank (Teoría de la Dependencia Conceptual)* . - Tiene como objetivo comprender conceptualmente el lenguaje, a un nivel profundo. Se enfoca en los conceptos y en el entendimiento y se fundamenta en dos axiomas básicos [65]:
    - Para cualquier par de frases que son idénticas en su significado, sin importar el idioma, sólo debería haber una representación.
    - Cualquier información contenida en una frase de forma implícita, se debe poder convertir en explícita en la representación del significado de dicha frase.

De estos dos axiomas, [65] crea su teoría, tomando como base un conjunto primario de conceptualizaciones, que son de dos tipos: acciones (una Acción es realizada por un Actor en un algún Objeto con alguna Dirección) y estados (un Objeto está en algún Estado con algún Valor).

- *Propuesta del Proyecto FrameNet* . - Propone roles, denominados marcos semánticos (*semantic frame*). Los marcos incluyen un conjunto de unidades léxicas, una lista de roles y un conjunto de ejemplos. Una unidad léxica es un par de palabra-sentido. Típicamente, cada sentido de una palabra polisémica pertenece a marcos semánticos diferentes. Por ejemplo, el marco Aplicar\_calor (*Apply\_heat*) describe una situación común que relaciona cocinar (*cook*), comida (*food*) y un instrumento de cocina (*heating\_instrument*) y es evocado por las palabras cocción (*bake*), hervir (*boil*), hervir a fuego lento (*simmer*), etc. Estos roles se denominan elementos de marco (*frame elements*), y las palabras que evocan cada marco son las unidades léxicas del marco Aplicar\_calor [63].
- *Propuesta del Proyecto PropBank* [3]. - Es un conjunto de anotaciones de estructuras analizadas sintácticamente. Dichas estructuras contienen elementos del tipo predicado-argumento. La anotación se realiza a través de etiquetador automático basado en reglas, cuya salida se corrige de forma manual. Un objetivo importante que persigue esta propuesta es proveer etiquetas consistentes para los argumentos, de acuerdo a las diferentes realizaciones sintácticas de un mismo verbo, por ejemplo, para las oraciones: “[ARG<sub>0</sub> John] broke [ARG<sub>1</sub> the window]” y “[ARG<sub>1</sub> The window] broke”, los argumentos del verbo son etiquetados como Argo, Arg1, Arg2, etc. Un segundo objetivo de PropBank es la asignación de etiquetas funcionales a todos los modificadores del verbo, esto es, manera (MNR), ubicación (LOC), temporal (TMP), entre otros.

Las propuestas que se han descrito anteriormente se han utilizado para crear diversos recursos lingüísticos como corpus anotados o léxicos. A continuación detallamos los proyectos más importantes:

- *FrameNet* . - Tiene como objetivo crear un recurso léxico para el inglés con información detallada de las posibles realizaciones sintácticas de elementos semánticos. La actual base de datos de FrameNet cuenta con 8.900 unidades léxicas y 625 marcos semánticos con más de 135.000 oraciones anotadas [51]. Es importante anotar que se cuenta con traducciones que se basan en el proyecto FrameNet a otros idiomas (alemán, español, japonés y rumano).
- *PropBank* . - El proyecto ha generado un recurso léxico de ficheros de frames anotados manualmente (almacena todos los significados de los verbos con su descripción y ejemplos) y un corpus anotado con un etiquetador automático basado en reglas y revisado manualmente. Este recurso, que se basa en el corpus del Wall Street Journal (inglés), cuenta con 3.300 verbos que referencian 4.500 marcos semánticos [51]. Al igual que el proyecto FrameNet, el PropBank ha servido de base para realizar traducciones a otros idiomas: vasco, chino, ruso y árabe.
- *NomBank* . - Tiene como objetivo anotar la estructura nominal de los argumentos en el corpus generado por el proyecto PropBank. Este proceso se fundamenta en la creación de las entradas léxicas que definen los posibles sentidos y argumentos para cada nombre. Por ejemplo, estas entradas léxicas determinan que constituyentes conforman los argumentos ARG<sub>0</sub>, ARG<sub>1</sub>, etc. para un nombre determinado. Una vez que se tiene una entrada léxica, el anotador debe revisar todas las instancias del nombre en el corpus (incluyendo plurales, variaciones ortográficas, etc.) e identificar que instancias son marcables [48].
- *VerbNet* . - Es un recurso léxico organizado de forma jerárquica, independiente del dominio y de cobertura general con equivalencias o mapeos a otros recursos léxicos como WordNet, Xtag y FrameNet [38]. Define un conjunto de 23 roles semánticos: actor, agente, activo, atributo, beneficiario, causa, ubicación, destino, fuente, experimentador, grado, instrumento, material, producto, paciente, predicato, recipiente, estímulo, tema, tiempo y tópico .
- *Áncora* . - Es un corpus anotado del español y del catalán que cuenta con 500 mil palabras para cada lengua. Posee diferentes niveles de anotación: categoría morfológica, constituyentes y funciones sintácticas, estructura argumental y papeles temáticos, clase semántica verbal, sentidos de WordNet nominales y entidades nombradas. Se constituye de textos periodísticos y posee dos léxicos verbales de 2.580 entradas para el español y 2.412 para el catalán, con información sobre la clase semántica del verbo y la subcategorización sintáctica, la estructura argumental y los roles temáticos para cada uno de los sentidos [47].

### 3.1 APLICACIONES DE LOS ROLES SEMÁNTICOS.

En esta sección detallamos un par de las aplicaciones que revisten de interés y que se sustentan en base a los roles semánticos:

- Búsqueda de respuestas [53] . - La habilidad de responder preguntas complejas, que han sido formuladas en lenguaje natural, depende de dos factores: a) la profundidad de las representaciones

semánticas y b) de los mecanismos inferenciales de soporte. Los sistemas clásicos buscan las respuestas a través de la implementación de tres etapas: 1) clasificar el tipo de respuesta esperada, 2) identificar pasajes candidatos de la respuesta en base a palabras clave de la pregunta y 3) puntuar las respuestas candidatas para definir que pasaje contiene la respuesta exacta. Sin embargo, esta aproximación está limitada por la asunción de que la respuesta puede ser encontrada por que usa palabras que forman parte de la pregunta. En ciertas ocasiones si se cumple la asunción antes mencionada, pero se omiten muchas respuestas correctas debido a que se requiere de un mecanismo más sofisticado de análisis que el reconocimiento de entidades nombradas e identificación del tipo de pregunta. En virtud de lo expuesto, un análisis semántico más profundo permitirá responder de forma más eficiente a preguntas complejas. Para lograr esto, se deben realizar llevar a cabo tres pasos fundamentales:

- Incorporar analizadores semánticos . - Su tarea debe ser reconocer las estructuras predicado-argumento o los marcos semánticos cuando procesan las preguntas y los documentos que contienen las respuestas.
  - Identificar el modelo de tema . - Esto permite interpretar mejor la pregunta y genera un posible índice en un conjunto de ontologías.
  - Generar un modelo expresivo y escalable de acciones y eventos . - Con esto se puede llevar a cabo el razonamiento sofisticado que exige la búsqueda de respuestas en escenarios complejos.
- Análisis de opinión . - La extracción de opiniones hoy en día es una tarea relevante, ya que gran parte de la información que generan los usuarios de Internet está contenida en foros, bitácoras (*blogs*), sitios personales, redes sociales, etc. Al llevar a cabo esta tarea, podemos apreciar que la incorporación de roles semánticos permite obtener mejoras importantes en cuanto a precisión y *recall* [21]. Otro aspecto a tener en cuenta, es que cuando se analizan las opiniones se puede identificar tres tipos de elementos: la opinión, su fuente y el tema. Las etapas que se llevan a cabo son las siguientes [37]:
    - Identificar las opiniones.
    - Etiquetar los roles semánticos relacionados con las opiniones.
    - Determinar las fuentes y temas de las opiniones, tomando como base los roles semánticos identificados.
    - Almacenar tripletas del tipo <opinión, fuente, tema> en una base de datos.

En la [Figura 3](#) podemos apreciar un ejemplo que ilustra el proceso seguido para identificar la opinión, su fuente y el tema.



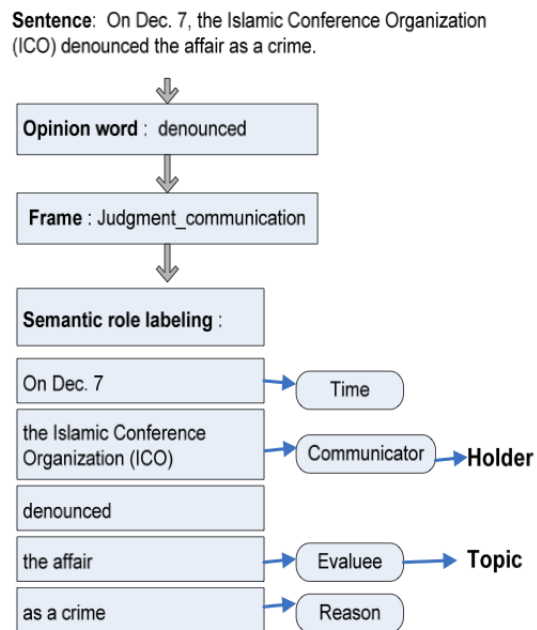


Figura 3: Ejemplo del proceso de extracción de opiniones. Se puede apreciar que se llevan a cabo tres subtareas: 1) recolección de las palabras de opinión y los marcos semánticos relacionados con las opiniones, 2) etiquetado de roles semánticos para esos marcos y 3) mapeo de roles semánticos a fuentes de opinión y tema. Fuente: [37]



## 4.1 ENTROPIA Y ME

Previo a revisar el concepto de Máxima Entropía (ME) y su aplicación, consideramos importante hacer una breve disgresión sobre la estadística, sus enfoques de estudio y la información, ya que esto permitirá situar al principio de ME de mejor forma.

La estadística es una ciencia de fundamento matemático, que trata de explicar diversos fenómenos de naturaleza aleatoria. Su campo de aplicación es muy amplio, ya que es utilizada en áreas tan variadas como la física, las ciencias sociales y la medicina. Dentro de la estadística existen tres posibles enfoques de estudio [62]:

- Clásico . - Se atribuye a Laplace y su fundamento es el conteo de casos que puede ocurrir de un evento, conociendo el número de casos totales (asume que todos los eventos tienen la misma posibilidad de ocurrir). Los problemas que podemos tratar con esta visión son limitados, ya que debemos conocer a priori el número de casos totales.
- Frecuentista o empírico . - En este caso se asume que la probabilidad se basa en una frecuencia relativa. Esta probabilidad tiene sentido cuando se refiere a una hipotética secuencia infinita de realizaciones de un fenómeno. De igual forma que en el caso anterior, los problemas a tratar con este enfoque son limitados, ya que serán aquellos en los que un fenómeno se repita infinidad de veces o al menos una gran cantidad, de modo que se tenga una buena aproximación.
- Bayesiano . - Dentro de este enfoque existen dos corrientes: una basada en probabilidades subjetivas y otra que se centra en consideraciones netamente objetivas. Varios autores consideran que el bayesianismo objetivo provee una estructura teórica con un campo de aplicación más amplio. Muchos de los casos de los enfoques clásico y frecuentista pueden ser tratados por el bayesianismo objetivo. El principio de ME no es una excepción.

*Varios autores consideran que el bayesianismo objetivo provee una estructura teórica con un campo de aplicación más amplio...*

Desde el punto de vista informático, es importante conocer que la información nos permite reducir el nivel de incertidumbre en el receptor de un determinado mensaje, luego de haber conocido el resultado de un fenómeno aleatorio [62]. Si suponemos que una fuente de información entrega símbolos de un alfabeto  $S = \{s_0, s_1, \dots, s_{k-1}\}$ , con probabilidades de aparición  $p_0, p_1, \dots, p_{k-1}$  para cada símbolo, se cumple con la siguiente condición:

$$\sum_{i=0}^{k-1} p_i = 1$$

Ahora asumimos que la fuente de información nos entrega el símbolo  $s_i$ . Si la probabilidad  $p_i$  de que ocurra este símbolo es 1, podremos

ver que la fuente no entrega información, ya que no nos aporta nada nuevo. En el caso contrario, si la fuente entrega un símbolo que no se esperaba, esto es, uno cuya probabilidad de aparición es muy baja, la información que aporta es grande.

Tomando en consideración este razonamiento, podemos emplear “criterios de información” para medir la entropía, como nos explica [69]. Consideremos que  $\Gamma$  es un espacio muestral asociado a un experimento aleatorio. Se puede suponer que este espacio se encuentra formado de particiones de sucesos  $A_1, A_2, \dots, A_n$ . Estos sucesos son incompatibles, es decir,  $A_1 \cup A_2 \cup \dots \cup A_n = \Gamma$  y  $A_i \cap A_j = 0$  para  $i \neq j$ . Cuanto más improbable sea un suceso  $A_k$ , mayor información proporcionará. En virtud de esto, podemos suponer que la información  $I$  de dicho suceso cumple la igualdad  $I(A_k) = f\left(\frac{1}{p_k}\right)$ , es decir, que la información es inversamente proporcional a una cierta función aplicada a la probabilidad de aparición de ese suceso. Para determinar la función  $f$  que relaciona la información y la probabilidad de aparición de un suceso, se consideran las siguientes premisas:

1. Todos los sucesos  $A_i$  son equiprobables, esto es,  $p(A_i) = \frac{1}{n}$ ,  $i = 1, 2, 3, \dots, n$
2. Si realizamos  $m$  experiencias independientes del experimento, se puede considerar la probabilidad de un evento de la forma  $A_{i1}, A_{i2}, \dots, A_{im}$  igual a  $p(A_{i1}, A_{i2}, \dots, A_{im}) = \frac{1}{n^m}$
3. La información que tiene cada evento al ser inversamente proporcional a su probabilidad, permite escribir la igualdad de la siguiente forma:  $I(A_{i1}, A_{i2}, \dots, A_{im}) = n^m$
4. La información de cada suceso  $A_{i1}, A_{i2}, \dots, A_{im}$ , es igual a  $m$  veces la información de un suceso cualquiera  $A_{ik}$ , por lo que la igualdad nos queda de esta manera:  $I(A_{i1}, A_{i2}, \dots, A_{im}) = n^m \rightarrow mI(A_{ik}) = mn$

Una función que permite realizar la transformación de la premisa 4, es el logaritmo neperiano. En consideración de lo expuesto, se define la información de un suceso en la siguiente ecuación:

$$I(A_k) = \log\left(\frac{1}{p_k}\right) = -\log(p_k) \quad (4.1)$$

A manera de resumen, podemos observar en la [Tabla 1](#), la relación información - incertidumbre, recogida por [62].

#### 4.1.1 Entropía

La entropía definida por Shannon dentro del contexto de la información y comunicación, indica la cantidad media de información que contiene una variable aleatoria o una fuente de transmisión binaria. Este concepto se puede expresar a través de la [Ecuación 4.2](#).

$$H(S) = - \sum_{i=1}^n p_i \times \log(p_i) \quad (4.2)$$

| Probabilidad | Información<br>previa<br>relevante | Información<br>obtenida | Sorpresa | Reducción de<br>incertidumbre | Capacidad de<br>predicción<br>del evento |
|--------------|------------------------------------|-------------------------|----------|-------------------------------|------------------------------------------|
| 1            | toda                               | ninguna                 | ninguna  | ninguna                       | completa                                 |
| alta         | mucha                              | baja                    | poca     | poca                          | alta                                     |
| media        | media                              | media                   | neutra   | media                         | media                                    |
| baja         | baja                               | alta                    | mucha    | alta                          | baja                                     |
| 0            | ninguna                            | muy alta                | muy alta |                               | nula                                     |

Tabla 1: Relación información e incertidumbre.

A continuación listamos las propiedades de la entropía de un sistema  $(H(S))$ , anotadas por [62].

1. Es simétrica respecto a las permutaciones de las probabilidades de los sucesos  $(p_i)$ .
2. Es invariante a traslaciones y escalamientos; para todo  $a \neq 0$  y todo  $b$ :  $H(aX + b) = H(X)$
3. Es una función continua respecto a las probabilidades de los sucesos  $(p_i)$ .
4. Es una función cóncava a las probabilidades de los sucesos  $(p_i)$ .
5. Es no negativa.
6. El valor máximo se alcanza para  $u = (\frac{1}{n}, \dots, \frac{1}{n})$  (distribución uniforme), por lo que  $0 \leq H(X) \leq \log(n)$
7. Dadas dos variables aleatorias  $X, Y$  se tiene que  $H(X, Y) \leq H(X) + H(Y)$ , donde  $(X, Y)$  es la distribución conjunta de  $X$  y  $Y$ . La igualdad se alcanza solo cuando  $X$  y  $Y$  son independientes.
8. Cualquier cambio que tienda a igualar las probabilidades  $(p_i)$  incrementa  $H$ .

#### 4.1.2 Máxima Entropía

Es importante tomar en consideración que muchos problemas del PLN pueden ser reformulados como problemas estadísticos de clasificación, como nos explica [61]. El contexto dentro del que se modelan las soluciones estadísticas debe tomar en cuenta las siguientes consideraciones:

- Se debe calcular la probabilidad de que la clase  $a$  ocurra en el contexto  $b$ , esto es,  $p(a, b)$ .
- El contexto  $b$  puede consistir en una o más palabras asociadas a etiquetas sintácticas o semánticas.
- Los corpora de gran tamaño pueden contener información sobre la coocurrencia de  $a$  y  $b$ , pero no la gama completa de valores de probabilidad  $p(a, b)$  para todos los posibles pares  $(a, b)$  y más aún, si los contextos  $b$  son escasos.

- Se debe contar con un método que use la escasa evidencia de  $a$  y  $b$  para modelar la probabilidad  $p(a, b)$  de la mejor forma posible.

En virtud de lo expuesto, consideramos de gran interés citar las palabras de [35], que analiza las ventajas de este principio:

*“... en el proceso de inferencia a partir de información parcial, debemos usar la distribución de probabilidad que posea la máxima entropía para todo aquello que es desconocido. Esta es la asignación más imparcial que podemos llevar a cabo....”*

En virtud de esto, realizaremos las siguientes asunciones:  $A$  especifica el conjunto de posibles clases y  $B$  denota el conjunto de contextos posibles, con ello  $p$  debería maximizar la entropía:

$$H(p) = - \sum_{x \in E} p(x) \times \log(p(x)) \quad (4.3)$$

Donde:  $x = (a, b)$ ,  $a \in A$ ,  $b \in B$  y  $E = A \times B$ . Es importante recalcar que esto debería ser consistente con la información o evidencia parcial que se tiene.

El siguiente aspecto a considerar, es que se debe representar de forma adecuada la información, ya que esto nos permitirá codificar los hechos como características e imponer restricciones a los valores de dichas características. Una característica es una función valuada binaria  $f_j : E \rightarrow \{0, 1\}$ . Si tenemos  $k$  características, las restricciones tienen la siguiente forma:

$$E_p f_j = E_{\tilde{p}} f_j \quad (4.4)$$

donde  $1 \leq j \leq k$ .  $E_p f_j$  es el modelo de la probabilidad esperada de  $f_j$ :  $E_p f_j = \sum_{x \in E} p(x) f_j(x)$  y está restringida a coincidir con la observación esperada,  $E_{\tilde{p}} f_j = \sum_{x \in E} \tilde{p}(x) f_j(x)$ . En este caso no se debe pasar por alto que  $\tilde{p}$  es la probabilidad observada de  $x$  en algún ejemplo de entrenamiento  $S$ . Un modelo de probabilidad  $p$  es consistente con la evidencia observada sí y sólo sí observó las restricciones indicadas en la Ecuación 4.4. El principio de máxima entropía recomienda usar  $p^*$ :

$$P = \{p \mid E_p f_j = E_{\tilde{p}} f_j, j = \{1 \dots k\}\}$$

$$p^* = \arg \max_{p \in P} H(p)$$

## MODELOS BASADOS EN CAMPOS ALEATORIOS CONDICIONALES

### 5.1 CAMPOS ALEATORIOS CONDICIONALES

Los Campos Aleatorios Condicionales (CAC) plantean un modelo probabilístico similar al que se usa en el principio de ME para realizar las tareas de etiquetado de secuencias de palabras. Empero, algo que lo diferencia del principio de ME, es que no presenta las dificultades en el caso de representar la interacción de múltiples características o largas secuencias de dependencia de las observaciones [41]. De igual forma, no debemos olvidar que su aplicación tiene un amplio campo de acción, verbigracia, bioinformática, lingüística computacional y reconocimiento del habla, entre otros [78].

#### 5.1.1 El Problema de la Etiqueta Bias

[41] plantea que los modelos basados en ME, Modelos Ocultos de Markov, Modelos Ocultos de Markov basados en ME y otros modelos discriminativos de Markov, comparten una gran debilidad: *el problema de la etiqueta bias*. Para entender el problema de la etiqueta bias, vamos a suponer que tenemos un autómata finito simple, que debe distinguir entre dos posibles palabras: *rib* y *rob* (Figura 4).

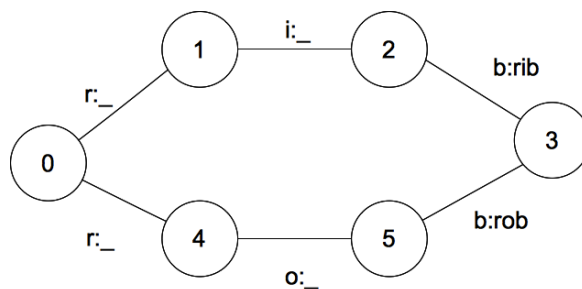


Figura 4: Autómata ejemplo, problema de la etiqueta bias [41].

Cuando llegue la primera letra (*r*), existen dos posibles caminos a seguir a través de los estados 1 y 4, por lo que la distribución de probabilidad se reparte de forma equitativa entre estos dos estados. A continuación se observa la letra *i*. El estado 1 habrá observado la ocurrencia de *i* durante el entrenamiento, mientras que es muy probable que el estado 4 no la haya visto nunca; sin embargo, el estado 4 (al igual que el estado 1) no tiene otra alternativa de transición y al pasar toda la distribución de probabilidad a su única salida, no genera una observación, únicamente está condicionando una. Tomando en consideración este caso, se puede notar que la ruta de transiciones a través de los estados 0-1-2-3 y la ruta 0-4-5-3 son independientes de la secuencia de observación. De igual forma, si alguna de las dos palabras aparece con mayor frecuencia durante el entrenamiento, las transiciones de salida del estado inicial (0) preferirán este camino y dicha palabra siempre ganará.

En la tesis doctoral de [11] se plantean dos soluciones al problema de la etiqueta bias. La primera sugiere cambiar la estructura de transiciones que posee el autómata, verbigracia, colapsando los estados 1 y 4 del ejemplo anterior. La segunda alternativa propone utilizar un modelo totalmente interconectado, de modo que el proceso de entrenamiento pueda inferir una buena estructura.

### 5.1.2 Campos Aleatorios Condicionales

Un campo aleatorio condicional es una distribución de probabilidad  $p(x|y)$  con estructura gráfica asociada [72]. En virtud de que el modelo es condicional, las dependencias en las variables de entrada ( $x$ ), no necesitan ser representadas de forma explícita. [41] nos presenta una definición muy interesante y compacta de los CAC:

Asumimos que  $G = (V, E)$  es un grafo tal que  $Y = (Y_v)_{v \in V}$ , de modo que  $Y$  se indexa a través de los vértices de  $G$ .  $(X, Y)$  será un *campo aleatorio condicional*, en caso de que cuando se condicione a  $X$ , las variables aleatorias  $Y_v$  cumplan la propiedad de Markov con respecto al grafo:  $p(Y_v | X, Y_w, w \sim v)$ , donde  $w \sim v$  significa que  $w$  y  $v$  son vecinos en  $G$ .

Con esto, un CAC está condicionado de modo global a una observación  $X$ . La forma más simple para modelar secuencias, es en la que  $G$  es una simple cadena o línea:  $G = (V = \{1, 2, 3 \dots m\}, E = \{(i, i + 1)\})$ .  $X$  debe tener una estructura natural de grafo. Como nos explica [41], no es necesario asumir que  $X$  y  $Y$  tienen la misma estructura. Por el teorema fundamental de los campos aleatorios [33], la distribución conjunta sobre la secuencia de etiquetas  $Y$  dada  $X$  tiene la siguiente forma:

$$p_\theta(y|x) \propto \exp \left( \sum_{e \in E, k} \lambda_k f_k(e, y|e, x) + \sum_{v \in V, k} \mu_k g_k(v, y|v, x) \right) \quad (5.1)$$

donde  $x$  es una secuencia de datos,  $y$  una secuencia de etiquetas,  $y|_S$  es el conjunto de componentes de  $y$  asociados con los vértices en el subgrafo  $S$ .

Como aspecto final de esta sección, es importante mencionar que el problema de estimación es determinar los parámetros  $\theta = (\lambda_1, \lambda_2, \dots; \mu_1, \mu_2, \dots)$  de los datos de entrenamiento  $D = \left\{ (x^{(i)}, y^{(i)}) \right\}_{i=1}^N$ , tomando en cuenta la distribución empírica  $\tilde{p}(x, y)$ .



## MODELOS BASADOS EN MAQUINAS DE SOPORTE VECTORIAL

### 6.1 MAQUINAS DE SOPORTE VECTORIAL

Las Máquinas de Soporte Vectorial (MSV) son una técnica que ha sido desarrollada por el científico ruso Vladímir Vapnik y su equipo de colaboradores, en los laboratorios AT&T [9]. Las MSV realizan un mapeo de los puntos de entrada a un espacio de características de una dimensión mayor, verbigracia, si los puntos están en  $\mathbf{R}^2$ , entonces la MSV los mapea a  $\mathbf{R}^3$  y encuentra un hiperplano que los separe y maximice el margen  $m$  entre las clases en este espacio [8]. En la Figura 5 se puede apreciar un ejemplo de lo expuesto.

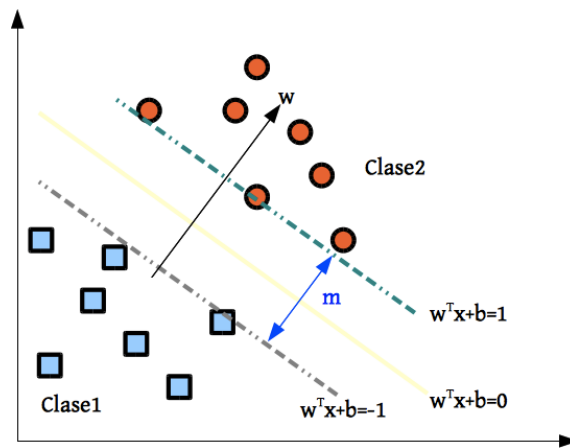


Figura 5: Frontera de decisión  $m$  en las MSV.

Maximizar el margen  $m$  es un problema de programación cuadrática<sup>1</sup> y se puede solucionar a través de la introducción de multiplicadores de Lagrange. La MSV es capaz de encontrar el hiperplano óptimo usando funciones de producto punto con funciones de espacio de características (*kernels*).

Consideramos el problema de separar un conjunto de vectores de entrenamiento  $(x_1, y_1), \dots, (x_l, y_l), \in \mathbf{R}^N$  que pertenecen a dos clases  $y_i = \{-1, +1\}$ . El objetivo que se busca alcanzar en este problema, es que se pueda separar las clases a través de un hiperplano [25]:

$$(\mathbf{w} \cdot \mathbf{x}) + b = 0, \mathbf{w} \in \mathbf{R}^N, b \in \mathbf{R}$$

donde  $\mathbf{w}$  y  $b$  son parámetros que se inducen de los ejemplos que corresponden a las funciones de decisión  $f(\mathbf{x}) = \text{sign}(\mathbf{w} \cdot \mathbf{x} + b)$ , de forma que puedan generalizar de forma adecuada los ejemplos no vistos [8]. Al realizar la clasificación de los vectores, existen dos casos posibles:

<sup>1</sup> La programación cuadrática es un caso de la programación matemática no lineal. Dentro de este caso, las restricciones son lineales, sin embargo, el objetivo es cuadrático.

- Caso linealmente separable . - Como se aprecia en la [Figura 6](#), caso *a*, tenemos un conjunto de vectores de entrenamiento, que se desea separar en dos clases  $y_i \in \{-1, 1\}$ . A fin de simplificar el ejemplo, asumiremos que cada vector  $x_1, x_2, \dots, x_l$  es bidimensional, es decir, representa un punto. Para la mayoría de los casos, encontrar un hiperplano adecuado en un espacio de entrada, suele ser una tarea restrictiva en el uso práctico. En virtud de ello, [25] nos explica que el hiperplano que separa de forma óptima las dos clases es aquel que minice el funcional  $\Phi(\mathbf{w}) = \frac{\|\mathbf{w}\|^2}{2}$ . Por ello, el problema de optimización puede ser reformulado en un problema equivalente no limitado, usando multiplicadores de Lagrange [49]:

$$L(\mathbf{w}, b, \alpha) = \frac{\|\mathbf{w}\|^2}{2} - \sum_{i=1}^l \alpha_i \{[(\mathbf{w} \cdot \mathbf{x}_i) + b] y_i - 1\} \quad (6.1)$$

donde  $\alpha_i$  son los multiplicadores de Lagrange. La [Ecuación 6.1](#), debe ser minimizada con respecto a  $\mathbf{w}$  y  $b$ :

$$\frac{dL}{db} = 0 \rightarrow \sum_{i=1}^l \alpha_i y_i = 0 \quad y$$

$$\frac{dL}{d\mathbf{w}} = 0 \rightarrow \mathbf{w}_0 = \sum_{i=1}^l y_i \alpha_i \mathbf{x}_i = 0$$

Si reemplazamos  $w_0$  en la [Ecuación 6.1](#), obtenemos un resultado en la forma dual de la función y ahora se deberá maximizar dicha función con respecto a las restricciones  $\alpha_i \geq 0$ :

$$W(\alpha) = \sum_{i=1}^l \alpha_i - \left(\frac{1}{2}\right) \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j y_i y_j (\mathbf{x}_i \cdot \mathbf{x}_j) \quad (6.2)$$

A través de programación cuadrática se ha podido encontrar una solución a la [Ecuación 6.2](#) en la forma de un vector  $\alpha^0 = (\alpha^1, \alpha^2, \dots, \alpha^l)$ , por lo tanto el hiperplano óptimo de separación está dado por:

$$\mathbf{w}_0 = \sum_{\text{vectores-soporte}} y_i \alpha_i^0 \mathbf{x}_i = 0 \quad y \quad b_0 = -\frac{1}{2} \mathbf{w}_0 \cdot [\mathbf{x}_r + \mathbf{x}_s]$$

donde  $\mathbf{x}_r$  y  $\mathbf{x}_s$  son los vectores de soporte para cada clase. El clasificador puede ser construido de la siguiente forma:

$$f(\mathbf{x}) = \text{sign}(\mathbf{w}_0 \cdot \mathbf{x} + b_0) = \text{sign} \left( \sum_{\text{vector-soporte}} y_i \alpha_i^0 (\mathbf{x}_i \cdot \mathbf{x}) + b_0 \right) \quad (6.3)$$

Únicamente los puntos  $\mathbf{x}_i$  que posean multiplicadores de Lagrange  $\alpha_i^0$  distintos de cero serán considerados *Vectores de Soporte*.

- Caso no linealmente separable . - Para esta situación, ([Figura 6](#), caso *b*) se debe permitir la introducción de violaciones a la clasificación en la formulación de la MSV. Es posible tratar con datos no linealmente separables, tomando como base el análisis anterior y

agregando variables no negativas  $\xi_i \geq 0$ . Estas variables miden la proporción en que se viola las restricciones (Figura 6, caso c). Por ello, cuando el margen  $m$  es maximizado, se debe pagar una “penalización” equivalente a la proporción en que se violan las restricciones [25]:

$$\text{Minimizar} \Rightarrow \Phi(\mathbf{w}) = \frac{\|\mathbf{w}\|^2}{2} + C \left( \sum \xi_i \right) \quad (6.4)$$

sujeto a  $\Rightarrow y_i [(\mathbf{w} \cdot \mathbf{x}_i)] \geq 1 - \xi_i$  y  $\xi_i \geq 0$   $i = 1, 2, \dots, l$   
 donde  $C$  es un parámetro escogido a priori que define el coste de violar las restricciones. El primer término de la Ecuación 6.4 provee una minimización para la dimensión VC (Vapnik-Chervonenkis) de la función de aprendizaje. Si se minimiza el segundo término, se controla el Riesgo Empírico<sup>2</sup>. Esta aproximación constituye una aplicación práctica de la Minimización de Riesgo Estructural<sup>3</sup>. Para solucionar este problema, se aplican nuevamente operadores de Lagrange:

$$L(\mathbf{w}, b, \alpha) = \frac{\|\mathbf{w}\|^2}{2} + C \left( \sum_{i=1}^l \xi_i \right) - \sum_{i=1}^l \alpha_i \{[(\mathbf{w} \cdot \mathbf{x}_i) + b] y_i - 1 + \xi_i\} - \sum_{i=1}^l \gamma_i \xi_i \quad (6.5)$$

donde  $\alpha_i$  y  $\gamma_i$  están asociados con las restricciones de la Ecuación 6.5. Los valores de  $\alpha_i$  deben estar limitados de la siguiente forma:  $0 \leq \alpha_i \leq C$ . Al igual que en el caso del conjunto de datos linealmente separables, este problema se puede solucionar a través de multiplicadores de Lagrange.

En caso de que una frontera lineal sea inapropiada o que la superficie de decisión no sea lineal, la MSV puede mapear el vector de entrada  $\mathbf{x}$  a un espacio mayor de características  $\mathbf{z}$ , tomando mapeo no-lineal a priori. Con esto, la MSV construye un hiperplano óptimo de separación en este espacio de mayor dimensión:

$$W(\alpha) = \sum_{i=1}^l \alpha_i - \left( \frac{1}{2} \right) \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j y_i y_j \mathbf{K}(\mathbf{x}_i \cdot \mathbf{x}_j) \quad (6.6)$$

donde  $\mathbf{K}(\mathbf{x}_i \cdot \mathbf{x}_j)$  es la función *kernel* que realiza el mapeo no lineal al espacio de características, de modo que las restricciones no se vean afectadas. Resolviendo la Ecuación 6.6 tenemos el

<sup>2</sup> El principio inductivo de Minimización del Riesgo Empírico es el más comúnmente utilizado en los procesos de aprendizaje clásico. Fija su atención en el siguiente principio: “Para obtener una buena generalización es suficiente con elegir los parámetros de la función aproximadora que aseguren el número mínimo de errores sobre el conjunto de entrenamiento” [4].

<sup>3</sup> Es un proceso de inferencia desarrollado sobre la Teoría de Aprendizaje Estadístico, específicamente para trabajar con problemas de aprendizaje a partir de un conjunto de entrenamiento pequeño [4].

clasificador que implementa el hiperplano óptimo en el espacio de características:

$$f(\mathbf{x}) = \text{sign} \left( \sum_{i=1}^l y_i \alpha^0 \mathbf{K}(\mathbf{x}_i \cdot \mathbf{x}) + b_0 \right) \quad (6.7)$$

De acuerdo al *kernel*  $\mathbf{K}$  que se use, se puede construir una gran variedad de funciones de aprendizaje. Algunas de las funciones que proveen mejores resultados son las de tipo polinomial, base radial y *sigmodal*:

- Función *kernel* de polinomial simple:  $K(\mathbf{x}, \mathbf{x}_i) = ((\mathbf{x}, \mathbf{x}_i) + 1)^d$ , donde el grado del polinomio  $d$  es definido por el usuario.
- Función *kernel* de base radial:  $K(\mathbf{x}, \mathbf{x}_i) = \exp(-\gamma |\mathbf{x}, \mathbf{x}_i|^2)$ , donde  $\gamma$  es definido por el usuario.
- Función *kernel* de red neuronal:  $K(\mathbf{x}, \mathbf{x}_i) = \tanh(b(\mathbf{x} \cdot \mathbf{x}_i) - c)$ , donde  $b$  y  $c$  son definidos por el usuario.

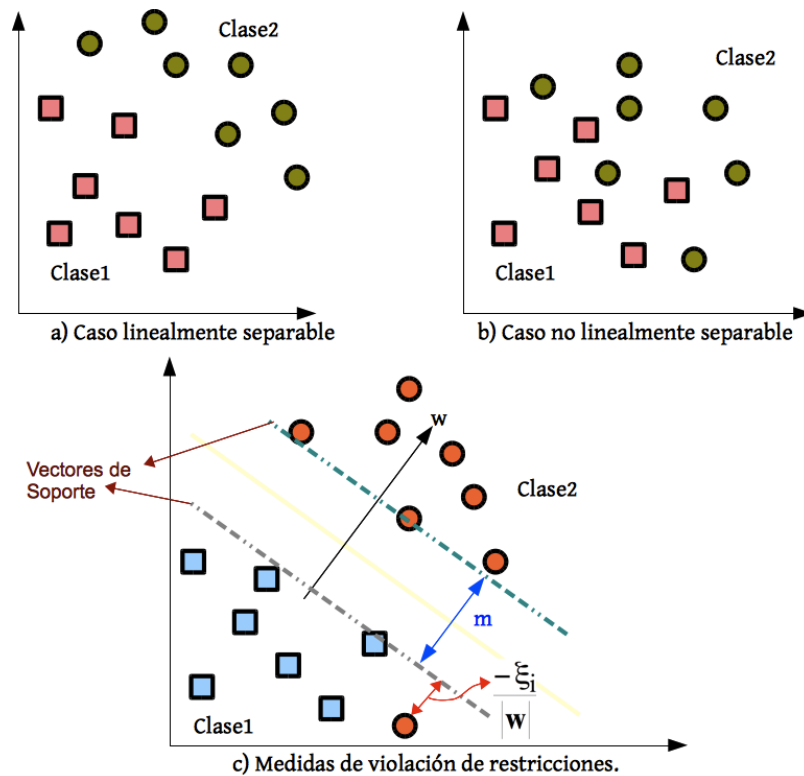


Figura 6: Casos lineal y no linealmente separables. Medidas de violación de restricciones  $\xi_i$ .

Es importante tomar en consideración que se puede dar el caso de que se tenga más de una clase para realizar el proceso discriminación. Si deseamos clasificar  $k$ -clases, debemos construir un conjunto de clasificadores binarios  $f^1, \dots, f^k$  y entrenarlos por separado, es decir, una clase separada del resto. Una vez que se ha realizado el entrenamiento, se deben combinar los clasificadores, aplicando un esquema de votación [66].

AdaBoost es un clasificador estadístico que combina un conjunto de clasificadores débiles ( $h_t$ ) para formar un clasificador fuerte ( $H(x)$ ), que entrega el resultado final del algoritmo. A fin de tener una idea del funcionamiento del algoritmo, se plantea un modelo de asignación en línea. El agente de asignación  $A$  tiene  $N$  estrategias u opciones para escoger, que se numeran con valores enteros:  $1 \dots N$ . En cada paso temporal  $t = 1, 2, \dots, T$ , el agente  $A$  define una distribución  $\mathbf{p}^t$  para las estrategias. La distribución asignada a la estrategia  $i$  debe ser mayor o igual que cero ( $\mathbf{p}_i^t \geq 0$ ) y si sumamos todas las distribuciones se debe tener 1:  $\sum_{i=1}^N \mathbf{p}_i^t = 1$ . Cada estrategia  $i$  tiene una pérdida  $l_i^t$  que esta determinada por el posible ambiente adverso. La pérdida sufrida por el agente  $A$  es  $\sum_{i=1}^N \mathbf{p}_i^t \cdot l_i^t = \mathbf{p}_i^t \cdot \mathbf{I}_i^t$ , por ejemplo, la pérdida promedio de las estrategias respecto a la regla de asignación elegida por  $A$ . Esta pérdida se define como pérdida por mezcla [26].

El objetivo del algoritmo  $A$  es minimizar su pérdida acumulativa, de modo que se pueda tener un valor de pérdida igual que la mejor estrategia:

$$L_A = \min_i L_i$$

donde:

$$L_A = \sum_{t=1}^T \mathbf{p}_i^t \cdot \mathbf{I}_i^t$$

es la pérdida acumulativa sufrida por el algoritmo  $A$  en los  $T$  primeros intentos, y

$$L_i = \sum_{t=1}^T \mathbf{I}_i^t$$

es la pérdida acumulativa de la estrategia  $i$ .

En la [Figura 7](#) podemos observar un ejemplo del algoritmo AdaBoost para tareas de clasificación binaria [81]. Como se aprecia, los clasificadores débiles son llamados de forma repetida en rondas de  $t = 1, 2, \dots, T$  (paso temporal). Para cada ejecución una distribución de pesos  $D_t$  es actualizada. En cada ronda los pesos de cada ejemplo mal clasificados son incrementados y los pesos de los ejemplos bien clasificados son decrementados, de modo que el nuevo clasificador se enfoca más en estos ejemplos.

Dado  $(x_1, y_1), \dots, (x_m, y_m)$  , donde  $x_i \in X, y_i \in Y = \{-1, +1\}$

Inicialiar  $D_1(i) = \frac{1}{m}, i = 1, \dots, m$

Desde  $t = 1, \dots, T$

- Buscar el clasificador  $h_t: X \rightarrow \{-1, +1\}$  que minimiza el error respecto a la distribución  $D_t$  :

$$h_t = \underset{h_j \in H}{\operatorname{argmin}}(\epsilon_j) \text{ , donde } \epsilon_j = \sum_{i=1}^m D_t(i)[y_i \neq h_j(x_i)]$$

- Si  $\epsilon_t > 0.5$  entonces parar.
- Escoger  $\alpha_t \in R$  , típicamente  $\alpha_t = \frac{1}{2} \ln\left(\frac{1-\epsilon_t}{\epsilon_t}\right)$  , donde  $\epsilon_t$  es la tasa de error ponderado del clasificador  $h_t$
- Actualizar:

$$D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t} \text{ , donde } Z_t \text{ es un factor de}$$

normalización (escogido para que  $D_{t+1}$  sea una distribución de probabilidad).

Salida del clasificador final:

$$H(x) = \operatorname{sign}\left(\sum_{t=1}^T \alpha_t h_t(x)\right)$$

La ecuación que permite actualizar  $D_t$  debe ser construida de modo que:

$$-\alpha_t y_i h_t(x_i) = \begin{cases} < 0, y(i) = h_t(x_i) \\ > 0, y(i) \neq h_t(x_i) \end{cases}$$

Figura 7: Ejemplo en pseudocódigo del algoritmo AdaBoost para tareas de clasificación binarias. Se puede apreciar que después de seleccionar el clasificador óptimo  $h_t$  para la distribución  $D_t$ , los ejemplos  $x_i$  que el clasificador  $h_t$  identificó correctamente se pesan con un menor valor. En el caso contrario, a los ejemplos que se han identificado incorrectamente se les asigna un peso mayor.

## Parte II

# ETIQUETADO DE ROLES SEMÁNTICOS





En este capítulo revisamos los objetivos y procedimientos de la *shared task* "Etiquetado de Roles Semánticos" (Semantic Role Labeling) CoNLL 2005. De igual forma, se ven de forma concreta las características más relevantes del corpus empleado, la metodología de etiquetado empleada usando los modelos expuestos en el [Capítulo 4](#) y el esquema de funcionamiento de los 5 mejores sistemas de dicha competición.

Antes de pasar a revisar en detalle los lineamientos de la competición CoNLL 2005, es importante tomar en consideración que se han realizado varias competiciones que involucran la tarea de etiquetado de roles semánticos, entre las más importantes podemos citar:

- CoNLL 2004 . - Plateaba como objetivo desarrollar un sistema de aprendizaje de máquina que reconozca los participantes de las proposiciones. Se consideraba como argumentos a todos los tipos de participantes en las proposiciones, incluyendo adjuntos, referencias y las realizaciones de los verbos.
- CoNLL 2009 . - Su objetivo principal era extraer las dependencias sintácticas y semánticas en múltiples lenguajes (inglés, catalán, chino, checo, alemán, japonés y español).
- SemEval 2010 . - Se proponía realizar una subtarea relacionadas con el etiquetado de roles semánticos: búsqueda de argumentos semánticos del predicado (reconocimiento de roles) y anotación con el rol correcto.

### 8.1 COMPETICIÓN CONLL 2005: ETIQUETADO DE ROLES SEMÁNTICOS

El objetivo de la competición fue evaluar sistemas de aprendizaje automático, que sean capaces de reconocer los participantes de las proposiciones de acuerdo al verbo que define a cada una de las mismas. Las anotaciones que se proveen como entrada de los sistemas consisten en información sintáctica y entidades nombradas. De acuerdo a competiciones anteriores, la información sintáctica está constituida por etiquetas de tipo POS, anotación de constituyentes (*chunks*) y cláusulas. Adicionalmente, se provee el análisis sintáctico completo.

Los conjuntos de datos de entrenamiento y desarrollo se proporcionan para construir los sistemas de aprendizaje, ya que poseen las anotaciones estimadas y las salidas correctas. El corpus de entrenamiento se utiliza para preparar al sistema, mientras que el de desarrollo se emplea para ajustar los parámetros de funcionamiento.

La evaluación del sistema se realiza a través del conjunto de pruebas, que contiene los verbos que deben analizarse, así como los argumentos anotados. Cada sistema se evalúa de acuerdo a tres parámetros: precisión, *recall* y *F-Measure* de los argumentos correctamente reconocidos. Para que un argumento sea reconocido de forma correcta, su tipo, así como todas las palabras que lo contienen, deben estar correctamente identificadas.

Los sistemas de preprocesado que se disponían para la competición son los siguientes:

- Procesadores UPC:
  - Etiquetador POS [45]
  - Etiquetador de componentes léxicos e identificador de constituyentes [17]
- Analizador Collins:
  - Analizador de [23], con modelo 2.
  - Analizador total del WSJ, con información léxica de cada constituyente sintáctico.
- Analizador Charniak:
  - Analizador completo de [19].
  - Etiquetador POS y analizador total del WSJ
- Extractor de Entidades Nombradas:
  - Extractor de entidades nombradas para el idioma inglés, desarrollado por [20].

## 8.2 CORPUS CONLL 2005

Este corpus ha servido de base para entrenar y probar los sistemas descritos en el [Capítulo 4](#), y de igual forma, ha permitido llevar a cabo la experimentación del presente trabajo. Entre los aspectos más importantes que posee el mencionado corpus, podemos destacar que está basado en algunas secciones del periódico *Wall Street Journal* (WSJ) y que se divide en tres partes: entrenamiento (secciones 02 a la 21), desarrollo (sección 24) y pruebas (*tWSJ* y *tBrown*, sección 23). Otra característica de interés, es que sigue el esquema de anotación Prop Bank 1.0, propuesto por [56] y posee las características que se apuntan a continuación[18]:

- Corpus WSJ, parte del Penn TreeBank, enriquecido con estructuras predicado-argumento.
- Tipos de argumentos:
  - Argumentos numerados (A0-A5, AA) . - Roles específicos de acuerdo a los verbos. Su semántica depende del verbo.
  - Adjuntos (AM-) . - Causa, dirección, tiempo, ubicación, manera, denegación, etc.
  - Referencias (R-)
  - Verbos (V)

Como se puede observar, en la [Tabla 2](#) describimos las características puntuales del corpus, esto es, número de sentencias, componentes léxicos (*tokens*), proposiciones, verbos y argumentos.

A fin de tener una idea más clara del formato que se ha seguido para etiquetar el corpus, así como la estructura que poseen los ficheros de datos, tomaremos la oración: “*Most of the stock selling pressure came from Wall Street professionals, including computer-guided program traders.*”, del conjunto de datos *tWSJ*. En la [Tabla 3](#) podemos observar que se

|               | Entrenamiento | Desarrollo | tWSJ  | tBrown |
|---------------|---------------|------------|-------|--------|
| Oraciones     | 39832         | 1346       | 2416  | 426    |
| Palabras      | 950028        | 32853      | 56684 | 7159   |
| Proposiciones | 90750         | 3248       | 5267  | 804    |
| Verbos        | 3101          | 860        | 982   | 351    |
| Argumentos    | 239858        | 8346       | 14077 | 2177   |

Tabla 2: Características del corpus CoNLL 2005 [18].

tienen tres columnas, donde la primera especifica cada constituyente de la oración, la segunda los argumentos del verbo *came* y la tercera los argumentos del verbo *including*. Para el primer caso (verbo *came*), el constituyente “*Most of the stock selling pressure*” ha sido marcado como argumento A1 y “*from Wall Street professionals, including computer-guided program traders*” se ha etiquetado como A2<sup>1</sup>. Para el segundo caso (verbo *including*), tenemos que el constituyente “*professionals*” se ha marcado como A2 y el constituyente “*computer-guided program traders*” como A1. Los roles A1 y A2 representan el objeto aceptado y la fuente de la que se acepta algo, respectivamente.

| Constituyente    | Roles del Verbo <i>came</i> | Roles del Verbo <i>including</i> |
|------------------|-----------------------------|----------------------------------|
| Most             | (A1*                        | *                                |
| of               | *                           | *                                |
| the              | *                           | *                                |
| stock            | *                           | *                                |
| selling          | *                           | *                                |
| pressure         | *)                          | *                                |
| <b>came</b>      | (V*)                        | *                                |
| from             | (A2*                        | *                                |
| Wall             | *                           | *                                |
| Street           | *                           | *                                |
| professionals    | *                           | (A2*)                            |
| ,                | *                           | *                                |
| <b>including</b> | *                           | (V*)                             |
| Computer-guided  | *                           | (A1*                             |
| program          | *                           | *                                |
| traders          | *)                          | *)                               |
| .                | *                           | *                                |

Tabla 3: Ejemplo del formato de almacenamiento y etiquetado de un oración del corpus tWSJ.

Finalmente, en la [Figura 8](#), tenemos un ejemplo completo de una oración anotada. Las anotaciones realizadas se agrupan de la siguiente forma:

- Palabras (*Words*) . - Palabras que forman la oración.

<sup>1</sup> A0 y A1 corresponden a argumentos válidos para el verbo *came* en el esquema de anotación de PropBank.

- NE . - Entidades nombradas.
- POS . - Análisis de constituyentes.
- Sintáxis Parcial (*Partial Synt*) . - Sintáxis parcial, anotación de los constituyentes (primera columna) y cláusulas (segunda columna).
- Sintáxis Completa (*Full Synt*) . - Arbol sintáctico completo.
- VS (*Verb Sense*). - Sentido de los verbos de acuerdo a VerbNet.
- Objetivos (*Targets*) . - Son los verbos que deben ser analizados en cada oración. Su modo es infinitivo.
- Propositiones (*Props*) . - Se tiene una columna con los argumentos para cada verbo objetivo.

| WORDS----> | NE----> | POS   | PARTIAL_SYNT | FULL_SYNT-----> | VS        | TARGETS | PROPS-----> |
|------------|---------|-------|--------------|-----------------|-----------|---------|-------------|
| The        | *       | DT    | (NP*         | (S*             | (S(NP*    | -       | (A0*        |
| \$         | *       | \$    | *            | *               | (ADJP(OP* | -       | *           |
| 1.4        | *       | CD    | *            | *               | *         | -       | *           |
| billion    | *       | CD    | *            | *               | *)        | -       | *           |
| robot      | *       | NN    | *            | *               | *         | -       | *           |
| spacecraft | *       | NN    | *)           | *)              | *)        | -       | *)          |
| faces      | *       | VBZ   | (VP*         | *               | (VP*      | 01      | face (V*    |
| a          | *       | DT    | (NP*         | *               | (NP*      | -       | (A1*        |
| six-year   | *       | JJ    | *            | *               | *         | -       | *           |
| journey    | *       | NN    | *)           | *)              | *         | -       | *           |
| to         | *       | TO    | (VP*         | (S*             | (S(VP*    | -       | *           |
| explore    | *       | VB    | *)           | *)              | (VP*      | 01      | explore (V* |
| Jupiter    | (ORG*)  | NNP   | (NP*         | *               | (NP(NP*   | -       | *) (A1*     |
| and        | *       | CC    | *            | *               | *         | -       | *           |
| its        | *       | PRP\$ | (NP*         | *               | (NP*      | -       | *           |
| 16         | *       | CD    | *            | *               | *         | -       | *           |
| known      | *       | JJ    | *            | *               | *         | -       | *           |
| moons      | *       | NNS   | *)           | *)              | *)        | -       | *)          |
| .          | *       | .     | *            | *)              | *)        | -       | *           |

Figura 8: Ejemplo de una oración anotada del corpus WSJ.

### 8.3 ETIQUETADO DE ROLES SEMÁNTICOS

En esta sección se detalla la metodología llevada a cabo para realizar el etiquetado de roles semánticos, desde los principales enfoques que se han descrito en el [Capítulo 4](#): máxima entropía, campos aleatorios condicionales y máquinas de soporte vectorial.

#### 8.3.1 Etiquetado basado en modelos de ME

El etiquetado de roles semánticos a través de modelos basados en ME se ha aplicado de forma exitosa en diversas competiciones de PLN. En esta sección se revisará de forma breve las soluciones implementadas por [42] y [43], en el marco de las competiciones organizadas por CoNLL, realizadas en idioma inglés, en los años 2004 y 2005, respectivamente. Según la definición de [7], dentro del contexto de la ME, la probabilidad condicional de predecir una salida  $y$  dada una historia  $x$ , se define como sigue (forma paramétrica de la ME):

$$P(x | y) = \frac{1}{Z(x)} \exp \left( \sum_{i=1}^k \lambda_i f_i(x, y) \right)$$

donde  $f_i(x, y)$  es la función de características,  $\lambda_i$  es parámetro de peso de  $f_i(x, y)$ ,  $k$  es el número de características y  $Z(x)$  es una constante de normalización para que se cumpla la igualdad  $\sum_y p(y | x) = 1$ .

Dado un predicado y su árbol de análisis sintáctico parcial representado por constituyentes como las etiquetas del sintagma y las cláusulas, el modelo probabilístico para el etiquetado de roles semánticos asigna los roles de acuerdo a lo que se describe en la [Ecuación 8.1](#) por [42]:

$$R_{best} = \arg \max_R P(R | c_{1n}, pred) = \arg \max_R \prod_{i=1}^n P(r_i | c_{1n}, pred, r_{1...i-1}) \quad (8.1)$$

donde,  $R$  es una secuencia de roles semánticos,  $c_{1n}$  es una secuencia de constituyentes,  $pred$  es el predicado dado,  $r_i$  es el  $i$ -ésimo rol semántico y  $n$  es el número de constituyentes.

[42] plantea un esquema basado en una aproximación incremental y clasifica dos cláusulas: una inmediata y una superior. La cláusula inmediata contiene el predicado señalado como objetivo, mientras que la cláusula superior contiene a la inmediata. Generalmente, la mayoría de los argumentos del predicado se encuentran en la cláusula inmediata, mientras que algunos de los restantes se ubican en la superior. Para aplicar el enfoque incremental, [42] implementa la [Ecuación 8.2](#) :

$$\begin{aligned} R_{best} &= \arg \max_R \prod_{i=1}^n P(r_i | c_{1n}, pred, r_{1...i-1}) \\ &\approx \arg \max_R \prod_{i=1}^m P(r_i | \Phi_1(c_{1n}, pred, r_{1...i-1})) \times \prod_{i=m+1}^n P(r_i | \Phi_2(c_{1n}, pred, r_{1...i-1})) \end{aligned} \quad (8.2)$$

donde  $m$  es el número de constituyentes cubiertos por la cláusula inmediata,  $\Phi_1$  es el conjunto de características de la cláusula inmediata y  $\Phi_2$  es el conjunto de características de las cláusulas superiores. Las características que se usan en este enfoque se describen en la [Tabla 4](#).

---

|                                                                                           |
|-------------------------------------------------------------------------------------------|
| etiqueta previa (pl)                                                                      |
| predicado-POS <sup>2</sup> (predpos), predicado-Lex (predlex)                             |
| predicado-tipo(predtype)                                                                  |
| tag(ctag), voz(v), posición(p), ruta(path), cabecera-lex(headlex)                         |
| cabecera-POS(hp), contenido-cabecera(chl), etiqueta-previa(ptag),                         |
| etiqueta-previa-cabecera-lex(phl), siguiente-etiqueta(ntag), siguiente-etiqueta-cab(nhl), |
| ruta-clausula-inmediata(pth-im-cl), ruta-inicio-fin(path-beg-end), nivel-clausula(l-cl),  |
| es-clausula-final(cl-bn), roles-clausula-inmediata(in-cl-roles)                           |

---

Tabla 4: Características empleadas por [42]

Un aspecto que es importante tomar en consideración, es que a fin de determinar los elementos correspondientes a las cláusulas inmediata

y superiores, se utiliza la notación BIO (Begin-Input-Output). Los resultados que se obtienen con este enfoque se detallan en la [Tabla 5](#). Como se puede observar, los valores de precisión, *recall* y F - Measure distan del ideal para el caso del corpus de desarrollo y pruebas, ya que como veremos más adelante, otros enfoques obtienen valores alrededor de 80 % para estas medidas.

|               | Precisión | <i>Recall</i> | F - Measure |
|---------------|-----------|---------------|-------------|
| Entrenamiento | 96.40 %   | 92.28 %       | 94.29       |
| Desarrollo    | 69.78 %   | 62.56 %       | 65.97       |
| Pruebas       | 68.42 %   | 61.47 %       | 64.76       |

Tabla 5: Resultados obtenidos usando del enfoque propuesto por [42]

La propuesta que lleva a cabo [43], considera el modelo de etiquetado con algunas variantes respecto al enfoque anterior, en virtud de que se realiza en la competición posterior (CoNLL 2006), lo que lógicamente enriquece la experimentación. Los cambios propuestos son los siguientes:

- Se utiliza el analizador sintáctico de Charniak, ya que comete menos errores durante las fases de análisis (10.08 % frente al 11.89 % que presenta el de Collins).
- Se considera los 5 argumentos que no concuerdan con argumentos durante el etiquetado (AM-MOD, A1, AM-NEG, AM-DIS y Ao) y realiza una fase de post-procesamiento para clasificar de mejor forma estos argumentos.
- Durante la fase de post-procesamiento se analiza los errores causados por la fase de análisis sintáctico.
- Utiliza una variante en las características empleadas para la función de clasificación ([Tabla 6](#)).

---

Lema (lemma), Path (ruta sintáctica del árbol de análisis del constituyente al predicado),

Tipo de frase, Posición (posición del constituyente respecto al predicado),

Voz (activa o pasiva), Subcategorizaciones.

**Características adicionales:**

POS del predicado, sufijo del predicado (3 últimos caracteres),

entidad nombrada (si termina con una entidad nombrada se agrega LOC, ORG, PER o MISC),

tamaño de la ruta(distancia del predicado y el constituyente),

ruta parcial (ruta parcial del constituyente al menor ancestro del predicado y el constituyente),

anidamiento de la cláusula(número de cláusulas en la ruta entre el constituyente y su predicado),

POS de la palabra-cabecera, POS de la última palabra.

---

Tabla 6: Características empleadas por [43].

Los resultados obtenidos por este segundo enfoque se detallan en la [Tabla 7](#)<sup>3</sup>. Como se observa, los valores generales de precisión, *recall*

<sup>3</sup> En la Parte II de este documento, se describen las características del corpus CoNLL 2005, que se ha empleado en estos enfoques y en nuestra propia experimentación.

y F - Measure son superiores al modelo anterior. Consideramos que se obtienen mejores resultados, debido a que se toman en cuenta los errores que ocasionan las fases de análisis sintáctico, lo que repercute de forma notable en las fases posteriores de análisis semántico.

|                | Precisión | Recall  | F - Measure |
|----------------|-----------|---------|-------------|
| Desarrollo     | 79.65 %   | 71.34 % | 75.27       |
| Test WSJ       | 80.48 %   | 72.79 % | 76.44       |
| Test Brown     | 71.13 %   | 59.99 % | 65.09       |
| Test WSJ+Brown | 79.30 %   | 71.08 % | 74.97       |

Tabla 7: Resultados obtenidos usando del enfoque propuesto por [43].

### 8.3.2 Etiquetado basado en modelos de CAC

En esta sección revisaremos un interesante enfoque propuesto por [22] durante la competición realizada en la CoNLL edición 2005.

La primera tarea que llevan a cabo los autores antes mencionados, es derivar un árbol para cada instancia de entrenamiento, usando la estructura sintáctica que genera el analizador Collins. Dentro de dichos árboles los nodos se renombran con el rol semántico correspondiente. Como se aprecia en la Figura 9, las líneas interpunktadas muestran los límites que serán eliminados, manteniéndose únicamente aquellos nodos que son hermanos de un nodo que esté en el camino que va del verbo a la raíz del árbol. Los hijos del nodo se etiquetan como externos, mientras que al nodo padre se le asigna la etiqueta correspondiente.

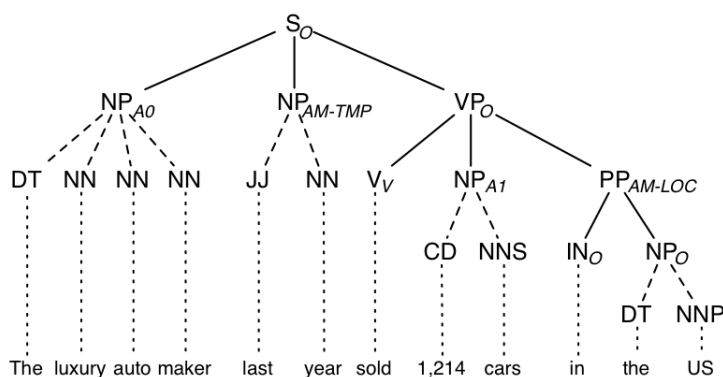


Figura 9: Árbol de análisis sintáctico etiquetado con roles semánticos respecto al predicado *sell* [22].

A más del enfoque basado en etiqueta externa (*O*, *output*), los autores han experimentado con dos propuestas alternativas de etiquetado:

1. Etiquetado de los hijos del constituyente con la etiqueta interna (*inside*). - Para el ejemplo de la figura Figura 9, a los hijos del nodo *PP*, esto es, *IN* y *NP*, se les asignaría la etiqueta interna (*I*, *inside*). Los resultados obtenidos con esta variante son similares a usar etiqueta externa, lo que indica que el modelo no puede predecir de forma adecuada los argumentos que forman parte de un constituyente. De igual forma, conocer que el hijo de un nodo está dentro de un argumento no constituye una estrategia

que permita mejorar sustancialmente la fase de predicción de las etiquetas.

2. Etiquetado de los hijos del constituyente con la etiqueta del padre . - En este caso, los nodos *IN* y *NP* tomarían la etiqueta *AM-LOC* del nodo padre. Los resultados obtenidos con esta alternativa son muy pobres, en virtud de que el número de nodos con etiqueta aumentó de forma drástica, lo que llevó a sobre-etiquetar los argumentos.

#### Modelo de etiquetado CAC

Los autores definen un campo aleatorio condicional para etiquetar  $y$ , cuando se ha observado un árbol  $x$  como sigue:

$$p(y | x) = \frac{1}{Z(x)} \exp \sum_{c \in \mathcal{C}} \sum_k \lambda_k f_k(c, y_c, x)$$

donde  $\mathcal{C}$  es el conjunto de *cliques*<sup>4</sup> en el árbol que se ha observado,  $\lambda_k$  son los parámetros del modelo y  $f_k$  es la función de características que establece la correspondencia de una etiqueta basada en el *clique* a un vector de valores escalares.  $Z(x)$  es una función de normalización, que garantiza que  $p$  es una distribución de probabilidad válida. Esta ecuación puede ser reescrita de la siguiente forma:

$$p(y | x) = \frac{1}{Z(x)} \exp \left\{ \sum_{v \in \mathcal{C}_1} \sum_k \lambda_k g_k(v, y_v, x) + \sum_{u, v \in \mathcal{C}_2} \sum_j \lambda_j h_j(u, v, y_u, y_v, x) \right\}$$

donde  $\mathcal{C}_1$  son los vértices en el grafo y  $\mathcal{C}_2$  son los máximos *cliques* en el grafo, considerando a todos los pares (padres e hijos). La función característica se ha dividido en  $g$  y  $h$ , donde cada una se relaciona con uno y dos *cliques* de nodos, respectivamente.

#### Características empleadas

La estructura de árbol empleada en este enfoque requiere que las características incorporen las etiquetas de cada nodo. Cada característica se compone de dos funciones binarias  $g$  y  $h$  (tipo de característica y valor). A continuación se puede apreciar un resumen de las características empleadas.

- Características básicas . - Palabra principal (*head*) y su POS, categoría sintáctica de la frase, ruta de la frase, marco sintáctico, posición relativa respecto al predicado, distancia al predicado, lemma, predicado, voz del predicado, subcategorización del predicado [80].
- Características de contexto . - Palabra principal del primer SN, palabras principales izquierda y derecha y su categoría sintáctica, primera y última palabra en la frase y su POS, categoría sintáctica del padre y la palabra principal.

<sup>4</sup> En la teoría de los grafos, un *clique* en un grafo no dirigido  $G$  es un conjunto de vértices  $V$  tal que para todo par de vértices de  $V$ , existe una arista que los conecta [34].



- Predecesor común del verbo . - Categoría sintáctica del predecesor del verbo y el nodo, que se encuentre en el nivel más superior.
- Conjunciones de características . - Estas características se usan de forma conjunta: lema del predicado + categoría sintáctica, lema del predicado + posición relativa, categoría sintáctica + primera palabra de la frase.

Los resultados obtenidos se pueden observar en la [Tabla 8](#). Como se aprecia, los valores en cuanto a *recall* son inferiores al 70 %. Esto quizá se debe a que el modelo no es capaz de generalizar de forma óptima los conjuntos de argumento - rol que posee cada una de las oraciones. A pesar de ello, la línea base que es alrededor del 50 % en precisión y 30 % de *recall*, ha sido superada con la aplicación de esta técnica.

|                | Precisión | Recall  | F - Measure |
|----------------|-----------|---------|-------------|
| Desarrollo     | 73.51 %   | 68.98 % | 71.17       |
| Test WSJ       | 75.81 %   | 70.58 % | 73.10       |
| Test Brown     | 67.63 %   | 60.08 % | 63.63       |
| Test WSJ+Brown | 74.76 %   | 69.17 % | 71.86       |

Tabla 8: Resultados obtenidos usando el enfoque basado en CAC [22].

### 8.3.3 Etiketado de roles semánticos usando MSV

En esta sección revisaremos los etiquetadores de roles semánticos implementados por [55] y [50], en la competición CoNLL 2005.

La implementación realizada por [55], se basa en una estrategia compuesta por dos fases: identificación de frases que pueden ser argumentos de los predicados y etiquetado de los argumentos de acuerdo a cada predicado. Para la fase de identificación, se utiliza un clasificador binario que indica si alguno de los constituyentes en el árbol sintáctico es un posible argumento. Los argumentos potenciales se pasan al clasificador binario de etiquetas, que decide si una etiqueta (rol) pertenece a dicho argumento. Una vez que se ha ejecutado este proceso, se lleva a cabo la fase de post-procesado, que seleccionará aquellos elementos etiquetados que cumplan de mejor manera con las restricciones de los argumentos del predicado. Los principales características que definen el funcionamiento de este sistema se detallan a continuación:

- Datos de entrada . - En cada oración primero se mapea el predicado a un constituyente en el árbol sintáctico. En los casos en los que el predicado no esté representado por constituyentes, el verbo generalmente se compone de dos o más palabras, donde la primera palabra es el verbo principal (*carry out, gotten away, served up*, etc.). En estas situaciones se utiliza la primera palabra del constituyente como representación del predicado. A continuación se identifica cada constituyente en el árbol como un potencial argumento y se lo etiqueta de forma apropiada.
- Componentes de aprendizaje . - Se trabaja con la librería libSVM<sup>5</sup>, una implementación de código abierto de MSV. De igual forma,

<sup>5</sup> <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>

la función de *kernel* utilizada es de tipo base radial. Es importante considerar que en el etiquetado de roles semánticos, las características son nominales y por ello se ha usado la representación estándar (si la característica nominal tiene  $n$  valores discretos, entonces se tienen  $n$  características binarias).

- Características del identificador-clasificador . - Básicamente se ha utilizado el conjunto de características propuestas por [30], salvo la exclusión de algunas características de tipo sintáctico (Tabla 9). Es importante observar que estas características reducidas han obtenido los mejores resultados en el F-Measure, en comparación con otras combinaciones que agregan la palabra principal, la ruta completa o el predicado.

---

Tipo de frase . - Etiqueta de la frase del argumento.

Posición . - Antes o después del predicado.

Voz . - Indica si el predicado está en voz activa o pasiva.

Subcategorización . - Etiquetas de los hijos del predicado en el árbol sintáctico.

Ruta corta . - Ruta desde el padre del argumento al padre del predicado en el árbol sintáctico.

---

Tabla 9: Características empleadas por [55].

- Clasificador para entrenamiento y pruebas . - En cada predicado todos los constituyentes del árbol sintáctico están marcados con un valor verdadero (*true*) , si existe algún argumento del predicado y con falso (*false*) en caso contrario. En virtud de que se presentaron una gran cantidad de falsos negativos, se probaron dos filtros, uno de tipo aleatorio y un segundo basado en las etiquetas de las frases. El primer filtro selecciona de forma aleatoria un porcentaje de argumentos por cada etiqueta del argumento. El segundo filtro elimina los constituyentes de una sola palabra que no son usados nunca. La selección de las frases se realiza en base a aquellas que superan un cierto umbral (0.01) en su frecuencia. En la fase de experimentación se comprobó que el primer filtro genera mejores resultados para el F-Measure. El último aspecto a considerar es el valor de los parámetros  $C$  y  $\gamma$  del *kernel* de base radial. Los valores que mejores resultados produjeron, fueron de 2 y 0.125, respectivamente.
- Características del etiquetador-clasificador . - A continuación se listan las características utilizadas en los clasificadores de etiquetado:
  - Predicado . - Lema del predicado.
  - Ruta . - Ruta sintáctica del constituyente del argumento al predicado.
  - Palabra principal . - Palabra principal del constituyente del argumento.
  - Tipo de frase, posición, voz y subcategorización . - Igual que en el identificador-clasificador.

Los resultados que se han obtenido con la aplicación de esta técnica se detallan en la Tabla 10. Es importante observar que se obtienen buenos

resultados, considerando que las MSV trabajan con valores numéricos y el proceso de representar texto como números es una tarea que requiere ser fundamentada de forma adecuada.

|                | Precisión | Recall  | F - Measure |
|----------------|-----------|---------|-------------|
| Desarrollo     | 73.57 %   | 71.87 % | 72.71       |
| Test WSJ       | 74.66 %   | 74.21 % | 74.44       |
| Test Brown     | 65.52 %   | 62.93 % | 64.20       |
| Test WSJ+Brown | 73.48 %   | 72.70 % | 73.09       |

Tabla 10: Resultados obtenidos usando del enfoque propuesto por [55].

El segundo enfoque que pasamos a revisar, es el que propone [50]. La primera tarea que se realiza es el cambio de representación de los datos originales, de acuerdo a la propuesta de [32]:

- La representación de los roles fue convertida a IOB2 . - De acuerdo a [73], la identificación de constituyentes (*chunking*), puede ser vista como una tarea de etiquetado usando tres tipos de etiquetas: I (*in*), O (*output*) y B(*begin*). Dentro de este tipo de anotación existen algunas variantes y la más usada es IOB2. En la [Tabla 11](#) podemos observar un ejemplo de una oración y el formato de anotación que se tendría.

|           |                                                                                                                                                                                      |
|-----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Oración   | In early trading in Hong Kong Monday,<br>gold was quoted at \$366.50 an ounce.                                                                                                       |
| Chunks    | In [Nearly trading <sub>N</sub> ] in [NHong Kong <sub>N</sub> ] [NMonday <sub>N</sub> ],<br>[Ngold <sub>N</sub> ] was quoted at [N\$366.50 <sub>N</sub> ] [Nan ounce <sub>N</sub> ]. |
| Anotación | In/O early/I trading/I in/O Hong/I Kong/I Monday/B ,/O<br>gold/I was/O quoted/O at/O \$/I 366.50/I an/B ounce/I ./O                                                                  |
| IOB1      | O I I O I I B O I O O O I I B I O                                                                                                                                                    |
| IOB2      | O B I O B I B O B O O O B I B I O                                                                                                                                                    |

Tabla 11: Representación de esquemas IOB [73].

- Cambio del metodo palabra por palabra a frase por frase . - En el trabajo realizado por [32], se demuestra que los clasificadores que analizan frases tienen un mejor rendimiento y requieren un menor número de muestras ([Tabla 12](#)).

| Metodo      | Precisión | Recall  | F-Measure |
|-------------|-----------|---------|-----------|
| Por frase   | 69.04 %   | 54.68 % | 61.02     |
| Por palabra | 68.34 %   | 45.16 % | 54.39     |

Tabla 12: Comparación de los métodos frase a frase y palabra a palabra [32].

A continuación se establece la forma que permitió codificar las características. En la [Figura 10](#) se puede apreciar un ejemplo de este proceso y a continuación se detallan los elementos de información extraídos:

- **1st** Palabras (Bag of Words) . - Todas las palabras que aparecen en el conjunto de entrenamiento.
- **2nd** POS
- **3rd** Etiquetas base de las frases . - Análisis parciales (*chunks*+cláusulas) predecidos por los procesadores UPC.
- **4th** Entidades nombradas
- **5th** Profundidad de los componentes léxicos (*Tokens*) . - Indica el grado de profundidad desde el predicado. Se ha usado los resultados generados por el analizador de Charniak.
- **6th** Palabras del predicado.
- **7th** Posición de los componentes léxicos . - Posición de la palabra actual del predicado. Puede tomar 3 valores posibles: "antes" (*before*), "después" (*after*) y "-" para el predicado.
- **8th** Distancia de la frase en ruta plana . - Distancia del componente léxico actual como un número en la frase.
- **9th** Ruta plana . - Ruta de la palabra actual al predicado, como una cadena de frases.
- **10th** Clases semánticas . - Se ha recogido las 1.000 palabras BP (*Begin Phrase*, Inicio de Frase) principales que ocurren con más frecuencia en el conjunto de entrenamiento y se ha tratado de clasificarlas de forma manual. Las 5 clases (persona, organización, tiempo, ubicación, número y moneda) son relativamente fáciles de clasificar. De entre las 1.000 palabras, 343 pueden ser clasificadas dentro de estas 5 clases. Las palabras restantes no se pueden clasificar.

| (1st)   | (2nd) | (3rd) | (4th) | (5th) | (6th) | (7th)  | (8th) | (9th)         | (10th)       | (11th)   |
|---------|-------|-------|-------|-------|-------|--------|-------|---------------|--------------|----------|
| The     | DT    | B-NP  | 0     | 0     | pay   | before | -1    | NP VP         | -            | B-A0     |
| company | NN    | I-NP  | 0     | 0     | pay   | before | -1    | NP VP         | organization | I-A0     |
| paid    | VBD   | B-VP  | 0     | 0     | pay   | -      | 0     | -             | -            | B-V      |
| five    | CD    | B-NP  | 0     | 2     | pay   | after  | 1     | VP NP         | -            | B-A1     |
| cents   | NNS   | I-NP  | 0     | 2     | pay   | after  | 1     | VP NP         | money        | I-A1     |
| a       | DT    | B-NP  | 0     | 2     | pay   | after  | 2     | VP NP         | -            | I-A1     |
| share   | NN    | I-NP  | 0     | 2     | pay   | after  | 2     | VP NP         | -            | I-A1     |
| in      | IN    | B-PP  | 0     | 1     | pay   | after  | 3     | VP NP PP      | -            | B-AM-TMP |
| April   | NNP   | B-NP  | 0     | 2     | pay   | after  | 4     | VP NP PP NP   | time         | I-AM-TMP |
| .       | .     | O     | 0     | -1    | pay   | after  | 4     | VP NP PP NP O | -            | O        |

Data representation

|         |     |      |   |    |     |        |    |               |              |          |
|---------|-----|------|---|----|-----|--------|----|---------------|--------------|----------|
| company | NN  | B-NP | 0 | 0  | pay | before | -1 | NP VP         | organization | B-A0     |
| paid    | VBD | B-VP | 0 | 0  | pay | -      | 0  | -             | -            | B-V      |
| cents   | NNS | B-NP | 0 | 2  | pay | after  | 1  | VP NP         | money        | B-A1     |
| share   | NN  | B-NP | 0 | 2  | pay | after  | 2  | VP NP         | -            | I-A1     |
| in      | IN  | B-PP | 0 | 1  | pay | after  | 3  | VP NP PP      | -            | B-AM-TMP |
| April   | NNP | B-NP | 0 | 2  | pay | after  | 4  | VP NP PP NP   | time         | I-AM-TMP |
| .       | .   | O    | 0 | -1 | pay | after  | 4  | VP NP PP NP O | -            | O        |

Figura 10: Ejemplo de una oración anotada [50].

El último aspecto que se debe especificar, es la configuración de la MSV. Los parámetros que se utilizaron se listan a continuación:

- Tipo de función *kernel* empleada . - Polinomial de grado 2.

- Rango de la ventana . - La MSV puede utilizar los componentes léxicos alrededor del componente de interés. Se ha trabajado con dos componentes a la izquierda y dos a la derecha.
- Método para resolver el problema multiclase . - Se ha utilizado el método 1 vs demás.
- Costo de la violación de las restricciones . - Se usó el valor por defecto de 1.0

Como aspecto final recogemos los resultados que obtuvo el segundo enfoque (Tabla 13).

|                | Precisión | Recall  | F - Measure |
|----------------|-----------|---------|-------------|
| Desarrollo     | 71.68 %   | 64.93 % | 68.14       |
| Test WSJ       | 74.15 %   | 68.25 % | 71.08       |
| Test Brown     | 63.24 %   | 54.20 % | 58.37       |
| Test WSJ+Brown | 72.77 %   | 66.37 % | 69.43       |

Tabla 13: Resultados obtenidos usando del enfoque propuesto por [50].

#### 8.4 MEJORES SISTEMAS DE LA COMPETICIÓN CONLL 2005

En esta sección revisamos los 5 mejores sistemas de la competición, su estructura de funcionamiento y el enfoque que han seguido los autores a fin de realizar el proceso de ERS. En la Tabla 14, se puede apreciar el listado de sistemas, ordenados por el *F-Measure* obtenido en el corpus de pruebas tWSJ+tBrown. A continuación listamos los sistemas y detallamos la base de su funcionamiento<sup>6</sup>:

- Punyakanok ([39]) . - Combina la salida de múltiples clasificadores de argumentos en una salida predicado-argumento coherente. Para realizar esta tarea, plantea un problema de optimización resuelto a través de Programación Entera Lineal.
- Haghghi ([74]) . - Implementa un modelo que captura las dependencias de los argumentos de un predicado a través de modelos *log-lineales*, todo ello dentro de un esquema de clasificación (*ranking*).
- Marquez ([46]) . - Usa la información sintáctica parcial y total y convierte la tarea en un proceso de etiquetado secuencial de tipo BIO. A más de ello, utiliza un clasificador binario de etiquetas, que ha sido entrenado con el algoritmo AdaBoost.
- Pradhan ([59]) . - Basa su sistema de etiquetado en características derivadas de distintas vistas sintácticas y las combina con un paradigma de análisis de constituyentes basado en frases. Utiliza representación IOB y clasificadores basados en MSV.
- Surdeanu ([71]) . - Utiliza análisis sintáctico completo. El problema de etiquetado es modelado a través de atributos léxicos, sintácticos y semánticos enriquecidos. Emplea clasificadores entrenados con el algoritmo AdaBoost.

<sup>6</sup> Para denominar a cada sistema hemos tomado el nombre con el que participaron dentro de la competición CoNLL.

| Systema     | Precisión | Recall  | F-Measure |
|-------------|-----------|---------|-----------|
| punayakanok | 81.18 %   | 74.92 % | 77.92     |
| haghighi    | 78.34 %   | 75.78 % | 77.04     |
| marquez     | 78.44 %   | 74.83 % | 76.59     |
| pradhan     | 80.93 %   | 71.69 % | 76.03     |
| surdeanu    | 79.35 %   | 71.17 % | 75.04     |

Tabla 14: Listado de los 5 mejores sistemas de la competición CoNLL 2005.

#### 8.4.1 Sistema Punayakanok

Este enfoque es presentado por [39] y divide el funcionamiento de su sistema en 4 etapas:

- Selección-eliminación . - Se consideran únicamente como argumentos a los constituyentes del árbol sintáctico. A más de ello, el sistema utiliza la heurística de análisis propuesta por [80]. Esta heurística es un proceso recursivo que inicia en el verbo del cual se deben identificar los argumentos. En primera instancia la heurística devuelve los nodos hermanos del verbo (que se encuentran en el mismo nivel). Una vez que ha realizado esta tarea, sube de nivel y analiza el nodo padre del verbo y sus hermanos. El proceso se repetirá hasta que se llegue a la raíz del árbol. Adicionalmente, mencionamos que si el argumento es una PP (*Propositional Phrase*, Frase Proposicional), sus hijos son tomados en cuenta. Los candidatos que se constituyan de un único signo de puntuación se descartan.
- Identificación de argumentos . - Para realizar esta etapa se utilizan clasificadores binarios, que permiten determinar si un candidato es un argumento o no. Las características utilizadas son el estándar definido en trabajos previos:
  - **Predicado y POS del predicado.** Indica el lema del predicado y su etiqueta tipo POS.
  - **Voz.** Indica la voz del predicado.
  - **Tipo de Frase** del constituyente.
  - **Palabra principal y su etiqueta POS.** Se usan las reglas introducidas [23] para extraer esta característica.
  - **Primera y última palabra y etiquetas POS** del constituyente.
  - **Dos etiquetas POS anteriores y posteriores** al constituyente.
  - **Posición** del constituyente (antes o después del predicado).
  - **Ruta transversal** del predicado al constituyente, en el árbol sintáctico.
  - **Subcategorización** que indica la estructura de la frase alrededor del padre del predicado.
  - **Clase del verbo**, descrita dentro de las categorías del Prop-Bank.

- **Longitud de los constituyentes**, en número de palabras y constituyentes anotados (*chunks*) por separado.
  - **Anotación de constituyentes** (*chunks*), que indica si el argumento es, contiene, se solapa o está considerado dentro de un constituyente anotado con ese tipo.
  - **Longitud de la constituyentes anotados**. Cuenta el número de constituyentes anotados del predicado al argumento.
  - **Posición relativa de la cláusula**. Indica la posición de la palabra relativa al predicado, dentro de un pseudo-árbol sintáctico construido a partir de la cláusula del constituyente.
  - **Cobertura de la cláusula**. Describe la porción de la cláusula local (del predicado) que está cubierta por el argumento.
- **Clasificación de argumentos** . - Esta etapa permite asignar las etiquetas definitivas a los argumentos candidatos que se han determinado en la etapa anterior. La tarea de asignación se puede llevar a cabo gracias a que se entrena un clasificador multiclase que determina los tipos de argumentos. A fin de reducir el número excesivo de candidatos incorrectos (falsos positivos) debido a la etapa anterior, el clasificador puede marcar como nulo (*NULL*) a un argumento. Para llevar a cabo esta tarea se utilizan las características empleadas en la etapa de identificación y se agregan las siguientes:
    - Marco sintáctico . - Describe el patrón secuencial de los sintagmas nominales y del predicado en la oración. Esta característica es introducida por [80].
    - Frase proposicional principal . - Es la palabra principal de la primera frase antes de la preposición.
    - Negación y Modo . - Característica que indica si el argumento es la base para los argumentos AM-NEG o AM-MOD.
    - NE . - Indica si el argumento es, contiene, se solapa o es contenido en una entidad nombrada de ese tipo.
  - **Inferencia** . - El objetivo de esta etapa es incorporar conocimiento lingüístico y una estructura de conocimiento, verbigracia, argumentos que no se solapan o que cada verbo debe tomar al menos un argumento de cada tipo. Este conocimiento se emplea para resolver problemas de inconsistencia de la etapa de clasificación. Se utiliza el proceso de inferencia propuesto por [60], en el cual se formula el planteamiento como un problema de Programación Entera Lineal. En este problema se toma como entradas la confianza o certeza de los argumentos propuestos por el clasificador. Como salida el problema devuelve una solución óptima que maximiza la suma de los puntajes de confianza, por ejemplo, las probabilidades condicionales estimadas por el clasificador de argumentos. La solución óptima que maximice dichos puntajes debe estar sujeta a las restricciones que posee el dominio.

En términos formales, el clasificador de argumentos debe asignar etiquetas a un conjunto de argumentos,  $S^{1:M}$ , indexados de 1 a  $M$ . Cada argumento  $S^i$  puede tomar cualquier etiqueta del conjunto de etiquetas  $P$ . El conjunto de argumentos indexados puede tomar un conjunto de etiquetas, esto es,  $c^{1:M} \in P^M$ . Si se asume

que el clasificador devuelve una distribución de probabilidad condicional,  $\text{Prob}(S^i = c^i)$ , entonces, dada una oración, el proceso de inferencia buscará una asignación que maximice la función objetivo:

$$\hat{c}^{1:M} = \arg \max_{c^{1:M} \in P^M} \sum_{i=1}^M \text{Prob}(S^i = c^i)$$

sujeto a las restricciones lingüísticas y estructurales. Esta función objetivo refleja el número correcto de predicciones de argumentos. Las restricciones se codifican como se especifica a continuación:

- Argumentos que no se solapan o que contienen a otros.
- Que no existan argumentos con clases duplicadas para los roles Ao-A5.
- Se debe considerar exactamente un argumento V (verbo) por cada predicado.
- Si existe un patrón C-V (complemento-verbo), entonces no puede existir un patrón V-A1-CV (verbo-rol A1-complemento verbal).
- Si existe un argumento *R-arg* (argumento relativo, pronombre relativo), entonces deberá existir un argumento *arg*.
- Si existe un argumento *C-arg* (continuidad del argumento).
- Dado un predicado, algunos tipos de argumentos son ilegales (por ejemplo, el predicado *stalk* puede tener únicamente los argumentos Ao o A1).

#### 8.4.2 Sistema Haghghi

Este enfoque es presentado por [74] y se basa en dos tareas fundamentales: uso de modelos locales para etiquetar a los posibles candidatos y aplicación de un modelo conjunto para recalcular las puntuaciones de dichos candidatos. A continuación describimos de manera breve las partes más importantes de esta propuesta:

- Modelos Locales . - Con estos modelos se plantea la idea de descomponer la probabilidad entre todas la etiquetas (argumentos y una etiqueta *NONE* para el caso de que no exista argumento). Este planteamiento se enfoca como un modelo de *identificación* y *clasificación*. La identificación se encarga de especificar si una frase es un argumento (ARG) o no lo es (*NONE*). La clasificación por su parte, etiqueta cada argumento potencial con su etiqueta correspondiente (Ao, A1, etc.). En el siguiente listado se recogen las características utilizadas para ambos modelos<sup>7</sup>.
  - Palabras Temporales . - Palabra principal de un nodo temporal, por ejemplo, “February” o “at the morning”.
  - Sujeto Faltante . - Indica el sujeto que debería tener el predicado en su ubicación predeterminada, en caso de estar ausente.

<sup>7</sup> Este listado únicamente se recogen las características nuevas que propone este enfoque, en virtud de que las demás son las mismas que se utilizan en el sistema Punyakanok.



- Ruta Proyectada . - Ruta desde la máxima proyección del predicado al nodo.
- Ruta y Sujeto Faltante.
- Ruta Proyectada y Sujeto Faltante.

En el análisis de las oraciones del corpus, se pudo observar que existe una gran cantidad de errores producidos por la división de los argumentos A<sub>0</sub> y A<sub>1</sub>. Por ejemplo, en la [Figura 11](#), se puede notar que el predicado *expected*, no posee un sujeto en la posición típica (por ello se marca con la etiqueta *NONE*). El auxiliar *is* lo ha subido a la posición actual. Para determinar estos casos se utiliza la característica Sujeto Faltante, que indica si el predicado carece de sujeto. Al combinar la característica antes mencionada con la Ruta, es posible aprender la rutas típicas de los sujetos que han sido desplazados en el árbol.

En la [Figura 11](#), también se presenta un segundo caso de desplazamiento. El predicado *widen* comparte *the trade gap* con el predicado *expect* como argumento A<sub>1</sub>. En virtud de que *expect* es un verbo que ha desplazado al sujeto y el sujeto correspondiente a *widen* no está en su posición típica, debería buscarse en la posición de *expect*. Esto indica que la mejor opción es utilizar rutas relativas al predicado *expect*, para encontrar los argumentos de *widen*. Para realizar esta tarea, se utiliza la característica Ruta Proyectada, que toma la ruta de la máxima proyección a un argumento del nodo.

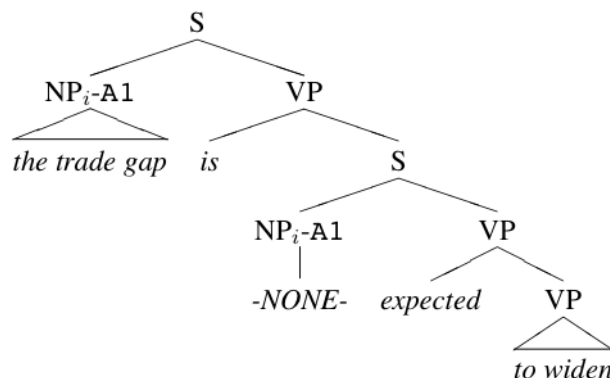


Figura 11: Ejemplo de error producido por el desplazamiento de argumentos [74].

- Modelo Conjunto. - Este modelo se encarga de asignar puntuaciones a las etiquetas de todos los nodos del árbol sintáctico. El modelo es entrenado para recalcular un conjunto de N etiquetas de acuerdo al modelo local y para realizar esta tarea opera en las categorías semánticas de los argumentos, observando las etiquetas y las características internas de cada argumento candidato. En la [Figura 12](#) se puede observar un ejemplo de anotaciones de roles:
  - Secuencia de etiquetas de los argumentos centrales . - Se concatena con la voz del predicado. Ejemplo: [voz:activa:A1,V,A3,A4]
  - Secuencia plana de etiquetas de los argumentos centrales . - Similar al anterior, pero mezclando etiquetas consecutivas iguales.

- Secuencias de etiquetas de los argumentos centrales y tipo de frase . - Igual al primero, pero agrega los tipos de las frases. Ejemplo: [voz:activa, NP-A1,V,PP-to-A3,PP-from-A4]
- Etiquetas repetidas de los argumentos centrales con el tipo de frase . - Tipos de frases para aquellos nodos con la misma etiqueta central.
- Etiquetas repetidas de los argumentos centrales con tipo de frase e información hermana/adyacente . - Similar a la característica anterior, pero indica si todos los argumentos repetidos son adyacentes en términos de cantidad de palabras.

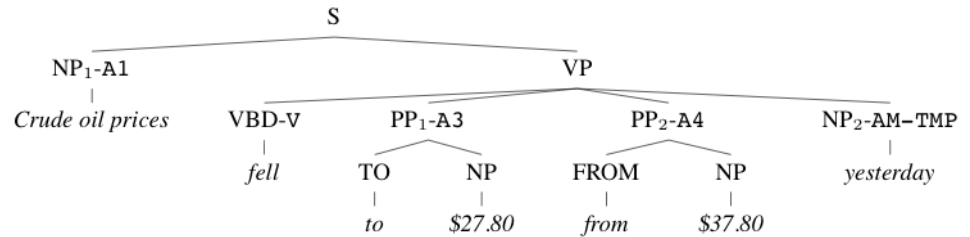


Figura 12: Ejemplo de un árbol sintáctico con anotaciones de roles [74].

Como paso final se deben combinar los dos modelos, debido a que el modelo local se entrena con todos los ejemplos negativos, mientras que el modelo conjunto hace lo propio con las categorías semánticas de los argumentos. El puntaje final se obtiene a partir de la mezcla del logaritmo de las probabilidades de los dos modelos (Ecuación 8.3):

$$\text{score}_{\text{SRL}}(L | t) = \alpha \text{score}_l(L | t) + \text{score}_j(L | t) \quad (8.3)$$

donde  $\alpha \text{score}_l(L | t)$  es el puntaje local de  $L$  (etiquetado),  $\text{score}_j(L | t)$  es el puntaje conjunto y  $\alpha$  es un parámetro de ajuste.

#### 8.4.3 Sistema Marquez

Los autores de esta propuesta de etiquetado, [46], plantean trabajar en base a dos objetivos principales: implementar un sistema competitivo de ERS, reduciendo el proceso a una tarea de etiquetado secuencial e investigar el efecto de reemplazar información sintáctica parcial por información sintáctica completa.

El sistema desarrollado realiza una exploración a través de la información sintáctica parcial y total, con la finalidad de recoger con subconjunto de constituyentes organizados de forma secuencial (por ejemplo, aquellos constituyentes que no sean contenedores). Las proposiciones son tratadas de forma independiente, esto es, de cada verbo del que se debe anotar sus argumentos, se crea una secuencia de componentes léxicos (*tokens*) que serán etiquetados. Este proceso es llamado **secuencialización**.

Estos componentes léxicos son seleccionados a través de la exploración de las oraciones: extensión y límites de las cláusulas (las regiones a la derecha del verbo correspondientes a los ancestros son omitidas en caso del análisis parcial). Las propiedades más importantes de este proceso son las siguientes:

- Los constituyentes sintácticos de más alto nivel que se correspondan con esta región son seleccionados.
- La estrategia es independiente del análisis sintáctico que se proporcione.
- En el caso del análisis sintáctico completo, la selección es equivalente al proceso definido por [80].
- Esta estrategia permite un obtener *recall* de 95.67% y 90.32% para los procesadores UPC y el analizador de Charniak, respectivamente. De la misma manera, los valores de *F-Measure* que se tienen son 97.79 y 94.91.

El algoritmo de aprendizaje se fundamenta en la técnica AdaBoost. De acuerdo a lo que menciona [77] en su trabajo de aplicación, AdaBoost es un proceso iterativo que en cada ronda va seleccionando un clasificador débil,  $h_t()$ , que minimiza:

$$Z_t = \sum_i D_t(i) \exp(-y_i h_t(x_i))$$

De acuerdo a la notación de Shapire and Singer,  $D_t(i)$  es el peso en la muestra  $i$  en la ronda  $t$ .  $y_i \in \{-1, 1\}$  es la etiqueta para el ejemplo  $x_i$  y  $h_t()$  es un clasificador binario de confianza. Después de cada ronda los pesos se actualizan de la siguiente forma:

$$D_{t+1}(i) = \frac{D_t(i) \exp(-y_i h_t(x_i))}{Z_t}$$

El clasificador  $h_t()$  toma dos posibles valores,  $h_+ = \log \sqrt{\frac{W_{++}}{W_{--}}}$  y  $h_- = \log \sqrt{\frac{W_{--}}{W_{++}}}$ , donde  $W_{pq}$  es el peso de los ejemplos dada la etiqueta  $p$  que tiene la verdadera etiqueta  $q$ . Estas predicciones aseguran que los pesos en la siguiente ronda estarán balanceados. Minimizar  $Z_t$  permite hacer lo mismo con la pérdida del peso exponencial en la ronda  $t$ .

Los clasificadores binarios AdaBoost se utilizan para etiquetar secuencias en un esquema izquierda-a-derecha, con ventana de desplazamiento. De igual forma, el módulo de etiquetado debe verificar ciertas características, verbigracia, una correcta estructura BIO, que los argumentos no crucen los límites de las cláusulas, etc.

Las características que se emplearon en esta propuesta son las mismas que se han utilizado en trabajos previos.

#### 8.4.4 Sistema Pradhan

En este sistema se propone emplear clasificadores de aprendizaje supervisado para etiquetar los roles, tomando como base las características extraídas de las entradas [59]. La tarea de ERS se formula como un problema de clasificación multiclase, usando MSV. Como herramientas para la implementación de las MSV se emplea TinySVM en combinación con YamCha [40]. El proceso de clasificación utilizado es "uno contra todos", donde se entrena  $n$  clasificadores binarios ( $n$  es el número de las clases semánticas, incluida la nula).

El esquema general de funcionamiento es entrenar sistemas de ERS separados para cada vista de los árboles sintácticos, y luego emplear estos roles como características adicionales en un clasificador semántico que utilice un enfoque sintáctico plano. Los procesos y características más relevantes que posee este enfoque se detallan a continuación:

- El clasificador basado en constituyentes explora el árbol sintáctico y clasifica cada nodo como nulo (NULL) o con algún rol.
- Los sistemas de anotación de constituyentes clasifican cada frase como inicio (Begin), dentro (Inside) o fuera (Out) de un rol semántico.
- Los roles a nivel de constituyentes son mapeados a la representación IOB usada por el anotador de constituyentes (*chunker*).
- Se aplica el paradigma de validación cruzada para entrenar los constituyentes.
- Se emplean los analizadores de Charniak y Collins.

#### 8.4.5 Sistema Surdeanu

El último sistema que revisamos, es el que han desarrollado [71], donde al igual que el sistema anterior, se entrenan clasificadores en base al paradigma AdaBoost. Las partes más importantes de esta propuesta son las siguientes:

- Mapeo de argumentos a constituyentes sintácticos. - Se emplea una aproximación que parte de los nodos hoja del árbol y realiza el recorrido a los nodos superiores. Sin embargo, esta aproximación no siempre se puede aplicar. Como se observa en la Figura 13, en la parte (a) se muestra el caso en el que el mapeo de cada argumento semántico a cada constituyente sintáctico es correcto, la parte (b) ilustra la situación en que un argumento semántico puede ser mapeado correctamente a dos constituyentes sintácticos y la parte (c) muestra un mapeo de uno a varios, debido a una incorrecta estructura sintáctica (el argumento A0 se mapea a dos frases, al terminal “by” y al sintagma nominal “Robert Goldberg”, debido a una incorrecta adición de la última frase proposicional “at the University of California”).

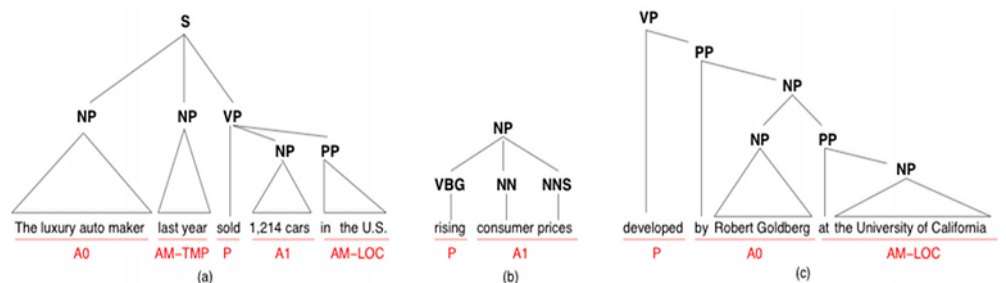


Figura 13: Ejemplo de mapeo de argumentos semánticos a constituyentes sintácticos [71].

En virtud de lo expuesto, los autores separan el mapeo uno a varios en dos clases: (a) cuando los constituyentes sintácticos

mapeados a argumentos semánticos tienen el mismo padre (Figura 13, caso b), el mapeo es correcto y (b) cuando los constituyentes sintácticos mapeados al mismo argumento tienen diferentes padres, el mapeo es producido por una sintáxis errónea.

- Características empleadas . - Las características empleadas se basan en trabajos previos y se pueden clasificar en cinco categorías:
  - Las que capturan la estructura interna de los argumentos candidatos.
  - Las que se extraen del contexto del argumento.
  - Las que describen las propiedades del predicado a analizar.
  - Las que se generan del contexto del predicado.
  - Las que modelan la distancia entre el predicado y el argumento.
- Clasificador . - Como se mencionó con anterioridad, el esquema de entrenamiento del clasificador se basa en el paradigma Ada-Boost. Para este caso, se combinan varios clasificadores o reglas simples (árboles de decisión de profundidad igual a 3) en un sólo clasificador robusto, empleando para ello un esquema de votación basado en pesos. Cada clasificador aprende secuencialmente de los ejemplos ponderados y los pesos son ajustados dinámicamente, a medida que se realiza cada iteración de aprendizaje. El ajuste de los pesos basa su comportamiento en técnicas voraces (greedy)<sup>8</sup>.

---

<sup>8</sup> Este tipo de algoritmos eligen la opción óptima en cada paso local, con la esperanza de llegar a una solución general óptima.



Parte III

ESQUEMAS DE VOTACIÓN





Los esquemas de votación Borda son una herramienta que se ha utilizado ampliamente en diversas tareas del PLN. Como ejemplo podemos citar algunas aplicaciones prácticas: desambiguación de sentidos [15], recuperación de información geográfica [16], reconocimiento de entidades nombradas [6], entre otras. En virtud de ello, consideramos que este tipo de metodología podría ser una alternativa de gran relevancia a la hora de combinar diversos sistemas de etiquetado de roles.

En este capítulo revemos los aspectos más interesantes de los esquemas de votación Borda y Fuzzy-Borda, así como su aplicación al ERS. De igual forma, realizamos un sencillo cálculo de las matrices de votación, que toman como base un par de oraciones anotadas del corpus de pruebas WSJ, que se ha explicado en el [Capítulo 8](#).

### 9.1 REGLA DE BORDA

Este método tiene sus orígenes en Francia, en el siglo XVIII, cuando Jean-Charles Borda denunció ante la Real Academia de Ciencias de París, que el sistema de votación con el que se elegía a sus miembros permitía que saliese ganador el candidato menos deseado por la mayoría de los votantes. Esto se debía, según se relata en [57], a que Borda detectó que los votantes únicamente informaban por el candidato más deseado por cada cual, descartando al resto en bloque. Para corregir este problema, Borda propuso un sistema en el que los votantes puntúan a todos los candidatos de mayor a menor, según su mérito<sup>1</sup>.

A fin de establecer el sistema de cálculo que posee el método propuesto por Borda, se deben formalizar algunas definiciones y notación [57]. Sea  $X = \{x_1, x_2, x_3, \dots, x_n\}$  un conjunto finito de candidatos, con  $n \geq 3^2$  y  $m$  votantes, debe cumplirse la igualdad  $m \geq 3$ . Se dice que  $P$  es una relación de preferencia sobre  $X$  si  $P$  es una relación binaria asimétrica, esto es, si  $x_i P x_j$ , entonces no puede ocurrir  $x_j P x_i$ . La relación de indiferencia  $I$  asociada a una relación de preferencia  $P$  recoge la ausencia de preferencia:  $x_i I x_j$ , lo que significa que ni  $x_i P x_j$  ni  $x_j P x_i$ . La relación de preferencia débil,  $P \cup I$  contempla tanto la preferencia como indiferencia:  $x_i (P \cup I) x_j$  quiere decir que  $x_i P x_j$  o  $x_i I x_j$ . Sea  $P^k$  la relación de preferencia del votante  $k$ , ( $k = 1, 2, \dots, m$ ), sobre el conjunto de  $n$  alternativas  $X$ . A tal relación de preferencia se le puede asociar la matriz:

$$\begin{pmatrix} r_{11}^k & r_{12}^k & \dots & r_{1n}^k \\ r_{21}^k & r_{22}^k & \dots & r_{2n}^k \\ \dots & \dots & \dots & \dots \\ r_{n1}^k & r_{n2}^k & \dots & r_{nn}^k \end{pmatrix}$$

donde

- <sup>1</sup> Varios autores se refieren a este procedimiento como la "regla de Borda clásica"
- <sup>2</sup> En la regla de Borda clásica se suele exigir que  $n \geq 3$ , pues en el caso de que  $n = 2$  el procedimiento coincide los métodos de pluralidad y mayorías simple y absoluta [29].

$$r_{ij}^k = \begin{cases} 1 & \text{si } x_i P x_j \\ 0 & \text{en caso contrario} \end{cases}$$

Una vez establecidos estos conceptos iniciales, se puede definir el método Borda de la siguiente forma:

- En primera instancia se consideran las puntuaciones Borda individuales, asignadas por el agente  $k$  sobre la alternativa  $x_i$ , que vienen dadas por:

$$r_k(x_i) = \sum_{\substack{j=1 \\ x_i P^k x_j}}^n r_{ij}^k = \sum_{j=1}^n r_{ij}^k$$

Este contador refleja la idea de Borda de que a cada alternativa  $x_i$  el agente  $k$  le asigna como puntuación el número de alternativas que para él son peores.

- Con las puntuaciones individuales se puede definir una alternativa colectiva:

$$r(x_i) = \sum_{k=1}^m r_k(x_i) \quad (9.1)$$

A continuación revisamos el ejemplo que plantea [10] para ilustrar su método de escrutinio.

Supongamos que se realiza una elección entre 3 candidatos y que el número de electores sea 21. Dentro de las preferencias por los candidatos se tienen 4 grupos con los siguientes valores:

- Grupo1: 6 votantes con preferencia  $x_3 \ x_2 \ x_1$
- Grupo2: 7 votantes con preferencia  $x_2 \ x_3 \ x_1$
- Grupo3: 7 votantes con preferencia  $x_1 \ x_3 \ x_2$
- Grupo4: 1 votante con preferencia  $x_1 \ x_2 \ x_3$

La regla Borda nos indica que  $x_1$  tiene pluralidad de los votos, ya que si se votase según este procedimiento, sus 8 primeras posiciones le darían como vencedor frente a los 7 obtenidos por  $x_2$  y los 6 de  $x_3$ . Según se puede observar,  $x_1$  no tiene a su favor la opinión de los electores, ya que es considerado como el peor por 13 de los 21 votantes.

Para calcular qué candidato tiene a su favor la mayor cantidad de opinión, aplicamos el método de puntuaciones Borda. Para ello, debemos construir las matrices de preferencia. A continuación ilustramos el caso de la construcción de la primera matriz, ya que las demás siguen idéntico procedimiento:

- No se puede establecer un valor de preferencia de un candidato sobre sí mismo (por ejemplo  $x_3$  sobre  $x_3$ ).

- Se debe colocar un valor igual a 1 en aquellas posiciones de la matriz donde se indique la preferencia de un candidato sobre otro:
  - En el Grupo 1 el candidato  $x_3$  tiene mayor preferencia sobre el candidato  $x_2$ , por ello se coloca 1 en la fila 3, columna 2.
  - De igual forma,  $x_3$  tiene mayor preferencia sobre  $x_1$ , por lo que colocamos 1 en la fila 3, columna 1.
  - El candidato  $x_2$  tiene preferencia sobre el candidato  $x_1$  por lo que se coloca 1 en la fila 2, columna 1.
- Las demás posiciones de la matriz deberán llenarse con valores 0.

$$M_{\text{Grupo1}} = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix} \quad M_{\text{Grupo2}} = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

$$M_{\text{Grupo3}} = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \quad M_{\text{Grupo4}} = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$$

Las puntuaciones que se obtienen (suma por filas de las matrices de preferencia ponderadas según el número de votantes) son:

$$r(x_1) = (7 \cdot 2) + 2 = 16;$$

$$r(x_2) = 6 + (7 \cdot 2) + 1 = 21;$$

$$r(x_3) = (6 \cdot 2) + 7 + 7 = 26;$$

donde el ganador resultaría ser  $x_3$ . Es interesante observar que a pesar que el candidato  $x_3$  ha quedado menos veces en primera posición que  $x_1$ , gana la elección. Esto se debe a que se considera que gran parte de los electores tienen mayor preferencia por  $x_3$ , mientras que los grupos 2 y 6 lo ubican en la peor posición.

#### 9.1.1 Borda clásico aplicada al ERS

En esta subsección planteamos un ejemplo que ilustra cómo se podría aplicar el esquema de votación Borda para combinar 3 sistemas de ERS. Si tomamos la oración “*As a result, the link between the futures and stock markets ripped apart.*” del corpus tWSJ, los tres primeros sistemas de etiquetado (Punyakankok, Haghghi, Marquez) producen el resultado que se puede apreciar en la [Tabla 15](#).

Si quisiéramos aplicar un esquema de votación Borda a los corpus que han etiquetado cada uno de los sistemas con que trabajamos ([Capítulo 8](#)), deberíamos contar con un número determinado de roles candidatos por cada argumento de la oración ( $n \geq 3$ ). Para el caso expuesto en la [Tabla 15](#), para el argumento “*As a result*”, a más del

| Constituyente | Sistema     |           |           |
|---------------|-------------|-----------|-----------|
|               | Punayakanok | Haghighi  | Marquez   |
| As            | (AM-CAU*    | (AM-CAU*  | (AM-DIS*  |
| a             | *           | *         | *         |
| result        | *)          | *)        | *)        |
| ,             | *           | *         | *         |
| the           | (A1*        | (A1*      | (A1*      |
| link          | *           | *         | *         |
| between       | *           | *         | *         |
| the           | *           | *         | *         |
| futures       | *           | *         | *         |
| and           | *           | *         | *         |
| stock         | *           | *         | *         |
| markets       | *)          | *)        | *)        |
| ripped        | (V*)        | (V*)      | (V*)      |
| apart         | (AM-DIR*)   | (AM-MNR*) | (AM-DIR*) |
| .             | *           | *         | *         |

Tabla 15: Comparativa de etiquetado realizado por 3 mejores sistemas de una oración del corpus tWSJ.

rol AM-CAU, cada sistema debería proveer dos o más posibles roles candidatos. Con esto se podrían construir las matrices de preferencia que la regla de votación específica.

Imaginemos que cada sistema provee los roles candidatos necesarios para construir las tablas de votación, con ello podríamos calcular de forma directa el orden de preferencia que tendría cada rol. Para ilustrar esta idea, vamos a trabajar con el argumento “As a result”, del ejemplo anterior y a continuación agregamos los roles que nos hacen falta (Tabla 16)<sup>3</sup>:

| Punayakanok | Haghighi | Marquez |
|-------------|----------|---------|
| AM-CAU      | AM-LOC   | AM-DIS  |
| AM-LOC      | AM-CAU   | AM-LOC  |
| AM-DIS      | AM-DIS   | AM-CAU  |

Tabla 16: Orden de preferencia de los roles candidatos etiquetados por cada sistema.

Con estos datos podemos pasar a formar la matriz de preferencia del esquema de votación Borda y calcular la votación general (considerando como candidato<sub>1</sub> al rol AM-CAU, candidato<sub>2</sub> a AM-LOC y candidato<sub>3</sub> a AM-DIS). Para el caso del sistema Punayakanok deberíamos tomar en cuenta las siguientes consideraciones <sup>4</sup>:

- No se puede expresar la preferencia de un candidato sobre sí mismo (candidato<sub>1</sub> sobre candidato<sub>1</sub>).

<sup>3</sup> Para ilustrar de mejor manera nuestro ejemplo hemos agregado dos roles candidatos a cada sistema y se ha cambiado el orden de preferencia de los roles del sistema Haghighi.

<sup>4</sup> Este proceso debe repetirse para el caso de los otros dos sistemas (Haghighi y Marquez).

- El sistema nos indica que el rol AM-CAU tiene preferencia sobre AM-LOC (candidato1 sobre candidato2), por lo que colocamos el valor 1 en la fila 1 y columna 2.
- De igual forma, el sistema muestra que el rol AM-LOC tiene preferencia sobre AM-DIS (candidato2 sobre candidato3), por lo que colocamos el valor 1 en la fila 2 y columna 3.
- Finalmente, el rol AM-CAU tiene prioridad sobre AM-DIS (candidato1 sobre candidato3), por lo que se coloca el valor 1 en la fila 2 y columna 1.

$$M_{\text{Punyanok}} = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$$

$$M_{\text{Haghighi}} = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$$

$$M_{\text{Marquez}} = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}$$

$$\text{Votacion}_{\text{final}} = \begin{pmatrix} 3 \\ 4 \\ 2 \end{pmatrix}$$

$$r(\text{AM} - \text{CAU}) = 1 + 1 + 1 = 3; \quad r(\text{AM} - \text{LOC}) = 1 + 1 + 1 + 1 = 4; \quad r(\text{AM} - \text{DIS}) = 1 + 1 = 2$$

Con este resultado la preferencia general de los roles nos queda especificada de la siguiente forma:

| <b>Punyanok</b> | <b>Haghighi</b> | <b>Marquez</b> | <b>Orden</b>  |
|-----------------|-----------------|----------------|---------------|
| AM-CAU          | AM-LOC          | AM-DIS         | <b>AM-LOC</b> |
| AM-LOC          | AM-CAU          | AM-LOC         | <b>AM-CAU</b> |
| AM-DIS          | AM-DIS          | AM-CAU         | <b>AM-DIS</b> |

Tabla 17: Orden de preferencia de los roles candidatos etiquetados por cada sistema, luego de aplicar la regla Borda.

## 9.2 FUZZY-BORDA APLICADA AL ERS

El esquema de votación Borda clásico es de bastante utilidad, sin embargo no toma en consideración el grado de preferencia que tiene cada experto por un candidato determinado. Los valores que maneja son de tipo binario, esto es, 1 o 0. Por ello, si tomamos en consideración el nivel de preferencia o peso respecto a un candidato, enriquecemos de forma sustancial el esquema de votación.

Para aplicar el esquema de votación Fuzzy – Borda, simplemente debemos añadir valores de certeza o peso a cada candidato. Estos valores podrían estar representados por los niveles de precisión, *recall* o *F-Measure* de los sistemas de etiquetado. Para ilustrar de mejor manera esta idea, tomaremos el mismo ejemplo del punto anterior y agregaremos estos valores (Tabla 18) <sup>5</sup>.

| Punyanok    | Haghighi    | Marquez     |
|-------------|-------------|-------------|
| AM-CAU: 8.2 | AM-LOC: 7.3 | AM-DIS: 9.2 |
| AM-LOC: 7.2 | AM-CAU: 5.2 | AM-LOC: 3.2 |
| AM-DIS: 6.7 | AM-DIS: 4.7 | AM-CAU: 2.7 |

Tabla 18: Orden de preferencia de los roles candidatos etiquetados por cada sistema, se agregan pesos o niveles de certeza.

A fin de poder calcular los pesos de las matrices de preferencia, se debe aplicar una fórmula que considere los valores de certeza de cada sistema:

$$r_{j,k}^i = \frac{w_j^i}{w_j^i + w_k^i} \quad (9.2)$$

donde,  $w_j^i$  representa el peso que el sistema  $i$  asigna al candidato  $j$ .  $r_{j,k}^i$  es el elemento de la fila  $j$ , columna  $k$  de la matriz de preferencia del sistema  $i$ .

Por ejemplo, si quisiésemos calcular el primer peso (fila 1, columna 1) de la matriz del primer sistema, tendríamos los siguientes valores:  $r_{1,1}^1 = \frac{w_1^1}{w_1^1 + w_1^1} = \frac{8,2}{8,2+8,2} = 0,5$ . Para calcular el tercer peso (fila 1, columna 3) se utilizarían los siguientes valores:  $r_{1,3}^1 = \frac{w_1^1}{w_1^1 + w_3^1} = \frac{8,2}{8,2+6,7} = 0,55033$ . Este proceso se debe repetir para cada elemento de la matriz de preferencia de todos los sistemas.

Si aplicamos la Ecuación 9.2 con los datos de la Tabla 18, obtendremos los siguientes resultados:

$$M_{\text{Punyanok}} = \begin{pmatrix} 0.5 & 0.53 & 0.55 \\ 0.47 & 0.5 & 0.52 \\ 0.45 & 0.48 & 0.5 \end{pmatrix}$$

<sup>5</sup> Los pesos de los candidatos son determinados por cada sistema, sin embargo, se debe procurar usar una escala de 0 (mínima preferencia) a 10 (máxima preferencia). Para nuestro ejemplo hemos usado valores de 0 a 10.

$$M_{\text{Haghighi}} = \begin{pmatrix} 0.5 & 0.42 & 0.53 \\ 0.58 & 0.5 & 0.6 \\ 0.47 & 0.39 & 0.5 \end{pmatrix}$$

$$M_{\text{Marquez}} = \begin{pmatrix} 0.5 & 0.46 & 0.23 \\ 0.54 & 0.5 & 0.26 \\ 0.77 & 0.74 & 0.5 \end{pmatrix}$$

$$\text{Votacion}_{\text{final}} = \begin{pmatrix} 4.2 \\ 4.5 \\ 4.8 \end{pmatrix}$$

$$r(\text{AM} - \text{CAU}) = 0,5 + 0,53 + 0,55 + 0,5 + 0,42 + 0,53 + 0,5 + 0,46 + 0,23 = 4,22$$

Como podemos observar en la [Tabla 19](#), al utilizar niveles de confianza el candidato ganador varía drásticamente. En el caso de Borda clásico el ganador era el rol AM-LOC, mientras que en el esquema Fuzzy-Borda el ganador pasa a ser AM-DIS.

| Punyakonok | Haghighi | Marquez | Orden         |
|------------|----------|---------|---------------|
| AM-CAU     | AM-LOC   | AM-DIS  | <b>AM-DIS</b> |
| AM-LOC     | AM-CAU   | AM-LOC  | <b>AM-LOC</b> |
| AM-DIS     | AM-DIS   | AM-CAU  | <b>AM-CAU</b> |

Tabla 19: Orden de preferencia de los roles candidatos etiquetados por cada sistema, luego de aplicar la regla Fuzzy-Borda.

Como se ha podido notar, si no se cuenta con el número de roles candidatos requeridos por los esquemas de votación Borda, no será posible aplicarlos. Por ello, consideramos que se deben tener presentes las siguientes apreciaciones:

- A fin de poder construir las matrices de votación de los esquemas Borda, es necesario que cada sistema proponga roles que formen parte de un mismo dominio, esto es, que todos los sistemas asignen los mismos roles candidatos a cada argumento, ordenados de acuerdo a su preferencia o nivel de confianza.
- Para acotar los roles candidatos con que los que se construirá las matrices, se debería utilizar como parámetros de selección el verbo y su sentido.
- Los niveles de confianza deben deducirse del nivel de precisión, *recall* o *F – Measure* de cada sistema, ya que estas medidas proveen información de la aptitud general de los mismos.





Parte IV

DESARROLLO Y EXPERIMENTACIÓN



En este capítulo revisamos los aspectos más relevantes del diseño e implementación de la solución que se plantea para combinar sistemas de ERS, todo ello bajo un modelo inspirado en esquemas de votación Borda.

### 10.1 METODOLOGIAS DE ANALISIS

Como se ha analizado en el [Capítulo 9](#), para aplicar un esquema de votación Borda, ya sea en la versión clásica o en la variante Fuzzy, se requiere que cada sistema de etiquetado provea un número determinado de roles candidatos para los argumentos de una oración. En virtud de que el corpus con el que se ha trabajado no provee dichos candidatos, se ha llevado a cabo un análisis que ha permitido establecer dos medidas de concordancia para realizar la combinación de dos o más sistemas de etiquetado. Estas medidas son el cálculo del *nivel de solapamiento* y el *análisis puntuado de verbos* y es importante destacar que trabajan de forma conjunta, es decir, la una complementa a la otra.

#### 10.1.1 Nivel de solapamiento entre roles candidatos

Esta medida nos permite analizar el nivel de coincidencia que tienen dos o más sistemas de etiquetado de roles. Mientras más alto sea este valor, en mayor medida coincidirá el criterio de dos o más sistemas. Dicha medida permite seleccionar los dos sistemas que tengan mayor valor de coincidencia, para que luego se pueda tomar los roles candidatos propuestos por el sistema con mayor puntuación en el análisis de verbos.

Para ejemplificar la forma en que calculamos este valor, tomamos la oración “‘*It screwed things up,*’ *said one major specialist.*” del corpus tWSJ y los roles candidatos que han propuesto los sistemas Punyakanok y Haghghi. Como se puede observar en la [Tabla 20](#), para el análisis del verbo *screwed*, el sistema Haghghi asigna el rol A<sub>0</sub> al constituyente *It*, mientras que el sistema Punyakanok no lo hace. Para el constituyente *things*, ambos sistemas coinciden en asignarle el rol A<sub>1</sub>. En este primer caso tendríamos que existe el solape en un solo argumento. Para el verbo *said* se produce un solape parcial en el rol A<sub>1</sub>, ya que para el primer sistema el argumento está formado por los constituyentes *It screwed things up*, mientras que para el segundo sistema se conforma de *screwed things up*. Para el rol A<sub>0</sub> ambos sistemas asignan los mismos constituyentes. En este segundo caso tendríamos un solape parcial (A<sub>1</sub>) y uno completo (A<sub>0</sub>).

Para calcular el nivel de solape total que existe entre los roles candidatos anotados por los sistemas, se debe considerar aquellos argumentos que se conforman por un solo constituyente y aquellos que tienen

| Constituyente | Punyakankok   |            | Haghighi      |            |
|---------------|---------------|------------|---------------|------------|
|               | Verbo screwed | Verbo said | Verbo screwed | Verbo said |
| "             | *             | *          | *             | *          |
| It            | *             | (A1*       | (Ao*)         | *          |
| screwed       | (V*)          | *          | (V*)          | (A1*       |
| things        | (A1*)         | *          | (A1*)         | *          |
| up            | *             | *)         | *             | *)         |
| ,             | *             | *          | *             | *          |
| "             | *             | *          | *             | *          |
| said          | *             | (V*)       | *             | (V*)       |
| one           | *             | (Ao*       | *             | (Ao*       |
| major         | *             | *          | *             | *          |
| specialist    | *             | *)         | *             | *)         |
| .             | *             | *          | *             | *          |

Tabla 20: Oración del corpus tWSJ, etiquetada por los sistemas Punyakankok y Haghighi.

más de un constituyente. Para obtener este valor hemos aplicado la [Ecuación 10.1](#):

$$\text{Solape}_{\text{oracion-S1-S2}} = \sum_{i=1}^m \left( \alpha \cdot \sum_{j=1}^p S_{S1}^j \wedge S_{S2}^j + \beta \cdot \sum_{j=1}^q C_{S1}^j \wedge C_{S2}^j \right) \quad (10.1)$$

donde:

- $S_{S1}^j$  y  $S_{S2}^j$  representan aquellos argumentos del sistema  $S_1$  y  $S_2$  que se conforman de más de un constituyente y se solapan.
- $C_{S1}^j$  y  $C_{S2}^j$  representan aquellos argumentos del sistema  $S_1$  y  $S_2$  que se conforman de un único constituyente y se solapan.
- $m$  es el número de argumentos que tiene el verbo del cual se anotan los roles.
- $p$  es el número de constituyentes que se solapan en los dos sistemas, que pertenecen a los argumentos  $S_{S1}$  y  $S_{S2}$ .
- $q$  es el número de constituyentes que se solapan en los dos sistemas, que pertenecen a los argumentos  $C_{S1}$  y  $C_{S2}$ .

Para el verbo *screwed*, tendríamos un nivel de solape igual a 1, en virtud de que solo se solapan en los argumentos A1. En el caso de los solapes parciales, debemos considerar cuántos constituyentes se solapan de un

argumento y cuántos constituyentes conforman ese argumento. Para calcular este valor hemos deducido la siguiente fórmula<sup>1</sup>:

$$\text{Solape}_{\text{parcial}} = S_{S1} \wedge S_{S2} = \alpha \cdot \sum_{i=1}^n \sum_{j=1}^m \left( \frac{CS_{S1}R_j^i}{CF_{S1}R_j^i} \right) \cdot \left( \frac{CS_{S2}R_j^i}{CF_{S2}R_j^i} \right) \quad (10.2)$$

donde:

- $CS_{S1}R_j^i$  y  $CS_{S2}R_j^i$  representan el número de constituyentes del rol R que se solapan entre los dos sistemas. El valor j indica el número de argumento del verbo, mientras que i especifica el verbo de la oración.
- $CF_{S1}R_j^i$  y  $CF_{S2}R_j^i$  representan el número de constituyentes que conforman el rol R anotado por el sistema  $S_1$  y el sistema  $S_2$ , respectivamente.
- m es el número de argumentos que tiene el verbo del cual se anotan los roles.
- n es el número de verbos que posee la oración.
- $\alpha$  es el parámetro de ajuste para el caso del solape parcial.

Como se puede notar, el número de constituyentes que se solapan entre los argumentos anotados por el sistema  $S_1$  y el sistema  $S_2$  ( $CS_{S1}R_j^i$  y  $CS_{S2}R_j^i$ ) siempre es el mismo, por lo que se puede reescribir la [Ecuación 10.2](#) de la siguiente forma:

$$\text{Solape}_{\text{parcial}} = \alpha \cdot \sum_{i=1}^n \sum_{j=1}^m \left( \frac{(CSR_j^i)^2}{CF_{S1}R_j^i \cdot CF_{S2}R_j^i} \right) \quad (10.3)$$

Para calcular el nivel de solape parcial de los argumentos del verbo *said*, tendríamos que sumar los solapes parciales que se producen entre los dos sistemas en el rol  $A1^2$ :

$$\text{Solape}_{\text{parcial}} = \text{solape}_{\text{parcialRol-A1}}$$

$$\text{Solape}_{\text{parcial}} = \left[ \left( \frac{3}{4} \right) \cdot \left( \frac{3}{3} \right) \right] = 0,75$$

El nivel de solape total que existe en el ejemplo expuesto en la [Tabla 20](#) se obtiene de la siguiente forma:

<sup>1</sup> Es importante tomar en cuenta que la fórmula que aquí se propone considera el solapamiento entre dos sistemas a la vez; por ello se especifica a que sistema corresponde cada constituyente ( $S_1$  o  $S_2$ ).

<sup>2</sup> Es importante observar que en el caso del rol  $A_0$  se produce un solape completo. De igual forma, destacamos que para los ejemplos aquí descritos hemos trabajado con un valor de  $\alpha = 1,0$ , ya que en la sección de experimentación se analiza el impacto que produce el cambio de dicho valor (dentro del rango de 0.1 a 1.0).

$$\text{Solape}_{\text{verbo-said}} = 1$$

$$\text{Solape}_{\text{verbo-screwed}} = \alpha \cdot \sum_{j=1}^3 S_{S_1}^j \wedge S_{S_2}^j + \left[ \left( \frac{3}{4} \right) \cdot \left( \frac{3}{3} \right) \right] = 3 + 0,75 = 3,75$$

De forma análoga a la regla de votación Borda, el nivel de solapamiento nos permite seleccionar los roles candidatos más votados, tomando como base la proporción de los argumentos etiquetados que coinciden entre los diferentes sistemas. En la [Figura 14](#) se puede observar un ejemplo completo que recoge los valores de solapamiento para la oración que analizamos anteriormente.

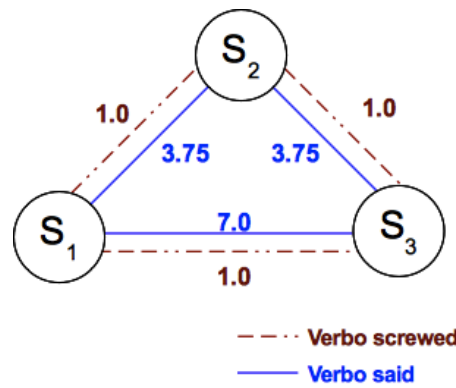


Figura 14: Ejemplo de los valores de solapamiento calculados para la combinación de sistemas S<sub>1</sub> - S<sub>2</sub> - S<sub>3</sub>. Se puede apreciar que los 3 sistemas coinciden en los roles candidatos para el caso del verbo *screwed*, mientras que para el verbo *said* existe un mayor valor de solapamiento entre los sistemas S<sub>1</sub> y S<sub>3</sub>.

### 10.1.2 Análisis puntuado de verbos

El análisis puntuado de verbos es una medida complementaria al nivel de solapamiento, ya que permite establecer los niveles de confianza con los que cada sistema asigna roles a los argumentos de la oración. De forma análoga, permite calcular un valor similar a los pesos del esquema de votación Fuzzy - Borda. Estos pesos o niveles de confianza se obtienen a partir de la precisión, *recall* o *F-Measure* con que cada sistema etiqueta cada tipo de rol. Por ejemplo, el sistema Punyakanok posee una precisión de 88.22%, *recall* de 87.88% y *F-Measure* de 88.05 cuando etiqueta el rol Ao.

Una vez que se han seleccionado los dos sistemas con el mayor valor de solapamiento, se deberá escoger aquel que posea las mejores cualidades de etiquetado. Para ilustrar correctamente esta idea, tomaremos como ejemplo la [Tabla 21](#). Como se puede apreciar, se han calculado las puntuaciones para los dos verbos que posee la oración, esto es, *screwed*

y *said*. Para obtener las puntuaciones de cada verbo se ha aplicado un simple fórmula de promedio ponderado:

$$\text{Puntuacion}_{\text{verbo}} = p \cdot \frac{1}{m} \sum_{i=1}^m \text{PuntuacionRol}_i \quad (10.4)$$

donde:

- $p$  indica la precisión general del sistema para realizar el etiquetado de roles<sup>3</sup>.
- $\text{PuntuacionRol}_i$  indica el nivel de precisión, *recall* o *F-Measure* con que etiqueta el sistema a cada argumento.
- $m$  indica el número total de roles etiquetados por verbo.

Las puntuaciones de los verbos anotados por el sistema Punyakanok se obtendrían de la siguiente forma:  $\text{Puntuacion}_{\text{screwed}} = 0,8225$  y  $\text{Puntuacion}_{\text{said}} = \frac{1}{2}(0,8225 + 0,8822) = 0,8523^4$ , para el caso del promedio simple. De igual manera, si queremos calcular las puntuaciones con promedio ponderado, simplemente se deberían multiplicar los valores obtenidos por el nivel de precisión de cada sistema:  $\text{Puntuacion}_{\text{screwed}} = 0,8228 \cdot 0,8225 = 0,6767$  y  $\text{Puntuacion}_{\text{said}} = 0,8228 \cdot \frac{1}{2}(0,8225 + 0,8822) = 0,701$ .

| Constituyente           | Punyakanok    |            | Haghighi      |            |
|-------------------------|---------------|------------|---------------|------------|
|                         | Verbo screwed | Verbo said | Verbo screwed | Verbo said |
| "                       | *             | *          | *             | *          |
| It                      | *             | (A1*)      | (Ao*)         | *          |
| screwed                 | (V*)          | *          | (V*)          | (A1*)      |
| things                  | (A1*)         | *          | (A1*)         | *          |
| up                      | *             | *)         | *             | *)         |
| ,                       | *             | *          | *             | *          |
| "                       | *             | *          | *             | *          |
| said                    | *             | (V*)       | *             | (V*)       |
| one                     | *             | (Ao*)      | *             | (Ao*)      |
| major                   | *             | *          | *             | *          |
| specialist              | *             | *)         | *             | *)         |
| .                       | *             | *          | *             | *          |
| <i>Puntuación-Verbo</i> | 0.8225        | 0.8523     | 0.83465       | 0.83465    |

Tabla 21: Oración del corpus tWSJ, etiquetada por los sistemas Punyakanok y Haghighi. Se incluyen los valores que genera el análisis puntuado de verbos.

<sup>3</sup> Para la experimentación que se ha llevado a cabo se ha utilizado promedio simple y promedio ponderado, es decir, con  $p$  igual a 1 y con  $p$  igual a la precisión de etiquetado general de cada sistema.

<sup>4</sup> En los anexos se pueden observar los valores de precisión, *recall* y *F-Measure* que poseen los 5 mejores sistemas en el etiquetado de cada tipo de rol.

## 10.2 DISEÑO DEL SISTEMA

En esta sección se describe el esquema general del sistema, así como los diferentes componentes que lo conforman y la función que realiza cada uno de ellos.

## 10.2.1 Estructura general

Como se puede apreciar en la Figura 15, la propuesta de combinación que se plantea en este trabajo toma los resultados generados por los mejores sistemas de la competición CoNLL 2005 (corpus tWSJ y tBrown<sup>5</sup>) y los combina en una sola salida. Para realizar dicha combinación nuestro sistema se basa en ciertos parámetros de configuración, como el nivel de solapamiento, si se emplea o no análisis puntuado de verbos, los factores de cálculo alfa ( $\alpha$ ) y beta ( $\beta$ ), etc. Todos estos parámetros permiten discernir que medidas de análisis son las más adecuadas y como influye la inclusión de uno o más sistemas en la combinación.

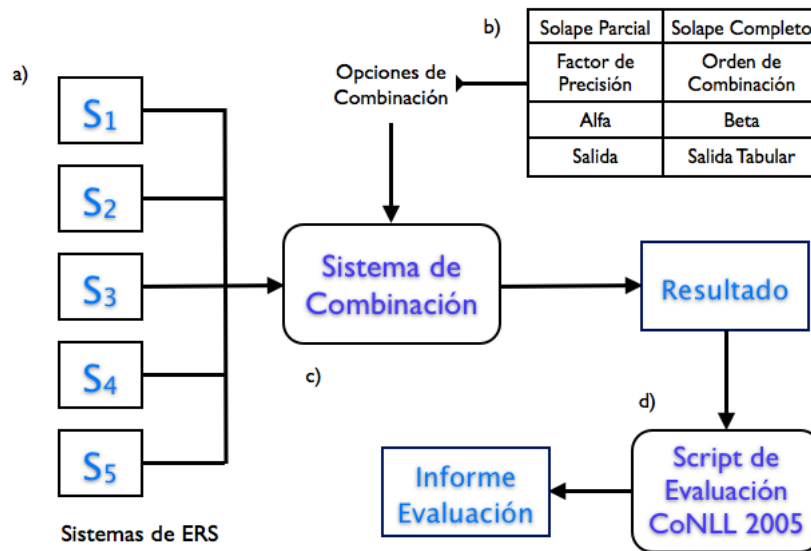


Figura 15: Esquema general de funcionamiento de la propuesta de combinación realizada. Se pueden observar 4 elementos fundamentales: a) sistemas participantes ( $S_1$  a  $S_5$ ), b) opciones de combinación, c) sistema que realiza la combinación y d) *script* de evaluación de la competición CoNLL 2005.

## 10.2.2 Estructura modular

El sistema que se encarga de llevar a cabo el proceso de combinación se compone de los siguientes elementos (Figura 16):

- Demonio de ejecución . - Se encarga de generar la planificación de las combinaciones que se realizarán, así como los parámetros que se aplicarán. Por ejemplo: combinación de los sistemas  $S_1 - S_2$  con factor de precisión, análisis puntuado de verbos y solape parcial. Una vez que han establecido las combinaciones que se aplicarán,

<sup>5</sup> Es importante observar que el proceso de combinación se realiza dos veces, en primera instancia para el corpus tBrown y luego para el tWSJ.



se encarga de invocar al módulo de análisis y combinación para ejecutar la planificación generada.

- Módulo de análisis y combinación . - Es el elemento más importante del sistema, ya que se encarga de realizar las tareas fundamentales del proceso de combinación: lectura del corpus de entrada, cálculo del nivel de solapamiento, generación de las matrices de puntuación, creación de los ficheros de análisis parcial e invocación a los módulos de soporte.
- Módulo de índices y promedios . - Su tarea consiste en leer los ficheros generados por el módulo de análisis y combinación para realizar el cálculo de los promedios (análisis puntuado de verbos). A más de esto, se encarga de reorganizar dichos ficheros a través de la creación de índices de acceso.
- Módulo de solape parcial . - Este módulo permite calcular los rangos del solape parcial que se producen entre los constituyentes de los argumentos.
- Enlazador de evaluación . - Este elemento realiza la invocación al *script* de evaluación de la competición CoNLL 2005 y luego almacena este resultado en un fichero.
- Módulo de formateo de resultados . - Cambia el formato de salida que genera el *script* de evaluación, de modo que se pueda realizar un tabulado posterior con estos datos.

Una vez que se han detallado los diferentes componentes del sistema, pasamos a describir los parámetros de combinación que se pueden emplear en el momento de ejecutar el proceso:

- Tipo de Solape . - Indica si se calcula el valor de solapamiento completo o parcial. En el solape parcial puntúan todos los roles, a pesar de que no todos sus constituyentes se solapen, mientras que en el solape completo se descartan si no se solapan todos los constituyentes.
- $\alpha$ . - Parámetro de ajuste que permite ponderar a los argumentos conformados por más de un constituyente.
- $\beta$ . - Es un parámetro de ajuste que permite ponderar a los argumentos conformados por un solo constituyente.
- Orden . - Especifica que sistemas se combinarán ( $S_1 - S_2$ ,  $S_1 - S_3 - S_5$ , etc.)
- Factor de Puntuación . - Tipo de puntuación que se utilizará para el caso del análisis de verbos ponderado (precisión, *recall* o *F-Measure*).
- Factor de Multiplicación . - Permite indicar si se aplica el factor de puntuación.

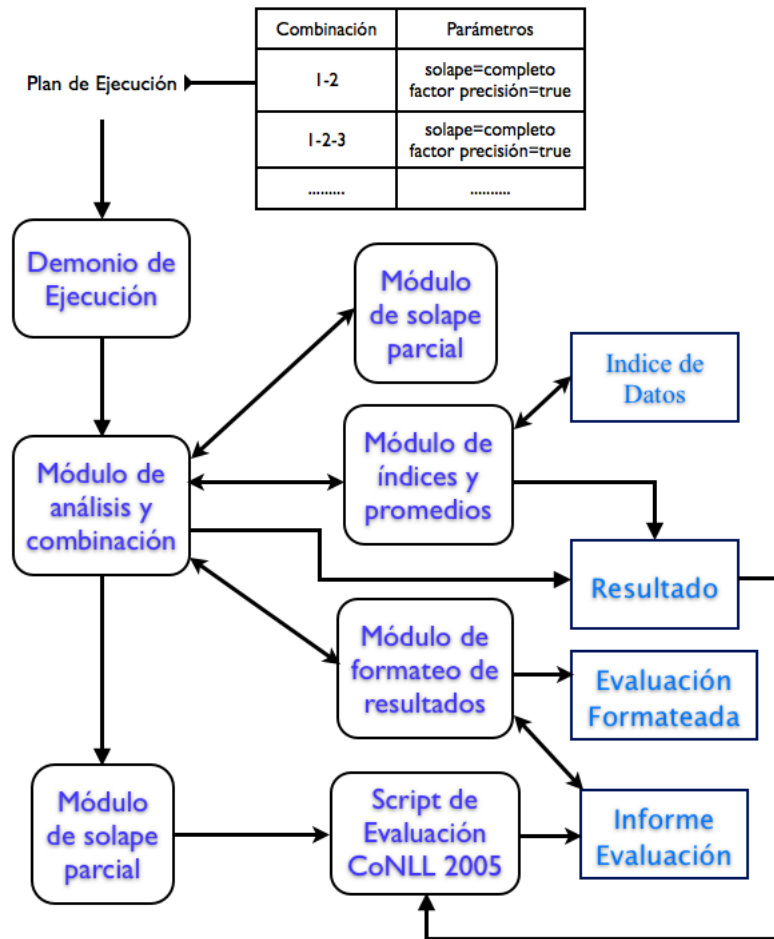


Figura 16: Esquema modular de la propuesta de combinación de los sistemas de ERS. Como se aprecia, para combinación y conjunto de parámetros se generará un resultado diferente.

## EXPERIMENTACIÓN Y RESULTADOS

A fin de llevar a cabo una experimentación más completa<sup>1</sup>, se ha trabajado con los dos corpus de pruebas de la competición (tWSJ y tBrown). Con cada corpus se ha realizado un total de 40 experimentaciones diferentes para el caso de los dos tipos de solapamiento y análisis puntuado de verbos. En la [Tabla 22](#) podemos observar un ejemplo de la lógica de combinación que se ha seguido durante la etapa de experimentación.

| Combinación                   | Solapamiento | Análisis puntuado       |
|-------------------------------|--------------|-------------------------|
| $S_1 - S_2$                   | Parcial      | Con factor de precisión |
| $S_1 - S_2$                   | Parcial      | Sin factor de precisión |
| $S_1 - S_2$                   | Completo     | Con factor de precisión |
| $S_1 - S_2$                   | Completo     | Sin factor de precisión |
| $S_1 - S_3$                   | Parcial      | Con factor de precisión |
| $S_1 - S_3$                   | Parcial      | Sin factor de precisión |
| $S_1 - S_3$                   | Completo     | Con factor de precisión |
| $S_1 - S_3$                   | Completo     | Sin factor de precisión |
| .....                         | .....        | .....                   |
| $S_1 - S_2 - S_3 - S_4 - S_5$ | Parcial      | Con factor de precisión |

Tabla 22: Oración del corpus tWSJ, etiquetada por los sistemas Punyakanok y Haghghi.

De igual forma, se seleccionaron las 3 mejores combinaciones de sistemas para el corpus tBrown ( $S_1 - S_4$ ,  $S_1 - S_5$  y  $S_1 - S_3 - S_4$ ) y con ellas se realizaron 108 experimentaciones diferentes, variando para ello los parámetros de ajuste  $\alpha$  y  $\beta$ . En la [Tabla 23](#) se ejemplifica el tipo de combinaciones que se ha realizado.

| Combinación | Solapamiento | alfa  | beta  |
|-------------|--------------|-------|-------|
| $S_1 - S_4$ | Parcial      | 0.1   | 0.1   |
| $S_1 - S_4$ | Parcial      | 0.1   | 0.3   |
| .....       | .....        | ..... | ..... |
| $S_1 - S_4$ | Parcial      | 0.3   | 0.1   |
| .....       | .....        | ..... | ..... |
| $S_1 - S_4$ | Parcial      | 1.0   | 1.0   |

Tabla 23: Ejemplo de combinaciones ejecutadas con variación de los parámetros  $\alpha$  y  $\beta$ . De igual forma se procede para las combinaciones de sistemas  $S_1 - S_5$  y  $S_1 - S_3 - S_4$ .

<sup>1</sup> En los anexos se pueden encontrar los resultados completos de la experimentación, donde se detallan los valores de precisión, *recall*, *F-Measure* de todas las combinaciones de sistemas.

## 11.1 RESULTADOS CORPUS TWSJ

Los resultados que se han obtenido con el corpus de pruebas tWSJ se describen en esta sección desde dos perspectivas: una visión general, que considera los valores promedio y una visión específica, que reve los valores puntuales de las mejores combinaciones de sistemas.

*Valores medios*

Como se puede observar en la [Figura 17](#), la combinación de sistemas, tanto en promedio general, como individual, permite obtener un mayor número de roles anotados correctamente. Los mejores resultados se obtienen con la combinación de los sistemas  $S_1 - S_3 - S_4$ , cuyo valor supera a todos los demás sistemas y combinaciones.

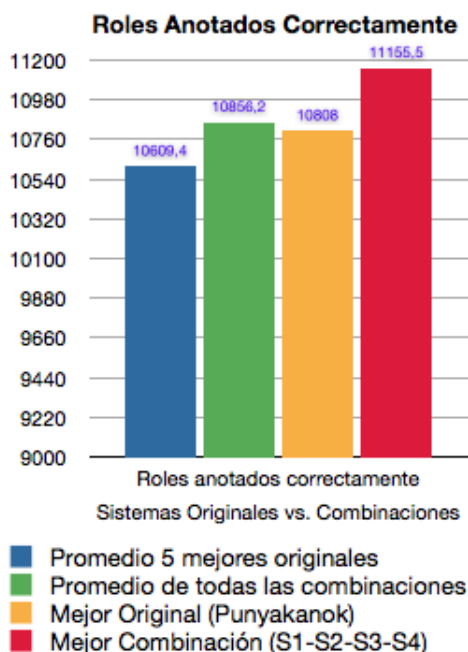


Figura 17: Roles anotados correctamente, valores promedio generales (corpus tWSJ). Se puede apreciar una comparativa entre los valores promedio de los mejores sistemas originales, todas las combinaciones, el mejor sistema original y la mejor combinación.

En la [Figura 18](#) se puede apreciar el número de roles que sobreclasifican tanto los sistemas originales como las combinaciones realizadas. Es importante tomar en consideración que la mejor combinación de sistemas ( $S_1 - S_3 - S_4$ ) omite menos roles, pero es la segunda en sobreclasificar la mayor cantidad de ellos.

En cuanto a los valores precisión, *recall* y *F-Measure* ([Figura 19](#)), podemos anotar lo siguiente: respecto a la precisión, el mejor sistema original supera en tres décimas a la mejor combinación de sistemas  $S_1 - S_3 - S_4$ ; en *recall* y *F-Measure* las combinaciones de sistemas superan de forma clara los sistemas originales. Es importante anotar que existen otras combinaciones de sistemas que superan en dos y tres décimas al mejor sistema original, sin embargo, no se las ha seleccionado debido a que la diferencia de valores no es muy amplia.

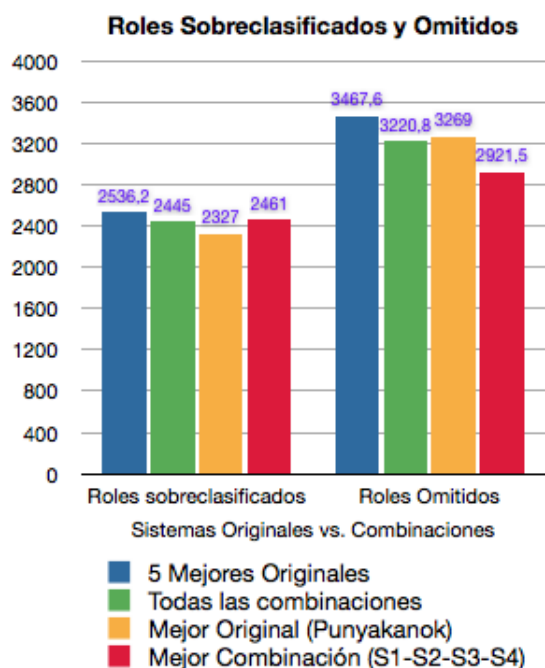


Figura 18: Roles sobreclasificados y omitidos, valores promedio generales (corpus tWSJ). Se observa una comparativa entre los valores promedio de los mejores sistemas originales, todas las combinaciones, el mejor sistema original y la mejor combinación.

#### Valores específicos

La Figura 20 recoge el número de roles anotados de forma correcta por los 5 mejores sistemas de la competición y las 4 mejores combinaciones. Se puede observar que la combinación de los sistemas  $S_1 - S_2 - S_3 - S_4 - S_5$  con factor de promedio ponderado y solape parcial obtiene el valor más alto. Esto se debe a que el factor de promedio ponderado (precisión) ayuda a elegir de mejor manera los roles candidatos.

En cuanto a los roles sobreclasificados y omitidos (Figura 21), la combinación que produce valores más bajos en conjunto, es la que se conforma por los sistemas  $S_1 - S_3$  y utiliza solape parcial sin factor de promedio ponderado. Es importante notar que las demás combinaciones tienen valores en conjunto más bajos que los mejores sistemas originales.

El último aspecto a tener en cuenta es que valores de precisión, *recall* y *F-Measure* alcanzan las diferentes combinaciones. Como se puede apreciar en la Figura 22, las combinaciones que alcanzan los máximos valores y permiten incrementar la precisión, el *recall* y por ende el *F-Measure*, son las que se conforman por los sistemas  $S_1 - S_3$  y  $S_1 - S_4$ . De igual manera, otro aspecto interesante que se debe tener presente, es que la combinación de dos sistemas produce mejores resultados que cada uno de ellos por separado.

## 11.2 RESULTADOS CORPUS TBROWN

Al igual que en el caso de la sección anterior, los resultados del corpus tBrown se revisan de forma general y específica.

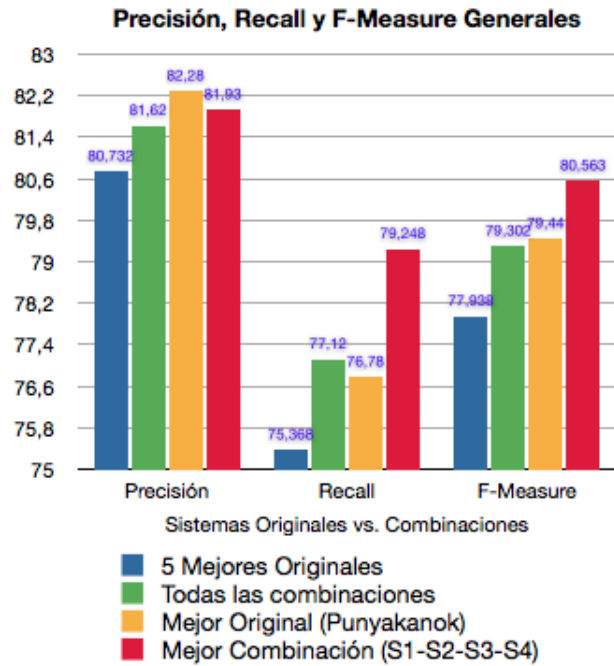


Figura 19: Precisión, *recall* y *F-Measure* de los mejores sistemas originales y combinaciones, valores promedio generales (corpus tWSJ). Se aprecia que el mejor sistema original posee el mayor valor de precisión.

#### Valores medios

Como se puede observar en la [Figura 23](#), las combinaciones de sistemas anotan de forma correcta una mayor cantidad de roles, respecto a los sistemas originales. La mejor combinación es la que se conforma de los sistemas  $S_1 - S_4$ .

Si analizamos el caso de los roles sobreclasificados y omitidos ([Figura 24](#)), podremos notar que las combinaciones tienen valores inferiores a los sistemas originales. Para este caso, nuevamente la combinación que posee los valores promedio más bajos es la que se conforma por los sistemas  $S_1 - S_4$ .

De igual forma, si se revisan los valores de precisión, *recall* y *F-Measure* ([Figura 25](#)), podremos notar que las combinaciones superan a los sistemas individuales. Para el caso del *recall* y el *F-Measure* se puede observar se tiene una diferencia que supera el punto porcentual.

#### Valores específicos

Como se observa en la [Figura 26](#), la combinación que alcanza el valor máximo de roles anotados correctamente, es la que se conforma de los sistemas  $S_1 - S_2 - S_3 - S_4 - S_5$  con factor de promedio ponderado y solape parcial. De igual forma, es importante tener presente que la combinación de los sistemas  $S_1 - S_2 - S_3$  sin factor de promedio ponderado y con solape parcial, anota correctamente más roles que los sistemas individuales.

En cuanto a roles sobreclasificados y omitidos ([Figura 27](#)), la combinación que produce valores más bajos (en conjunto), es la que se conforma de los sistemas  $S_1 - S_4$  sin factor de promedio ponderado y

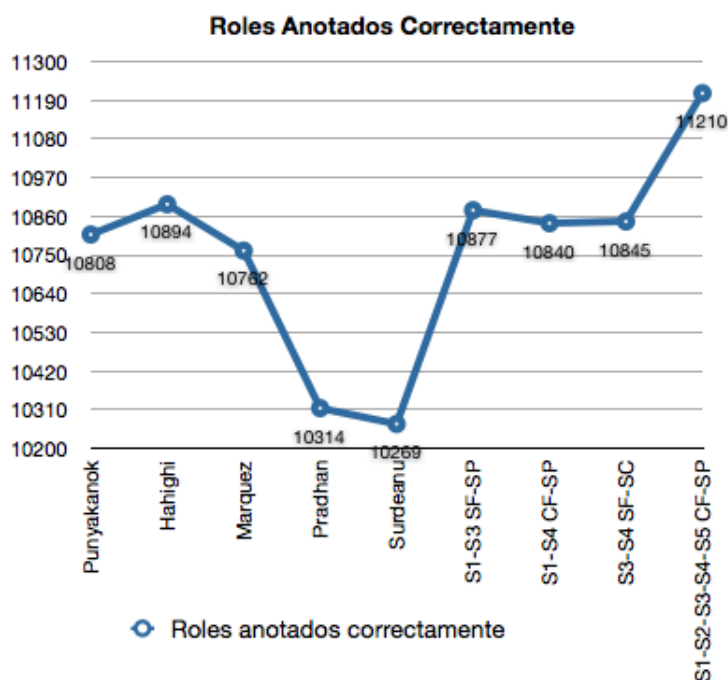


Figura 20: Roles anotados correctamente, valores puntuales (corpus tWSJ). Se puede apreciar que la combinación  $S_1 - S_2 - S_3 - S_4 - S_5$  posee el índice más alto de roles anotados correctamente. El significado de las abreviaturas es el siguiente: SF-SP=Sin Factor de promedio ponderado con Solape Parcial, SF-SC=Sin Factor de promedio ponderado con Solape Completo y CF-SP=Con Factor de promedio ponderado con Solape Parcial.

con solape parcial. Para este caso, se debe notar que las combinaciones obtienen valores más bajos que los sistemas originales.

Asimismo, si observamos la [Figura 28](#), podremos constatar que las combinaciones alcanzan niveles mayores de precisión (alrededor de un punto porcentual), recall (alrededor de dos puntos porcentuales) y F-Measure (alrededor de dos puntos porcentuales).

### 11.3 RESULTADOS CORPUS TBROWN PARA PARÁMETROS $\alpha$ Y $\beta$

En esta sección recogemos los experimentos realizados respecto a los parámetros de ajuste  $\alpha$  y  $\beta$ . Como se ha visto anteriormente, el parámetro  $\beta$  permite asignar un peso a los argumentos que poseen un solo constituyente. De igual forma, el parámetro  $\alpha$  se encarga de ponderar a los argumentos que se conforman de más de un constituyente. La experimentación llevada a cabo ha demostrado que ciertos valores de  $\alpha$  y  $\beta$  permiten mejorar el desempeño en algunas de las combinaciones de sistemas. Como se puede apreciar en la [Figura 29](#), existen ciertas valores que posibilitan aumentar ligeramente el valor de precisión (por ejemplo,  $\alpha = 1,0$  y  $\beta = 0,1$ ).

En el caso del *recall* y del *F-Measure* ([Figura 30](#) y [Figura 31](#)), podemos observar que se produce un efecto similar, generándose ligeras mejoras en el desempeño con ciertas combinaciones de valores. Un aspecto que reviste interés, es que los parámetros de ajuste afectan de forma similar a los tres valores (precisión, *recall* y *F-Measure*).

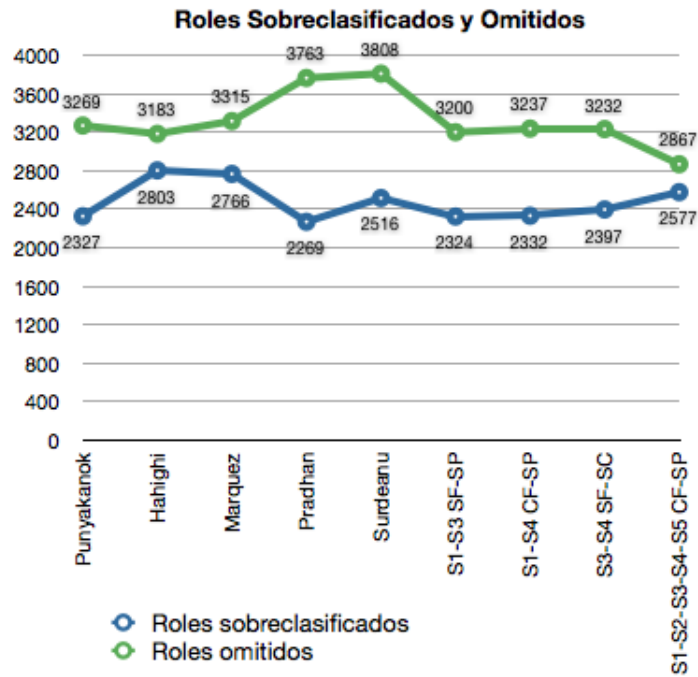


Figura 21: Roles sobreclasificados y omitidos, valores puntuales (corpus tWSJ). Como se aprecia, la combinación que posee los valores más bajos es la que se conforma de los sistemas  $S_1 - S_3$  sin factor de promedio ponderado.

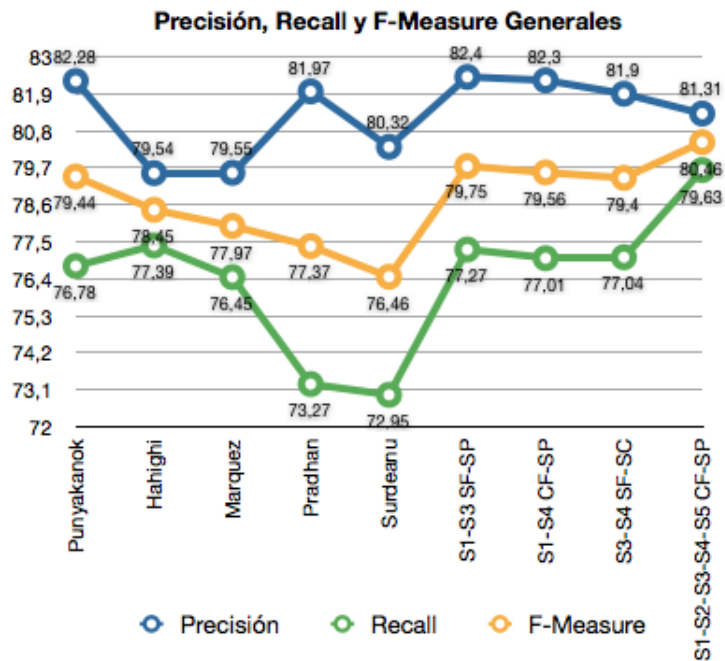


Figura 22: Precisión, *recall* y *F-Measure* de los mejores sistemas originales y combinaciones, valores puntuales (corpus tWSJ). La combinación que logra obtener valores más altos de precisión, recall y por ende, *F-Measure*, es  $S_1 - S_3$  sin factor de promedio ponderado.



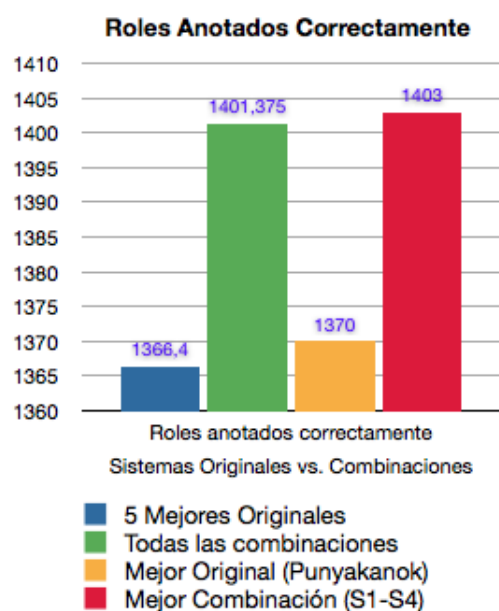


Figura 23: Roles anotados correctamente, valores promedio generales (corpus tBrown). Se puede apreciar una comparativa entre los valores promedio de los mejores sistemas originales, todas las combinaciones, el mejor sistema original y la mejor combinación.

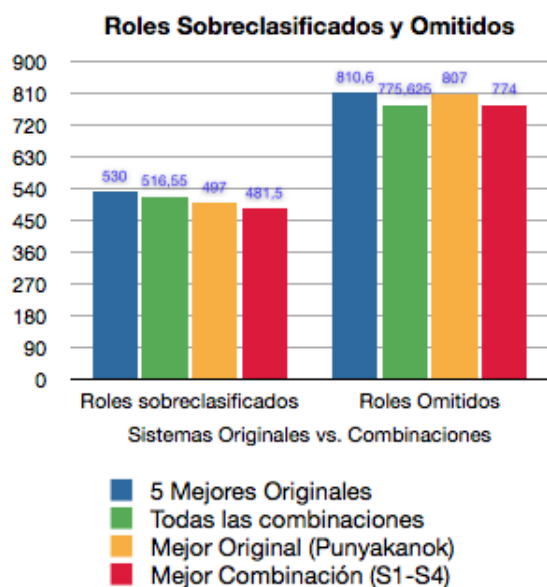


Figura 24: Roles sobreclasificados y omitidos, valores promedio generales (corpus tBrown). Se observa una comparativa entre los valores promedio de los mejores sistemas originales, todas las combinaciones, el mejor sistema original y la mejor combinación.

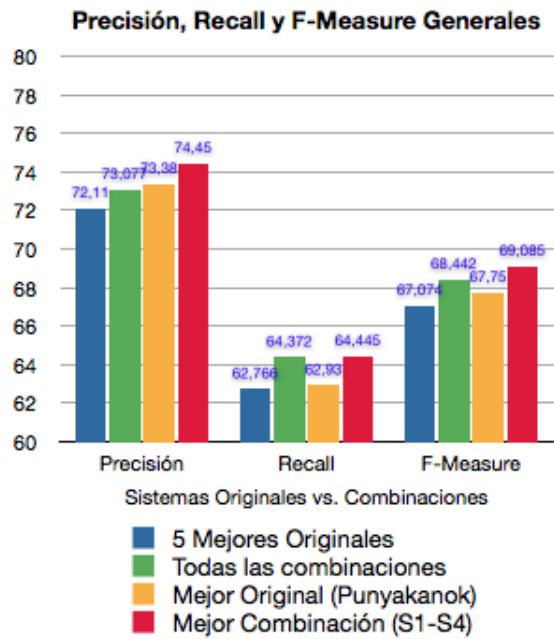


Figura 25: Precisión, *recall* y *F-Measure* de los mejores sistemas originales y combinaciones, valores promedio generales (corpus tBrown). Se puede notar que tanto el valor medio de las combinaciones, como la mejor combinación son superiores a los sistemas originales.

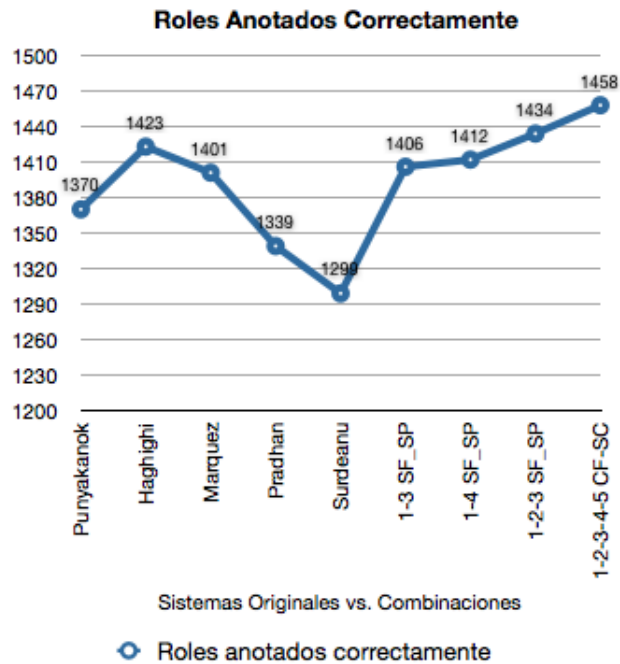


Figura 26: Roles anotados correctamente, valores puntuales (corpus tBrown). Como se aprecia, la combinación que alcanza el número de roles anotados correctamente es la que se conforma de los sistemas  $S_1 - S_2 - S_3 - S_4 - S_5$  y utiliza solape completo con factor de promedio ponderado. El significado de las abreviaturas es el siguiente: SF-SP=Sin Factor de promedio ponderado con Solape Parcial y CF-SC=Con Factor de promedio ponderado con Solape Completo.

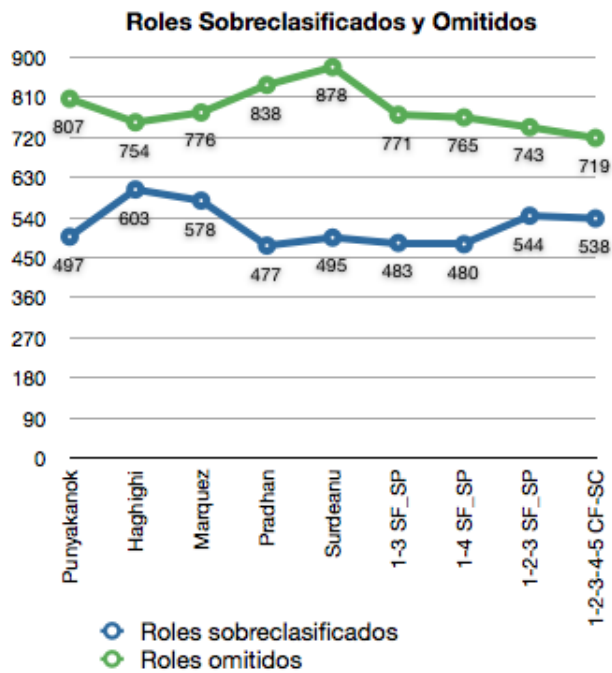


Figura 27: Roles sobreclasificados y omitidos, valores puntuales (corpus tBrown). Se puede apreciar que los valores conjuntos más bajos pertenecen a la combinación  $S_1 - S_4$  con solape parcial y sin factor de promedio ponderado.

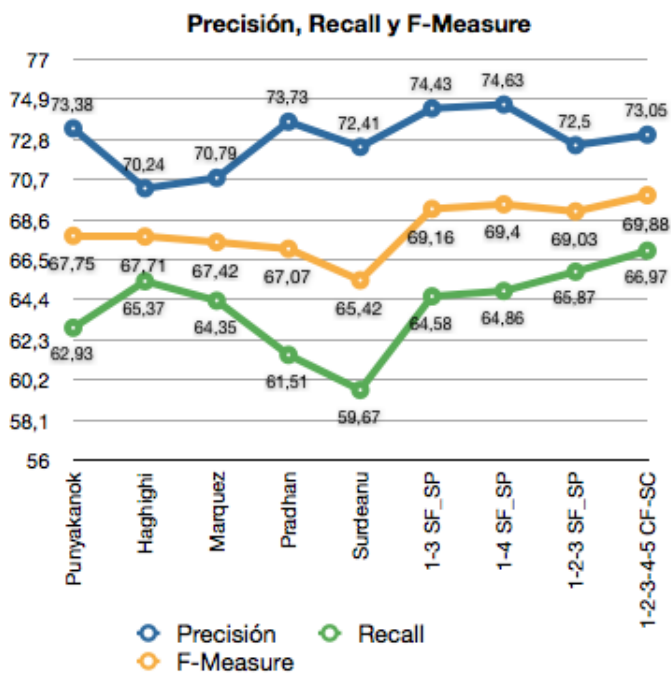


Figura 28: Precisión, recall y F-Measure de los mejores sistemas originales y combinaciones, valores específicos (corpus tBrown). Se puede observar que las combinaciones obtienen mejores valores que los sistemas originales. Para el caso del F-Measure se alcanzan diferencias de 2 puntos.

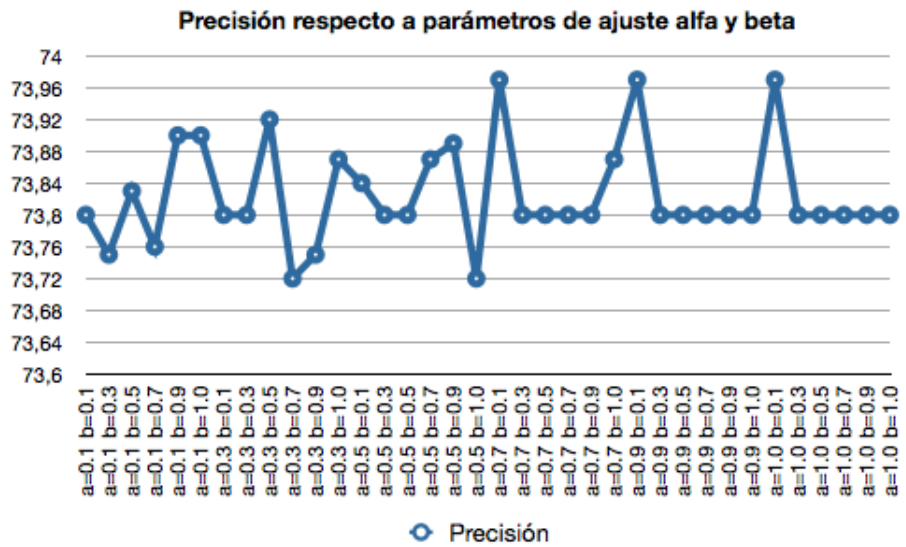


Figura 29: Variación de la precisión en el combinado de sistemas, de acuerdo a los parámetros de ajuste. La ilustración corresponde a la combinación de los sistemas  $S_1 - S_3 - S_4$  sin factor de promedio ponderado y con solape parcial (corpus tBrown).



Figura 30: Variación del *recall* en el combinado de sistemas, de acuerdo a los parámetros de ajuste. La ilustración corresponde a la combinación de los sistemas  $S_1 - S_3 - S_4$  sin factor de promedio ponderado y con solape parcial (corpus tBrown).

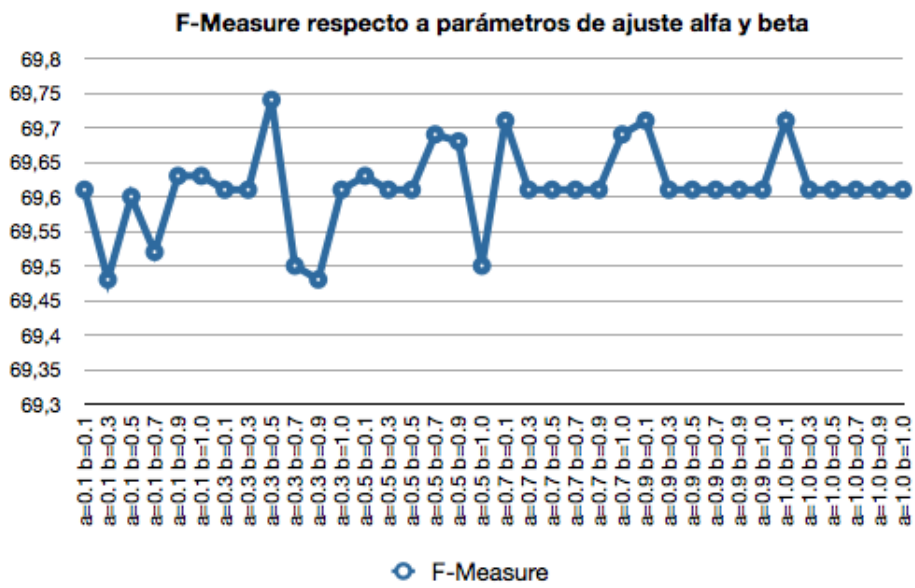


Figura 31: Variación del *F-Measure* en el combinado de sistemas, de acuerdo a los parámetros de ajuste. La ilustración corresponde a la combinación de los sistemas  $S_1 - S_3 - S_4$  sin factor de promedio ponderado y con solape parcial (corpus tBrown).



Parte V

CONCLUSIONES Y TRABAJO FUTURO





## CONCLUSIONES Y TRABAJO FUTURO

Con la realización de este trabajo, se ha podido comprobar que cuando no se cuenta con el número suficiente de roles candidatos para construir las matrices de votación Borda, esto es, al menos 3, es posible usar medidas de solapamiento y análisis puntuado de verbos para combinar dos o más sistemas de forma eficiente. Estas dos medidas que se introducen han permitido crear combinaciones de sistemas que posibilitan alcanzar valores de precisión, *recall*, y por ende, *F-Measure* más altos que los sistemas originales.

Asimismo, se ha comprobado con la experimentación realizada que si se combinan dos sistemas donde el primero tiene mayor precisión que el segundo, y por el contrario, posee menor *recall* que el segundo, la combinación resultante se acerca al mayor valor de precisión y *recall* de los dos sistemas. Por ejemplo, el sistema Marquez posee 77.95% de precisión y 76.45% de *recall*, mientras que el sistema Pradhan tiene 81.97% de precisión y 73.27% de *recall*, la combinación de los dos produce valores de 82.11% y 76.93%. En los casos en los que las combinaciones no obtienen valores mayores de estas dos medidas, se alcanza un valor de *F-Measure* superior.

Otro aspecto de interés es que los valores medios de las combinaciones que se han probado son superiores a los valores medios de los sistemas originales, para el caso de los dos corpus con que se ha trabajado (tWSJ y tBrown). De igual forma, se ha podido constatar que los parámetros de ajuste  $\alpha$  y  $\beta$  permiten mejorar los niveles de precisión, *recall* y por ende, de *F-Measure*, de las combinaciones de sistemas de ERS.

También se debe anotar que no todas las combinaciones alcanzan valores óptimos, ya que si dos sistemas poseen valores bajos de precisión o *recall*, no se podrá obtener mejoras en este aspecto.

Finalmente, es importante destacar que consideramos que los esquemas de votación Borda, tanto en la versión clásica como en su variante Fuzzy, se constituyen en un herramienta de gran utilidad en la combinación de sistemas de etiquetado de roles semánticos.

Como trabajo futuro se plantea lo siguiente:

- Incluir en el proceso de combinación información de FrameNet, lo que permitirá enriquecer el proceso y mejorar la eficiencia de la combinación resultante.
- Estudiar la posibilidad de aplicar HTM<sup>1</sup> para crear un nuevo modelo de aprendizaje de roles semánticos.

*... es posible usar medidas de solapamiento y análisis puntuado de verbos para combinar dos o más sistemas de forma eficiente.*

<sup>1</sup> HTM son las siglas de Memoria Temporal Jerárquica (*Hierarchical Temporal Memory*), un nuevo paradigma de reconocimiento de patrones que se inspira en el funcionamiento del neocórtex del cerebro humano.



Part VI

REFERENCIAS Y ANEXOS



## BIBLIOGRAFÍA

---

### A

---

- [1] S. Abney. Parsing by chunks. *Principle-based parsing*, pages 257–278, 1991. (Cited on page 7.)
- [2] C. Armentano-Oller, A.M. Corbi-Bellot, M.L. Forcada, M. Ginesti-Rosell, M.A.M. Belda, S. Ortiz-Rojas, J.A. Pérez-Ortiz, G. Ramirez-Sánchez, and F. Sánchez-Martinez. *Apertium, una plataforma de código abierto para el desarrollo de sistemas de traducción automática*. Universidad Politécnica de Alicante, 2007. (Cited on page 8.)

### B

---

- [3] O. Babko-Malaya. PROPBANK ANNOTATION GUIDELINES. 2005. (Cited on page 12.)
- [4] C.A. Bahón. *Aprendizaje con máquinas núcleo en entornos de multi-clasificación*. PhD thesis, Universidad Politécnica de Cataluña, 2001. (Cited on page 25.)
- [5] D. Barreto et al. Principios y representaciones en sintaxis. *Verba*, (10):5, 1983. (Cited on page 8.)
- [6] Y. Benajiba, M. Diab, and P. Rosso. Arabic named entity recognition using optimized features sets. EMNLP, 2008. (Cited on page 55.)
- [7] A.L. Berger, V.J.D. Pietra, and S.A.D. Pietra. A maximum entropy approach to natural language processing. *Computational linguistics*, 22(1):39–71, 1996. (Cited on page 34.)
- [8] G.A. Betancourt. *Las máquinas de soporte vectorial (svms)*, 2005. (Cited on page 23.)
- [9] J. Bi and V.N. Vapnik. Learning with rigorous support vector machines. In *Learning theory and Kernel machines: 16th Annual Conference on Learning Theory and 7th Kernel Workshop, COLT/Kernel 2003, Washington, DC, USA, August 24-27, 2003: proceedings*, page 243. Springer Verlag, 2003. (Cited on page 23.)
- [10] J.C. Borda. Histoire de l'Académie Royale de Science. *Memoire sur les Elections au Scrutin, Paris*, 1781. (Cited on page 56.)
- [11] L. Bottou. Une approche théorique de l'apprentissage connexionniste: Applications a la reconnaissance de la parole. *These de doctorat, Université d'Orsay*, 1991. (Cited on page 22.)
- [12] E. Brill. Part-of-speech tagging. *Handbook of Natural Language Processing*, 1999.
- [13] Robert Bringhurst. *The Elements of Typographic Style*. Version 2.5. Hartley & Marks, Publishers, Point Roberts, WA, USA, 2002. (Cited on page 131.)

- [14] T. Briscoe. Prospects for practical parsing of unrestricted text: Robust statistical parsing techniques. *Corpus-based research into language: in honour of Jan Aarts*, page 97, 1994. (Cited on page 7.)
- [15] D. Buscaldi and P. Rosso. Upv-wsd: Combining different wsd methods by means of fuzzy borda voting. *Fourth International Workshop on Semantic Evaluations (Semeval-2007)*, pages 434–437, 2007. (Cited on page 55.)
- [16] D. Buscaldi, J. M. Perea, P. Rosso, L. A. Ureña, D. Ferrés, and H. Rodríguez. Geotexmess: Result fusion with fuzzy borda ranking in geographical information retrieval. *Revised Selected Papers CLEF-2008*, pages 867–874, 2008. (Cited on page 55.)

## C

---

- [17] X. Carreras and L. Marquez. Phrase recognition by filtering and ranking with perceptrons. *Recent advances in natural language processing III: selected papers from RANLP 2003*, page 205, 2004. (Cited on page 32.)
- [18] X. Carreras and L. Màrquez. Introduction to the CoNLL-2005 shared task: Semantic role labeling. In *Proceedings of the Ninth Conference on Computational Natural Language Learning*, pages 152–164. Association for Computational Linguistics, 2005. (Cited on pages xvii, 32 y 33.)
- [19] E. Charniak. A maximum-entropy-inspired parser. In *ANLP'00*, 2000. (Cited on page 32.)
- [20] H.L. Chieu and H.T. Ng. Named entity recognition with a maximum entropy approach. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, page 163. Association for Computational Linguistics, 2003. (Cited on page 32.)
- [21] Y. Choi, E. Breck, and C. Cardie. Joint extraction of entities and relations for opinion recognition. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 431–439. Association for Computational Linguistics, 2006. (Cited on pages 1 y 14.)
- [22] T. Cohn. and P. Blusom. Semantic role labelling with tree conditional random fields. *CoNLL*, 2005. (Cited on pages xiii, xvii, 37 y 39.)
- [23] M. Collins. Head-driven statistical models for natural language parsing. *Computational linguistics*, 29(4):589–637, 2003. (Cited on pages 32 y 44.)
- [24] J. Cowie and W. Lehnert. Information extraction. *Communications of the ACM*, 39(1):80–91, 1996. (Cited on page 9.)

## D

---

- [25] Y.B. Dibike, S. Velickov, D. Solomatine, and M.B. Abbott. Model induction with support vector machines: Introduction and applications. *Journal of Computing in Civil Engineering*, 15(3):208–216, 2001. (Cited on pages 23, 24 y 25.)

## F

---

- [26] Y. Freund and RE Schapire. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997. (Cited on page 27.)

## G

---

- [27] JF Gantz et al. IDC-The Expanding Digital Universe: A Forecast of Worldwide Information Growth Through 2010. Technical report, Technical report, March 2007. (Cited on page 1.)
- [28] J.L. García-Lapresta and M. Martínez-Panero. Borda count versus approval voting: A fuzzy approach. *Public Choice*, 112(1):167–184, 2002. (Cited on page 2.)
- [29] J.L. García-Lapresta and M. Martínez-Panero. Extensiones discretas de la regla de borda: un estudio comparativo. 2003. (Cited on page 55.)
- [30] D. Gildea and D. Jurafsky. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3):245–288, September 2002. (Cited on page 40.)
- [31] M. Gutiérrez and J. Pablo. Directed motion in english and spanish. *Estudios de Lingüística del Español*, (11), 2001. (Cited on page 11.)

## H

---

- [32] K. Hacioglu, S. Pradhan, W. Ward, J.H. Martin, and D. Jurafsky. Semantic role labeling by tagging syntactic chunks. In *Proceedings of CoNLL*, volume 4, 2004. (Cited on pages xviii y 41.)
- [33] J.M. Hammersley and P. Clifford. Markov fields on finite graphs and lattices. *Unpublished manuscript*, 3, 1971. (Cited on page 22.)
- [34] F. Harary. *Graph theory*. Addison-Wesley, Reading, MA, 1994. (Cited on page 38.)

## J

---

- [35] ET Jaynes. On the rationale of maximum-entropy methods. *Proceedings of the IEEE*, 70(9):939–952, 1982. (Cited on page 20.)

## K

---

- [36] M. Kaisser and B. Webber. Question answering based on semantic roles. In *Proceedings of the ACL*, volume 7, 2007. (Cited on page 1.)
- [37] S.M. Kim and E. Hovy. Extracting opinions, opinion holders, and topics expressed in online news media text. In *Proceedings of the Workshop on Sentiment and Subjectivity in Text*, pages 1–8. Association for Computational Linguistics, 2006. (Cited on pages [xiii](#), [1](#), [14](#) y [15](#).)
- [38] K. Kipper, A. Korhonen, N. Ryant, and M. Palmer. Extending VerbNet with novel verb classes. In *Proceedings of LREC*. Citeseer, 2006. (Cited on page [13](#).)
- [39] P. Koomen, V. Punyakanok, D. Roth, and W. Yih. Generalized inference with multiple semantic role labeling systems. In *Ninth Conference on Computational Natural Language Learning*, page 181, 2005. (Cited on pages [43](#) y [44](#).)
- [40] T. Kudoh and Y. Matsumoto. Use of support vector learning for chunk identification. In *Proceedings of the 2nd workshop on Learning language in logic and the 4th conference on Computational natural language learning-Volume 7*, page 144. Association for Computational Linguistics, 2000. (Cited on page [49](#).)

## L

---

- [41] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Machine Learning-International Workshop Then Conference*, pages 282–289. Citeseer, 2001. (Cited on pages [xiii](#), [21](#) y [22](#).)
- [42] J.H. Lim, Y.S. Hwang, S.Y. Park, and H.C. Rim. Semantic role labeling using maximum entropy model. In *Proceedings of CoNLL*, pages 122–125, 2004. (Cited on pages [xvii](#), [34](#), [35](#) y [36](#).)
- [43] T. Liu, W. Che, S. Li, Y. Hu, and H. Liu. Semantic role labeling system using maximum entropy classifier. In *Ninth Conference on Computational Natural Language Learning*, page 189, 2005. (Cited on pages [xvii](#), [34](#), [36](#) y [37](#).)
- [44] F. Llopis, R. Muñoz, A. Suárez, and A. Montoyo. EXIT: Propuesta de un sistema de extracción de información de textos notariales. *Novatica*, 133:26–30, 1998. (Cited on page [9](#).)

## M

---

- [45] L. Màrquez and J. Gimenez. Fast and accurate part-of-speech tagging: The svm approach revisited. *Recent advances in natural language processing III: selected papers from RANLP 2003*, page 153, 2004. (Cited on page [32](#).)



- [46] L. Marquez, P. Comas, J. Giménez, and N. Catala. Semantic role labeling as sequential tagging. In *Proceedings of the Ninth Conference on Computational Natural Language Learning*, pages 193–196. Association for Computational Linguistics, 2005. (Cited on pages 43 y 48.)
- [47] M.A. Martí, M. Taulé, M. Bertran, and L. Màrquez. Ancora: Multilingual and multilevel annotated corpora. *MS, Universitat de Barcelona*, 2007. (Cited on page 13.)
- [48] A. Meyers. Annotation guidelines for nombank-noun argument structure for propbank. Technical report, Technical report, New York University, 2007. (Cited on page 13.)
- [49] M. Minoux. *Mathematical programming: theory and algorithms*. John Wiley & Sons, 1986. (Cited on page 24.)
- [50] T. Mitsumori, M. Murata, Y. Fukuda, K. Doi, and H. Doi. Semantic role labeling using support vector machines. In *Proceedings of CoNLL-2005*, 2005. (Cited on pages xiii, xviii, 39, 41, 42 y 43.)
- [51] P. Moreda. *Los Roles Semánticos en la Tecnología del Lenguaje Humano: Anotación y Aplicación*. PhD thesis, Universidad de Alicante, 2008. (Cited on pages 2, 8, 9, 11 y 13.)
- [52] P. Moreda, B. Navarro, and M. Palomar. Using semantic roles in information retrieval systems. *Natural Language Processing and Information Systems*, pages 192–202, 2005. (Cited on page 1.)

## N

---

- [53] S. Narayanan and S. Harabagiu. Question answering based on semantic structures. In *Proceedings of the 20th international conference on Computational Linguistics*, page 693. Association for Computational Linguistics, 2004. (Cited on page 13.)

## O

---

- [54] T. O'Hara and J. Wiebe. Exploiting semantic role resources for preposition disambiguation. *Computational Linguistics*, 35(2):151–184, 2009. (Cited on page 1.)
- [55] N.E. Ozgencil and N. McCracken. Semantic role labeling using libSVM. In *Proceedings of the Ninth Conference on Computational Natural Language Learning*, pages 205–208. Association for Computational Linguistics, 2005. (Cited on pages xviii, 39, 40 y 41.)

## P

---

- [56] M. Palmer, D. Gildea, and P. Kingsbury. The proposition bank: A corpus annotated with semantic roles. *Computational Linguistics*, 31(1):71–106, 2005. (Cited on page 32.)
- [57] M. Panero. Métodos de votación híbridos bajo preferencias ordinarias y difusas. In *Anales de Estudios Económicos y Empresariales*, number 16, pages 187–219. 2006. (Cited on page 55.)

- [58] F. Pla. *Etiquetado léxico y análisis sintáctico superficial basado en modelos estadísticos*. PhD thesis, Universidad Politécnica de Valencia, 2000. (Cited on pages [xiii](#), [7](#) y [8](#).)
- [59] S. Pradhan, K. Hacioglu, W. Ward, J.H. Martin, and D. Jurafsky. Semantic role chunking combining complementary syntactic views. In *Ninth Conference on Computational Natural Language Learning*, page 217, 2005. (Cited on pages [43](#) y [49](#).)
- [60] V. Punyakanok, D. Roth, W. Yih, and D. Zimak. Semantic role labeling via integer linear programming inference. In *Proceedings of the 20th international conference on Computational Linguistics*. Association for Computational Linguistics Morristown, NJ, USA, 2004. (Cited on page [45](#).)

## R

---

- [61] A. Ratnaparkhi. A simple introduction to maximum entropy models for natural language processing. *IRCS Report*, pages 97–08, 1997. (Cited on page [19](#).)
- [62] J. Romero. Máxima entropía. *Universidad Autónoma de México*, 2008. (Cited on pages [17](#), [18](#) y [19](#).)
- [63] J. Ruppenhofer, M. Ellsworth, M.R.L. Petruck, C. Johnson, and J. Scheffczyk. *FrameNet II: Extended Theory and Practice*. University of California, Berkeley, 2010. (Cited on page [12](#).)

## S

---

- [64] Omar Sabaj M. El uso de los participantes semánticos en los predicados de cambio de estado del español: una aproximación basada en corpus. *Literatura y lingüística*, pages 267 – 301, 2006. (Cited on page [11](#).)
- [65] R.C. Schank. Conceptual dependency: A theory of natural language understanding. *Cognitive psychology*, 3(4):552–631, 1972. (Cited on page [12](#).)
- [66] B. Schölkopf. *Support vector learning*. Oldenbourg Verlag, 1997. (Cited on page [26](#).)
- [67] D. Shen and M. Lapata. Using semantic roles to improve question answering. In *Proceedings of EMNLP-CoNLL*, pages 12–21, 2007. (Cited on page [1](#).)
- [68] N. Smith. *Chomsky: Ideas e Ideales*. Cambridge University Press, 2001. (Cited on page [2](#).)
- [69] R.V. Solé and S.C. Manrubia. *Orden y caos en sistemas complejos*. Ediciones UPC, 2001. (Cited on page [18](#).)
- [70] S. Stenchikova, D. Hakkani-Tür, and G. Tur. QASR: Question answering using semantic roles for speech interface. In *Ninth International Conference on Spoken Language Processing*. Citeseer, 2006. (Cited on page [1](#).)

- [71] M. Surdeanu and J. Turmo. Semantic role labeling using complete syntactic analysis. *Training*, 96(2.49):1–45. (Cited on pages [xiii](#), [43](#) y [50](#).)
- [72] C. Sutton and A. McCallum. An introduction to conditional random fields for relational learning. *Introduction to statistical relational learning*, page 93, 2007. (Cited on page [22](#).)

## T

---

- [73] Erik F. Tjong Kim Sang and Jorn Veenstra. Representing text chunks. In *Proceedings of EACL'99*, pages 173–179. Bergen, Norway, 1999. (Cited on pages [xviii](#) y [41](#).)
- [74] K. Toutanova, A. Haghighi, and C.D. Manning. Joint learning improves semantic role labeling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 589–596. Association for Computational Linguistics Morristown, NJ, USA, 2005. (Cited on pages [xiii](#), [43](#), [46](#), [47](#) y [48](#).)

## V

---

- [75] M. Vallez and R. Pedraza-Jiménez. El Procesamiento del Lenguaje Natural en la Recuperación de Información Textual y áreas afines. *Hipertext. net*, (5):7, 2007. (Cited on pages [xiii](#) y [9](#).)
- [76] S. Vazquez, A. Montoyo, and G. Rigau. Método de desambiguación léxica basada en el recurso léxico dominios relevantes. *Procesamiento del lenguaje natural*, (31):141–148, 2003. (Cited on page [7](#).)
- [77] P. Viola and M. Jones. Fast and robust classification using asymmetric adaboost and a detector cascade. *Advances in Neural Information Processing Systems*, 2:1311–1318, 2002. (Cited on page [49](#).)

## W

---

- [78] H.M. Wallach. Conditional random fields: An introduction. *Rapport technique MS-CIS-04-21, Department of Computer and Information Science, University of Pennsylvania*, 50, 2004. (Cited on page [21](#).)
- [79] D. Wu and P. Fung. Semantic roles for SMT: A hybrid two-pass model. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, pages 13–16. Association for Computational Linguistics, 2009. (Cited on page [1](#).)

## X

---

- [80] N. Xue and M. Palmer. Calibrating features for semantic role labeling. In *Proceedings of EMNLP*, volume 4, 2004. (Cited on pages [38](#), [44](#), [45](#) y [49](#).)

**Z**

---

- [81] T. Zhang. Statistical behavior and consistency of classification methods based on convex risk minimization. *Annals of Statistics*, 32(1):56–85, 2004. (Cited on page [27](#).)

ANEXO1. RESULTADOS COMPLETOS DE LA EXPERIMENTACIÓN.

En esta sección se listan las gráficas completas de todo el proceso de experimentación llevado a cabo. Los valores que se detallan toman en consideración todas las combinaciones posibles, así como los valores de precisión, *recall* y *F-Measure*.

Corpus *tWSJ* y *tBrown*

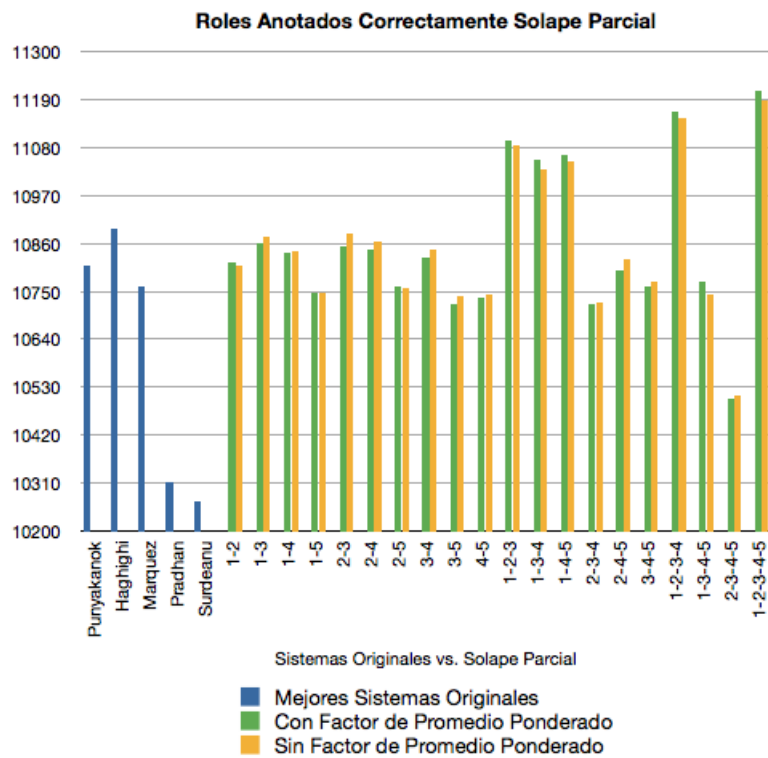


Figura 32: Roles Anotados Correctamente usando Solape Parcial, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus *tWSJ*.

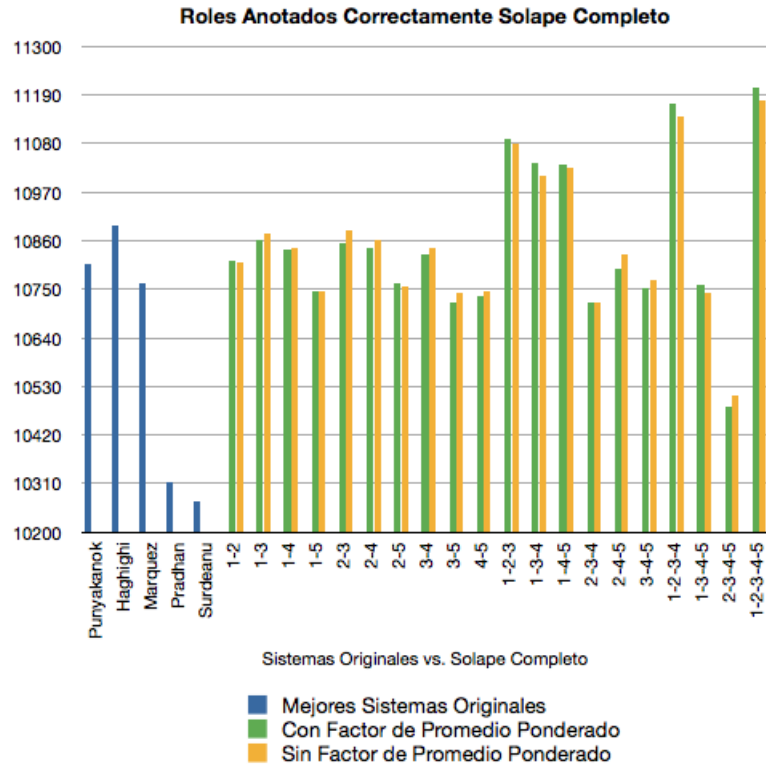


Figura 33: Roles Anotados Correctamente usando Solape Completo, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tWSJ.

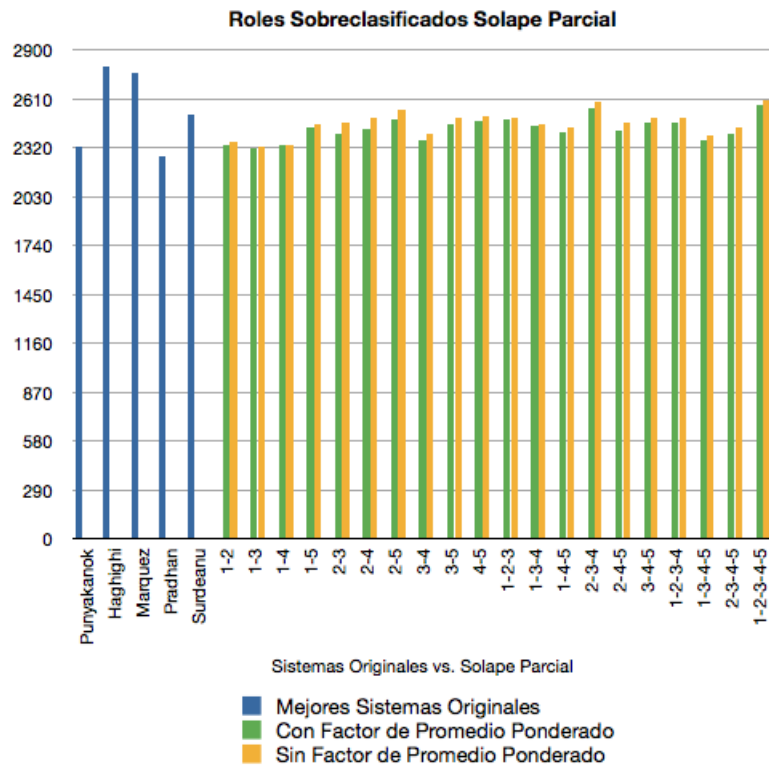


Figura 34: Roles Sobreclasificados usando Solape Parcial, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tWSJ.

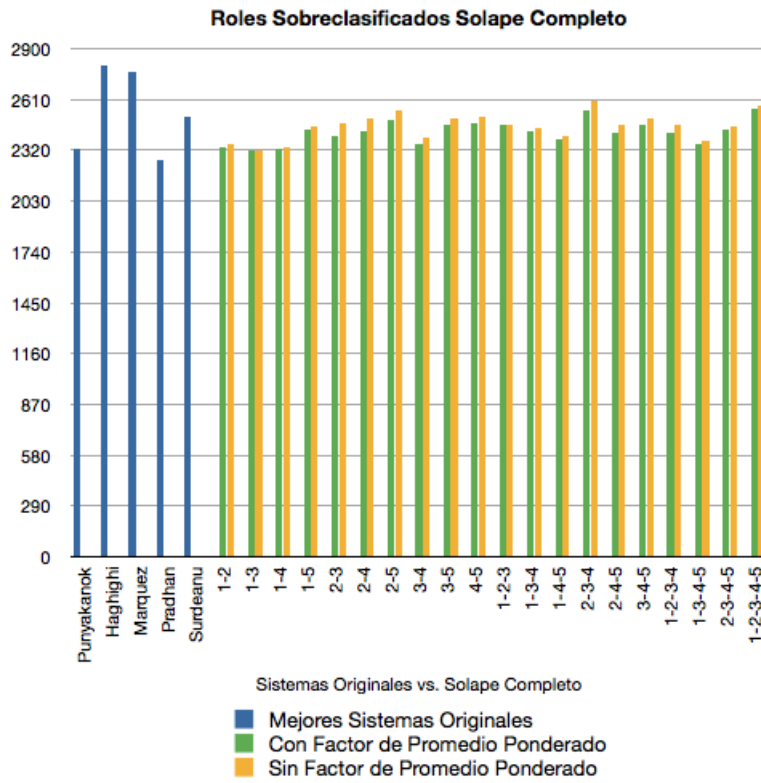


Figura 35: Roles Sobrecalificados usando Solape Completo, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tWSJ.

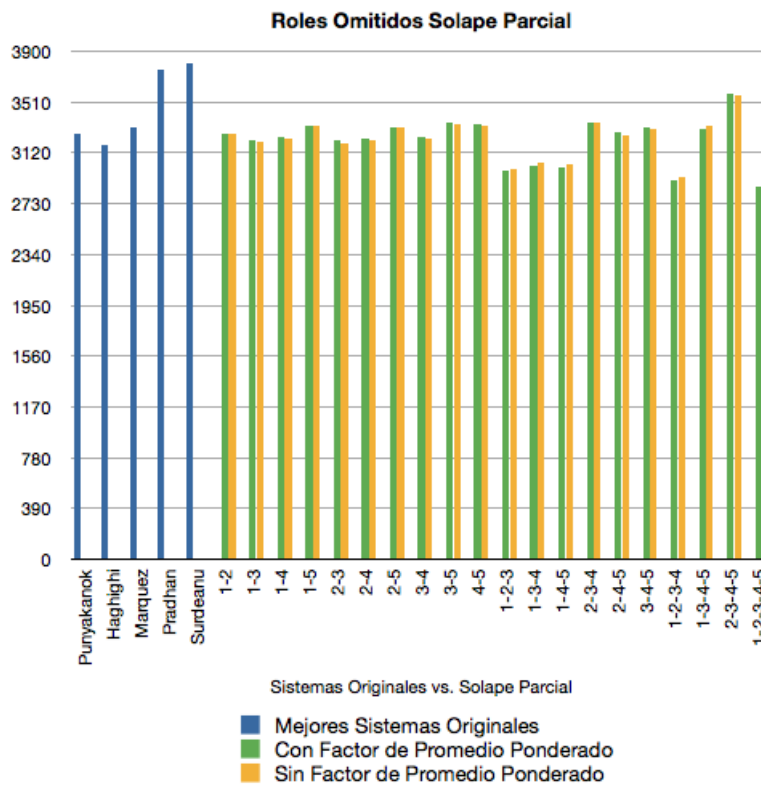


Figura 36: Roles Omitidos usando Solape Parcial, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tWSJ.

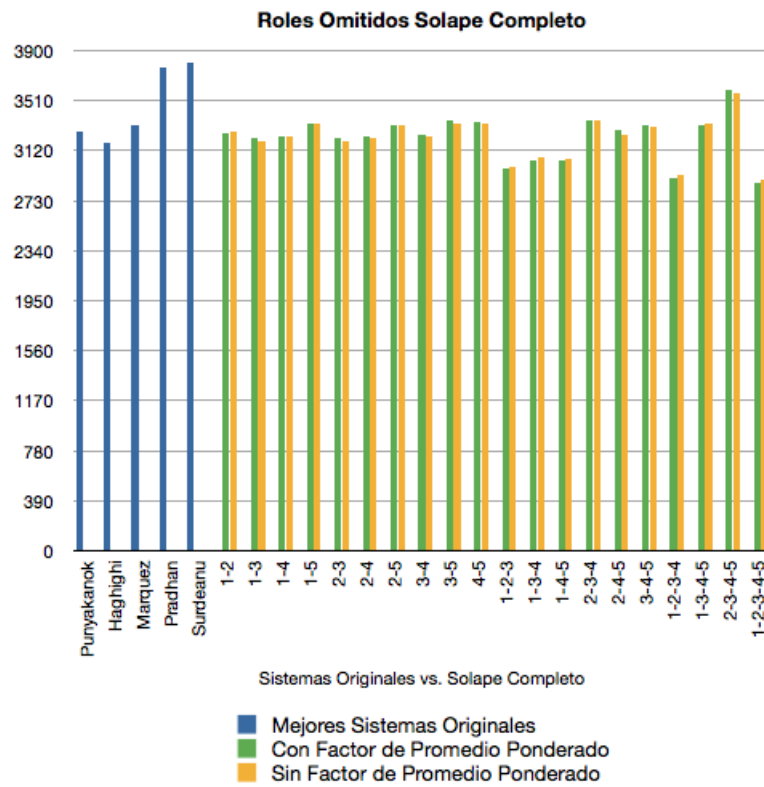


Figura 37: Roles Omitidos usando Solape Completo, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tWSJ.

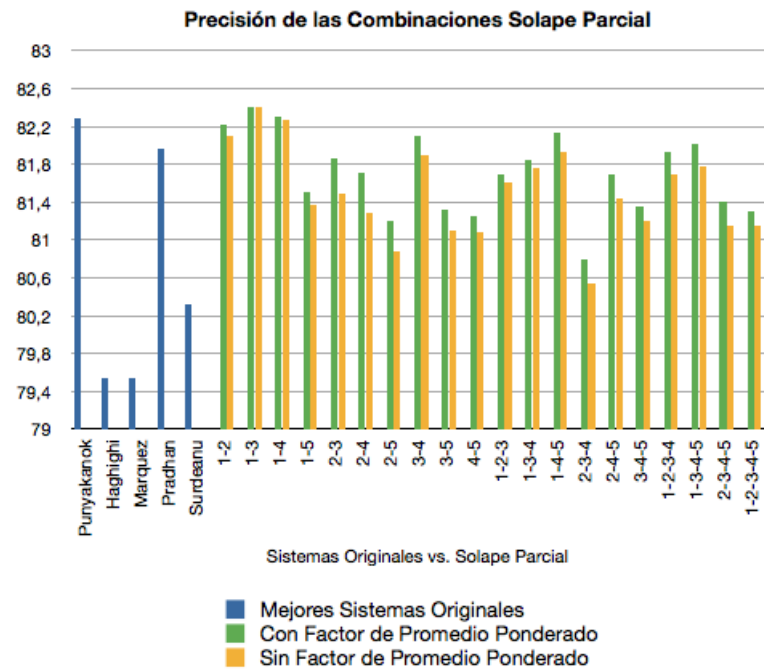


Figura 38: Precisión de las Combinaciones usando Solape Parcial, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tWSJ.



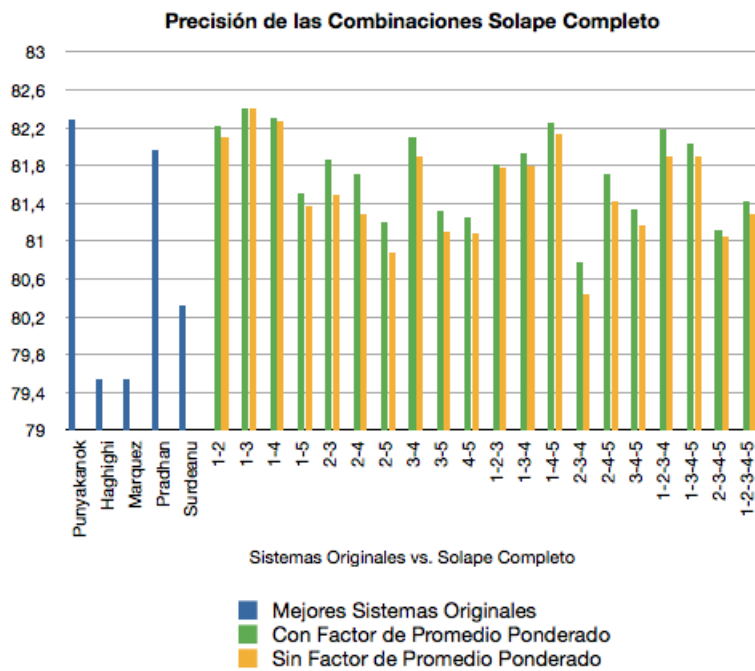


Figura 39: Precisión de las Combinaciones usando Solape Completo, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tWSJ.

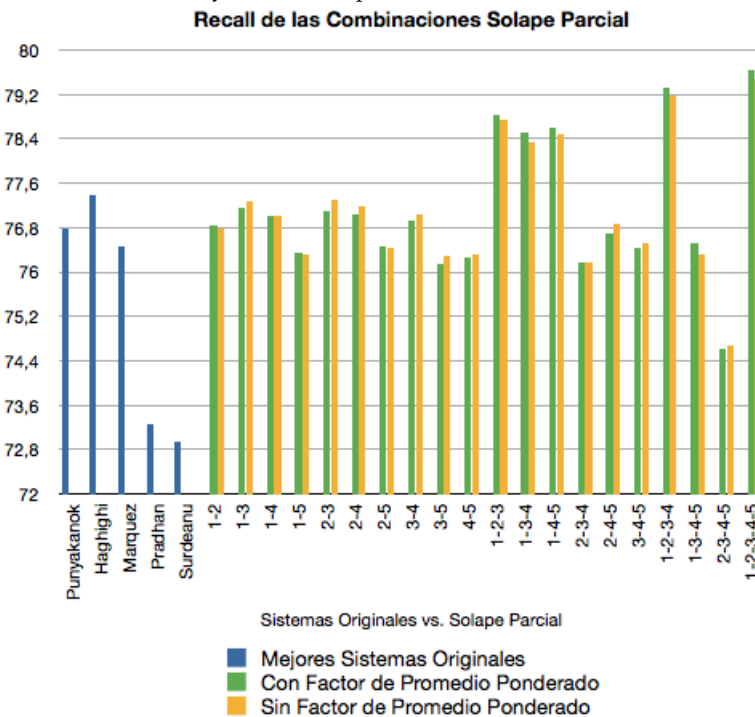


Figura 40: Recall de las Combinaciones usando Solape Parcial, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tWSJ.

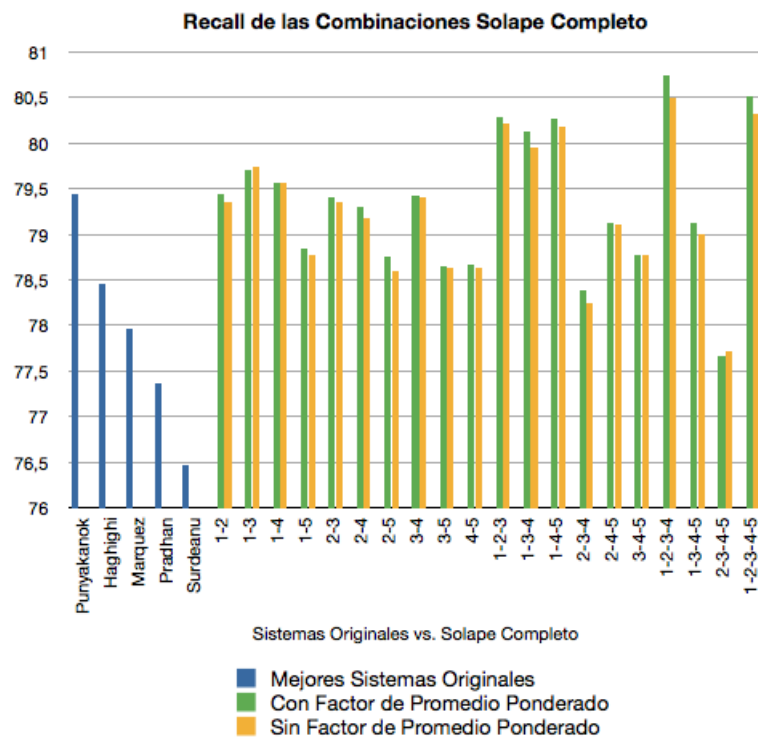


Figura 41: Recall de las Combinaciones usando Solape Completo, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tWSJ.

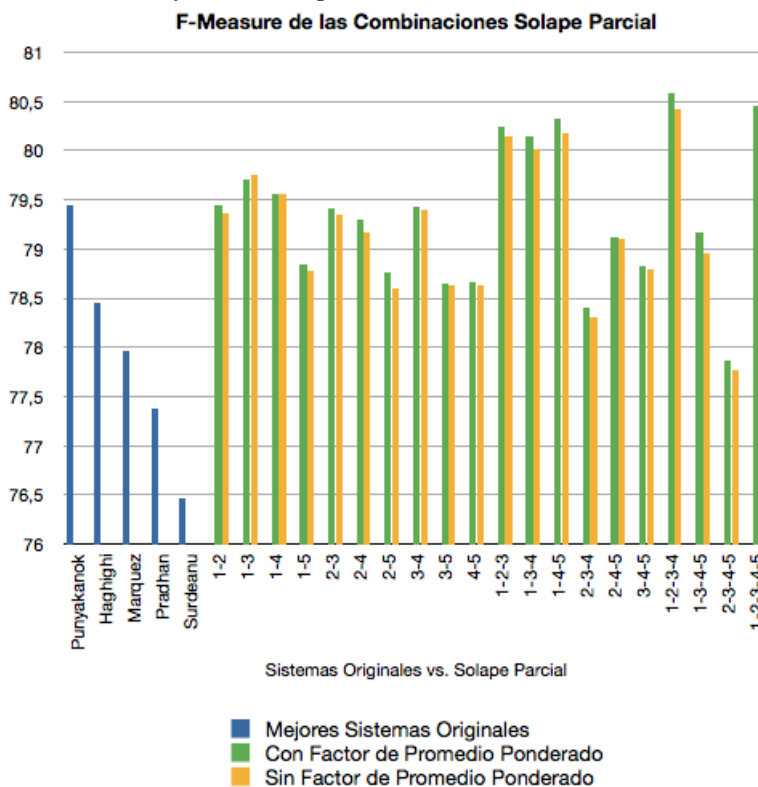


Figura 42: F-Measure de las Combinaciones usando Solape Parcial, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tWSJ.

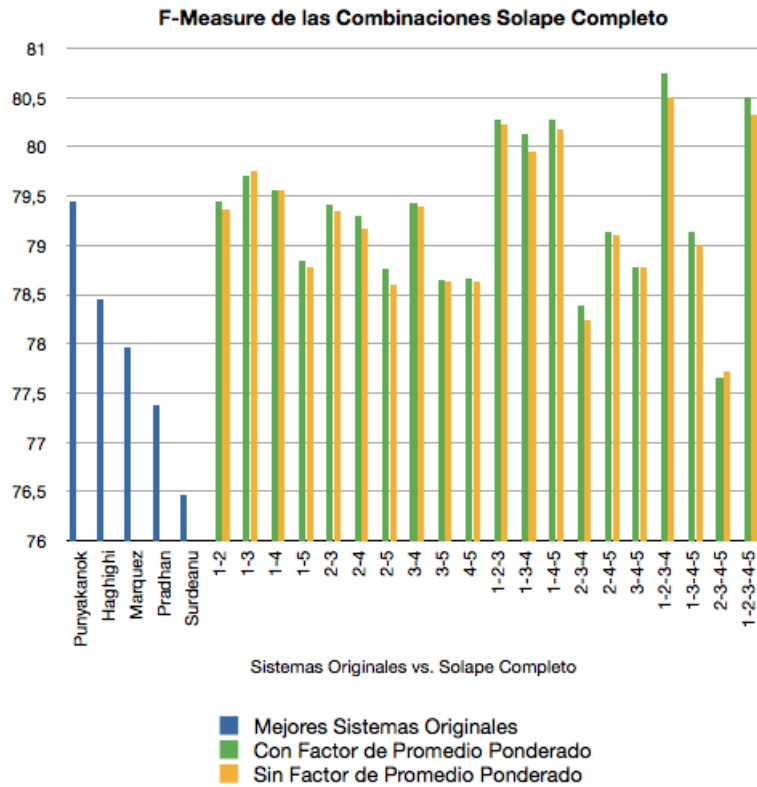


Figura 43: F-Measure de las Combinaciones usando Solape Completo, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tWSJ.

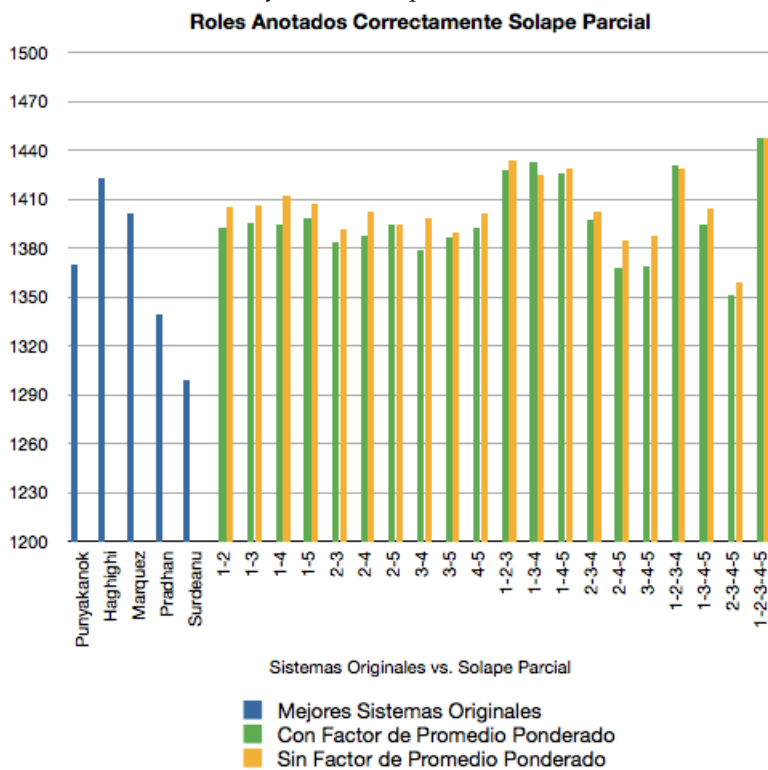


Figura 44: Roles Anotados Correctamente usando Solape Parcial, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tBrown.

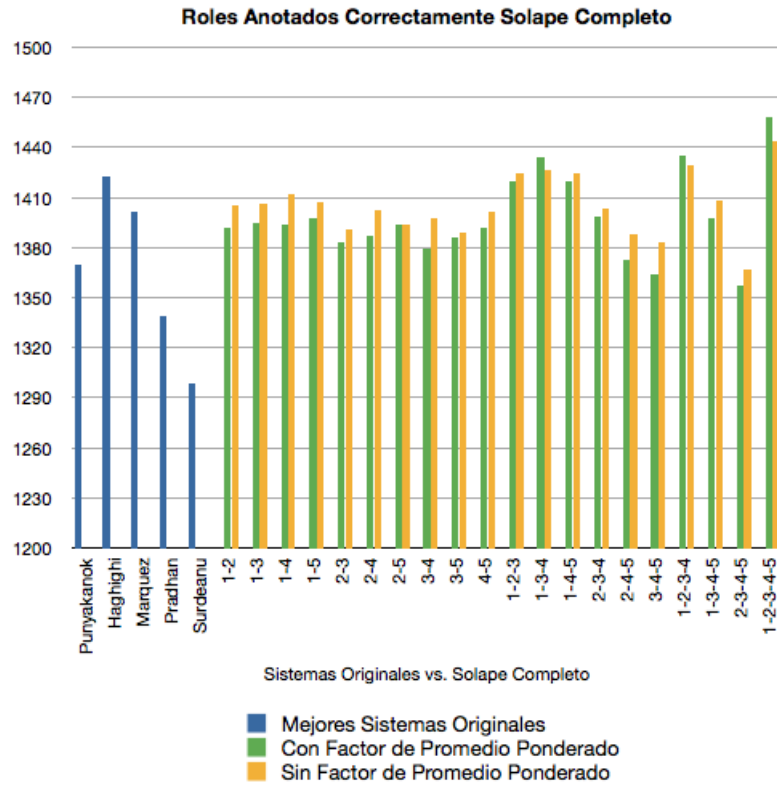


Figura 45: Roles Anotados Correctamente usando Solape Completo, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tBrown.

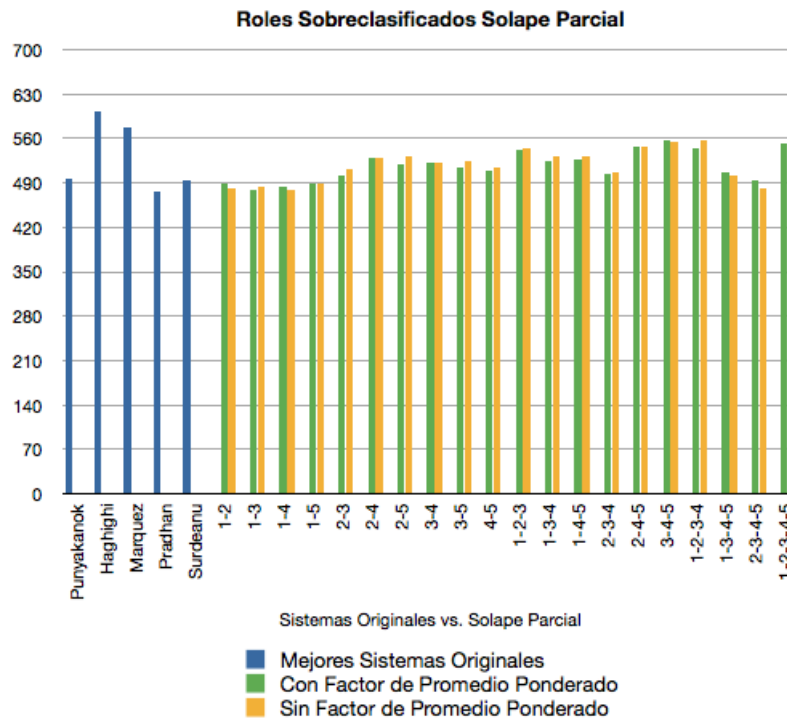


Figura 46: Roles Sobreclasificados usando Solape Parcial, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tBrown.

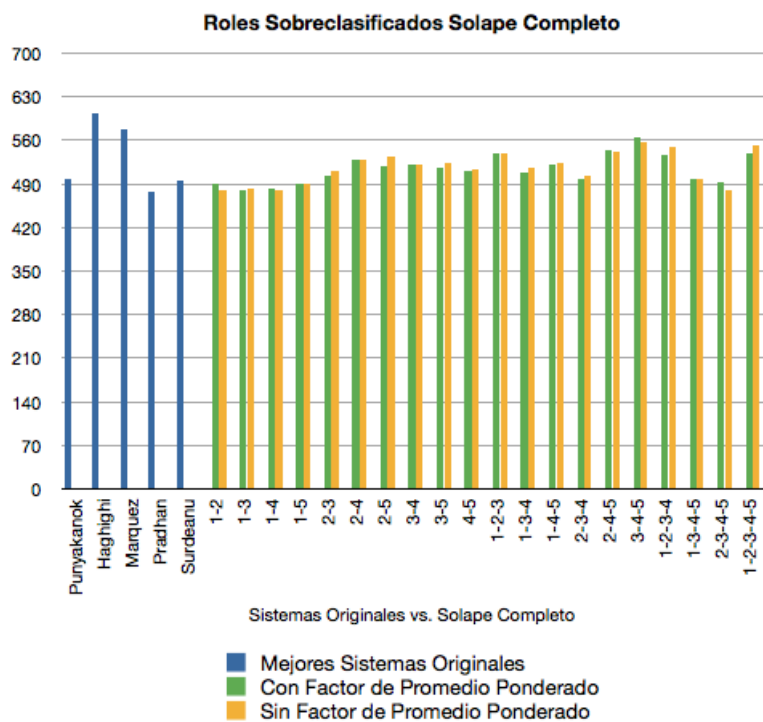


Figura 47: Roles Sobrecalificados usando Solape Completo, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tBrown.

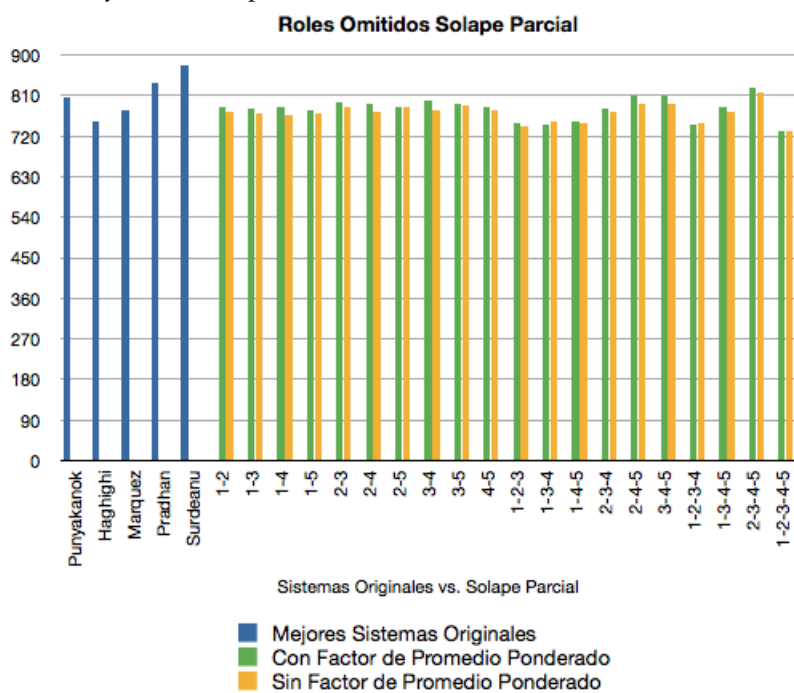


Figura 48: Roles Omitidos usando Solape Parcial, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tBrown.

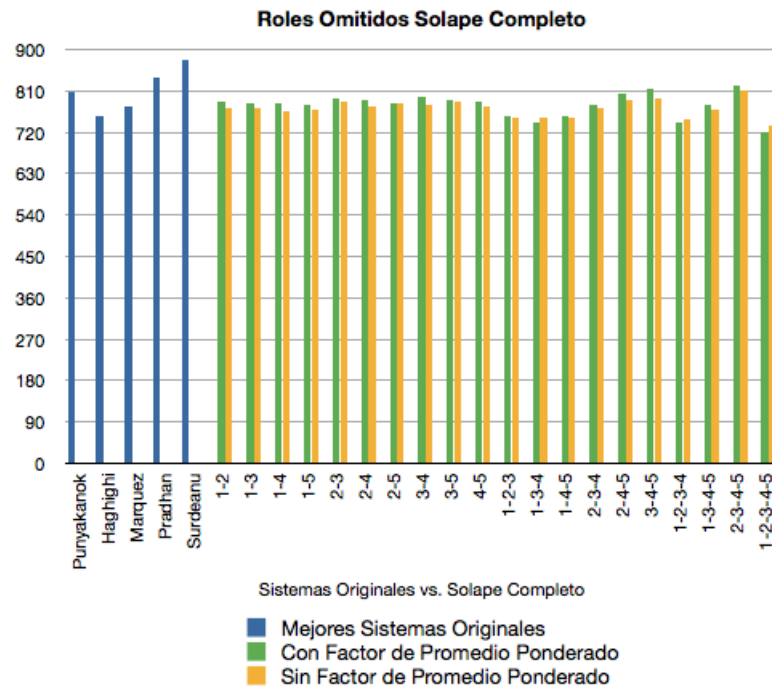


Figura 49: Roles Omitidos usando Solape Completo, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tBrown.

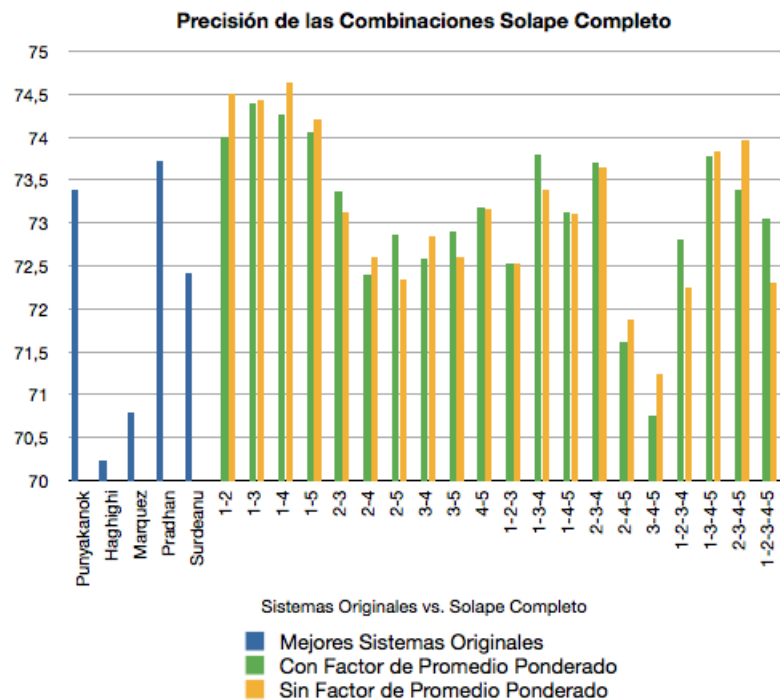


Figura 50: Precisión de las Combinaciones usando Solape Completo, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tBrown.

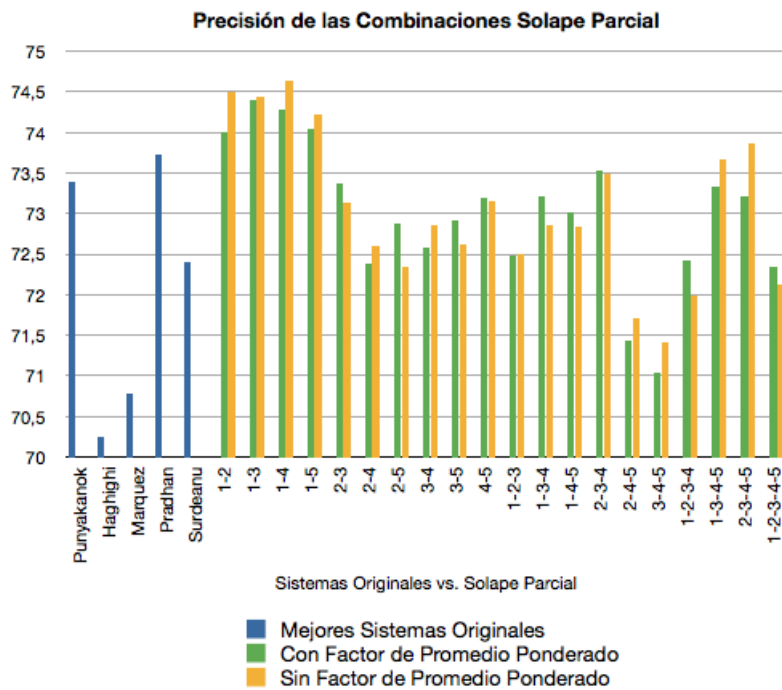


Figura 51: Precisión de las Combinaciones usando Solape Parcial, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tBrown.

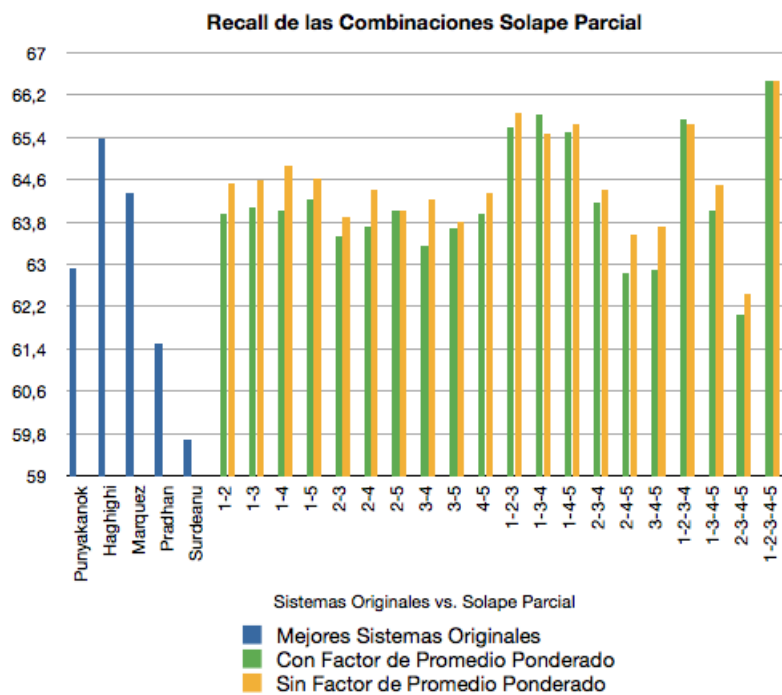


Figura 52: Recall de las Combinaciones usando Solape Parcial, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tBrown.

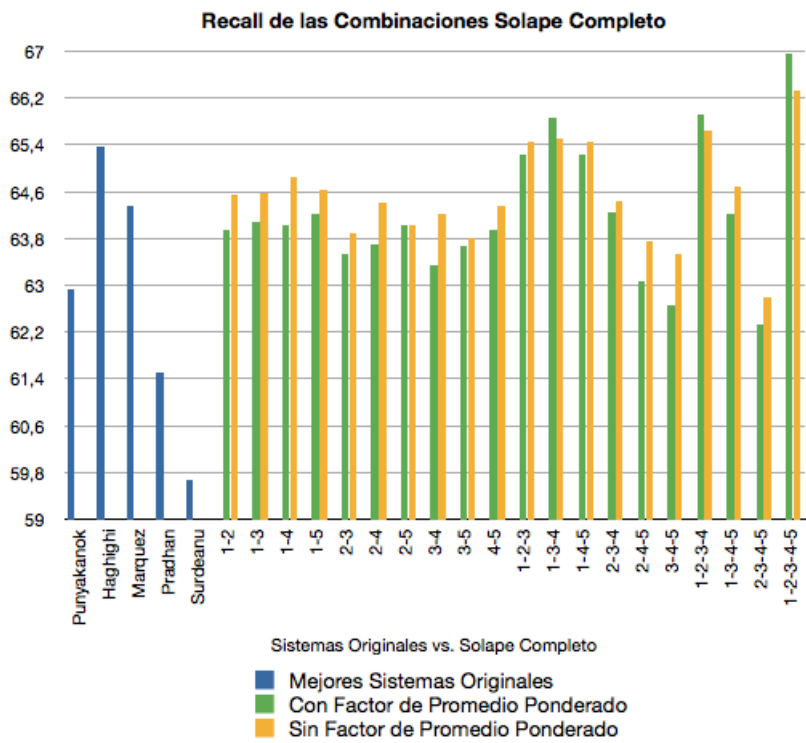


Figura 53: Recall de las Combinaciones usando Solape Completo, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tBrown.

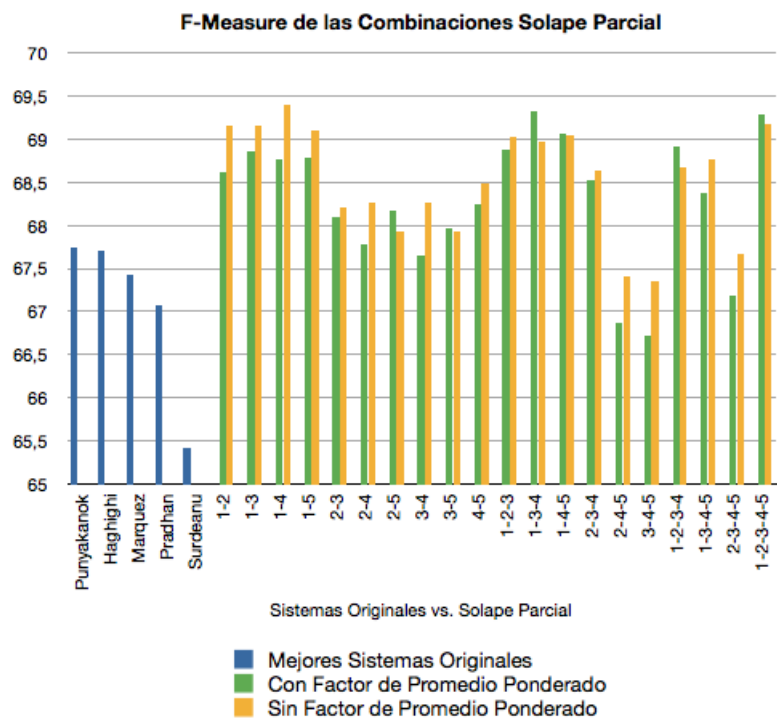


Figura 54: F-Measure de las Combinaciones usando Solape Parcial, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tBrown.



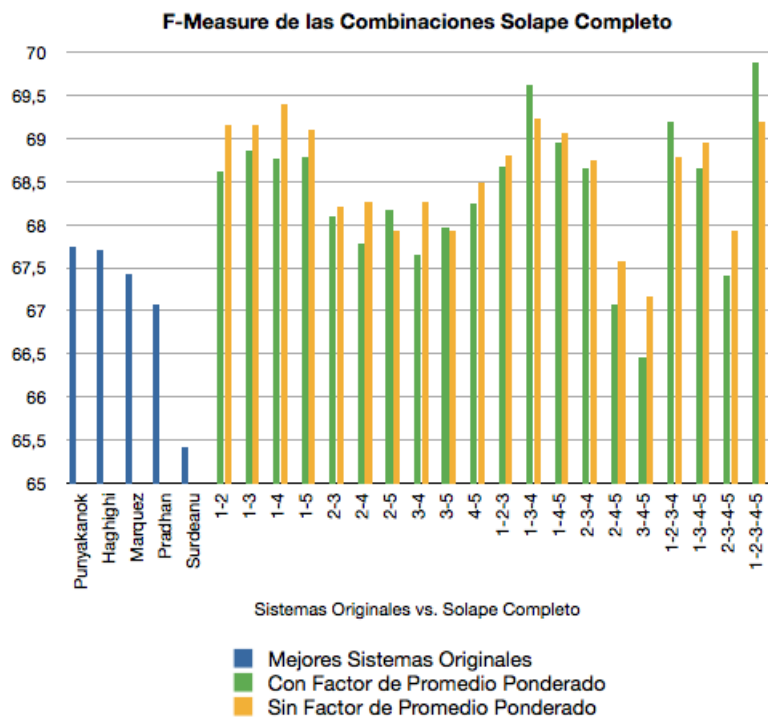


Figura 55: F-Measure de las Combinaciones usando Solape Completo, con factores  $\alpha = 1,0$  y  $\beta = 1,0$ , corpus tBrown.

## ANEXO 2. NIVELES DE PRECISIÓN, RECALL Y F-MEASURE DE CADA SISTEMA.

En esta sección recogemos los niveles de precisión, *recall* y *F-Measure* con que etiquetan cada tipo de rol los 5 mejores de la competición CoNLL 2005<sup>2</sup>. Los valores que se detallan en cada figura son los siguientes:

*corr.* - Indica el número de roles que el sistema anota correctamente.

*excess.* - Roles que se anotan de forma incorrecta.

*missed.* - Falsos negativos que genera el sistema.

*prec.* - Precisión

*rec.* - Recall

*F1.* - F-Measure

Corpus *tWSJ* y *tBrown*

Number of Sentences : 2416  
 Number of Propositions : 5267  
 Percentage of perfect props : 53.79

|          | corr. | excess | missed | prec. | rec.  | F1    |
|----------|-------|--------|--------|-------|-------|-------|
| Overall  | 10000 | 2327   | 3269   | 82.28 | 76.78 | 79.44 |
| A0       | 3131  | 418    | 432    | 88.22 | 87.88 | 88.05 |
| A1       | 3828  | 826    | 1099   | 82.25 | 77.69 | 79.91 |
| A2       | 670   | 186    | 440    | 78.27 | 60.36 | 68.16 |
| A3       | 91    | 19     | 82     | 82.73 | 52.60 | 64.31 |
| A4       | 73    | 14     | 29     | 83.91 | 71.57 | 77.25 |
| A5       | 0     | 0      | 5      | 0.00  | 0.00  | 0.00  |
| AM-ADV   | 284   | 161    | 222    | 63.82 | 56.13 | 59.73 |
| AM-CAU   | 34    | 19     | 39     | 64.15 | 46.58 | 53.97 |
| AM-DIR   | 33    | 24     | 52     | 57.89 | 38.82 | 46.48 |
| AM-DIS   | 258   | 84     | 62     | 75.44 | 80.62 | 77.95 |
| AM-EXT   | 15    | 7      | 17     | 68.18 | 46.88 | 55.56 |
| AM-LOC   | 200   | 100    | 163    | 66.67 | 55.10 | 60.33 |
| AM-MNR   | 183   | 91     | 161    | 66.79 | 53.20 | 59.22 |
| AM-MOD   | 544   | 22     | 7      | 96.11 | 98.73 | 97.40 |
| AM-NEG   | 225   | 6      | 5      | 97.40 | 97.83 | 97.61 |
| AM-PNC   | 42    | 28     | 73     | 60.00 | 36.52 | 45.41 |
| AM-PRD   | 0     | 0      | 5      | 0.00  | 0.00  | 0.00  |
| AM-REC   | 0     | 0      | 2      | 0.00  | 0.00  | 0.00  |
| AM-TMP   | 834   | 233    | 253    | 78.16 | 76.72 | 77.44 |
| R-A0     | 192   | 22     | 32     | 89.72 | 85.71 | 87.67 |
| R-A1     | 119   | 51     | 37     | 70.00 | 76.28 | 73.01 |
| R-A2     | 6     | 1      | 10     | 85.71 | 37.50 | 52.17 |
| R-A3     | 0     | 0      | 1      | 0.00  | 0.00  | 0.00  |
| R-A4     | 0     | 0      | 1      | 0.00  | 0.00  | 0.00  |
| R-AM-ADV | 0     | 0      | 2      | 0.00  | 0.00  | 0.00  |
| R-AM-CAU | 0     | 0      | 4      | 0.00  | 0.00  | 0.00  |
| R-AM-EXT | 0     | 0      | 1      | 0.00  | 0.00  | 0.00  |
| R-AM-LOC | 12    | 2      | 9      | 85.71 | 57.14 | 68.57 |
| R-AM-MNR | 0     | 0      | 6      | 0.00  | 0.00  | 0.00  |
| R-AM-TMP | 34    | 13     | 18     | 72.34 | 65.38 | 68.69 |
| V        | 5114  | 56     | 153    | 98.92 | 97.10 | 98.00 |

Figura 56: Niveles de precisión, recall y F-Measure del sistema Punyakanok, corpus *tWSJ*.

<sup>2</sup> Estos valores se han obtenido utilizando los *scripts* de evaluación que se provee en la página oficial de la competición

Number of Sentences : 2416  
 Number of Propositions : 5267  
 Percentage of perfect props : 56.52

|          | corr. | excess | missed | prec.  | rec.  | F1    |
|----------|-------|--------|--------|--------|-------|-------|
| Overall  | 10894 | 2803   | 3183   | 79.54  | 77.39 | 78.45 |
| A0       | 3146  | 416    | 417    | 88.32  | 88.30 | 88.31 |
| A1       | 3863  | 1051   | 1064   | 78.61  | 78.40 | 78.51 |
| A2       | 756   | 286    | 354    | 72.55  | 68.11 | 70.26 |
| A3       | 95    | 35     | 78     | 73.08  | 54.91 | 62.71 |
| A4       | 72    | 21     | 30     | 77.42  | 70.59 | 73.85 |
| A5       | 4     | 0      | 1      | 100.00 | 80.00 | 88.89 |
| AM-ADV   | 259   | 186    | 247    | 58.20  | 51.19 | 54.47 |
| AM-CAU   | 39    | 22     | 34     | 63.93  | 53.42 | 58.21 |
| AM-DIR   | 41    | 37     | 44     | 52.56  | 48.24 | 50.31 |
| AM-DIS   | 258   | 79     | 62     | 76.56  | 80.62 | 78.54 |
| AM-EXT   | 14    | 5      | 18     | 73.68  | 43.75 | 54.90 |
| AM-LOC   | 203   | 127    | 160    | 61.52  | 55.92 | 58.59 |
| AM-MNR   | 196   | 140    | 148    | 58.33  | 56.98 | 57.65 |
| AM-MOD   | 546   | 12     | 5      | 97.85  | 99.09 | 98.47 |
| AM-NEG   | 226   | 6      | 4      | 97.41  | 98.26 | 97.84 |
| AM-PNC   | 50    | 51     | 65     | 49.50  | 43.48 | 46.30 |
| AM-PRD   | 1     | 0      | 4      | 100.00 | 20.00 | 33.33 |
| AM-REC   | 0     | 0      | 2      | 0.00   | 0.00  | 0.00  |
| AM-TMP   | 732   | 246    | 355    | 74.85  | 67.34 | 70.90 |
| R-A0     | 201   | 16     | 23     | 92.63  | 89.73 | 91.16 |
| R-A1     | 128   | 29     | 28     | 81.53  | 82.05 | 81.79 |
| R-A2     | 8     | 5      | 8      | 61.54  | 50.00 | 55.17 |
| R-A3     | 0     | 1      | 1      | 0.00   | 0.00  | 0.00  |
| R-A4     | 0     | 0      | 1      | 0.00   | 0.00  | 0.00  |
| R-AM-ADV | 0     | 0      | 2      | 0.00   | 0.00  | 0.00  |
| R-AM-CAU | 2     | 0      | 2      | 100.00 | 50.00 | 66.67 |
| R-AM-EXT | 0     | 0      | 1      | 0.00   | 0.00  | 0.00  |
| R-AM-LOC | 12    | 2      | 9      | 85.71  | 57.14 | 68.57 |
| R-AM-MNR | 2     | 5      | 4      | 28.57  | 33.33 | 30.77 |
| R-AM-TMP | 40    | 25     | 12     | 61.54  | 76.92 | 68.38 |
| V        | 5126  | 141    | 141    | 97.32  | 97.32 | 97.32 |

Figura 57: Niveles de precisión, recall y F-Measure del sistema Haghighi, corpus tWSJ.

Number of Sentences : 2416  
 Number of Propositions : 5267  
 Percentage of perfect props : 51.85

|          | corr. | excess | missed | prec. | rec.  | F1    |
|----------|-------|--------|--------|-------|-------|-------|
| Overall  | 10762 | 2766   | 3315   | 79.55 | 76.45 | 77.97 |
| A0       | 3074  | 455    | 489    | 87.11 | 86.28 | 86.69 |
| A1       | 3780  | 969    | 1147   | 79.60 | 76.72 | 78.13 |
| A2       | 752   | 335    | 358    | 69.18 | 67.75 | 68.46 |
| A3       | 97    | 30     | 76     | 76.38 | 56.07 | 64.67 |
| A4       | 71    | 18     | 31     | 79.78 | 69.61 | 74.35 |
| A5       | 0     | 0      | 5      | 0.00  | 0.00  | 0.00  |
| AM-ADV   | 265   | 183    | 241    | 59.15 | 52.37 | 55.56 |
| AM-CAU   | 42    | 15     | 31     | 73.68 | 57.53 | 64.62 |
| AM-DIR   | 30    | 12     | 55     | 71.43 | 35.29 | 47.24 |
| AM-DIS   | 243   | 72     | 77     | 77.14 | 75.94 | 76.54 |
| AM-EXT   | 14    | 8      | 18     | 63.64 | 43.75 | 51.85 |
| AM-LOC   | 197   | 117    | 166    | 62.74 | 54.27 | 58.20 |
| AM-MNR   | 182   | 153    | 162    | 54.33 | 52.91 | 53.61 |
| AM-MOD   | 526   | 21     | 25     | 96.16 | 95.46 | 95.81 |
| AM-NEG   | 227   | 2      | 3      | 99.13 | 98.70 | 98.91 |
| AM-PNC   | 46    | 40     | 69     | 53.49 | 40.00 | 45.77 |
| AM-PRD   | 0     | 0      | 5      | 0.00  | 0.00  | 0.00  |
| AM-REC   | 0     | 0      | 2      | 0.00  | 0.00  | 0.00  |
| AM-TMP   | 856   | 246    | 231    | 77.68 | 78.75 | 78.21 |
| R-A0     | 198   | 30     | 26     | 86.84 | 88.39 | 87.61 |
| R-A1     | 119   | 39     | 37     | 75.32 | 76.28 | 75.80 |
| R-A2     | 6     | 5      | 10     | 54.55 | 37.50 | 44.44 |
| R-A3     | 0     | 0      | 1      | 0.00  | 0.00  | 0.00  |
| R-A4     | 0     | 0      | 1      | 0.00  | 0.00  | 0.00  |
| R-AM-ADV | 0     | 0      | 2      | 0.00  | 0.00  | 0.00  |
| R-AM-CAU | 0     | 0      | 4      | 0.00  | 0.00  | 0.00  |
| R-AM-EXT | 0     | 0      | 1      | 0.00  | 0.00  | 0.00  |
| R-AM-LOC | 0     | 0      | 21     | 0.00  | 0.00  | 0.00  |
| R-AM-MNR | 0     | 0      | 6      | 0.00  | 0.00  | 0.00  |
| R-AM-TMP | 37    | 16     | 15     | 69.81 | 71.15 | 70.48 |
| V        | 5223  | 44     | 44     | 99.16 | 99.16 | 99.16 |

Figura 58: Niveles de precisión, recall y F-Measure del sistema Marquez, corpus tWSJ.

Number of Sentences : 2416  
 Number of Propositions : 5267  
 Percentage of perfect props : 50.14

|          | corr. | excess | missed | prec.  | rec.  | F1    |
|----------|-------|--------|--------|--------|-------|-------|
| Overall  | 10314 | 2269   | 3763   | 81.97  | 73.27 | 77.37 |
| A0       | 2930  | 276    | 633    | 91.39  | 82.23 | 86.57 |
| A1       | 3756  | 951    | 1171   | 79.80  | 76.23 | 77.97 |
| A2       | 695   | 318    | 415    | 68.61  | 62.61 | 65.47 |
| A3       | 88    | 31     | 85     | 73.95  | 50.87 | 60.27 |
| A4       | 70    | 19     | 32     | 78.65  | 68.63 | 73.30 |
| A5       | 3     | 1      | 2      | 75.00  | 60.00 | 66.67 |
| AM-ADV   | 233   | 145    | 273    | 61.64  | 46.05 | 52.71 |
| AM-CAU   | 32    | 10     | 41     | 76.19  | 43.84 | 55.65 |
| AM-DIR   | 32    | 28     | 53     | 53.33  | 37.65 | 44.14 |
| AM-DIS   | 203   | 49     | 117    | 80.56  | 63.44 | 70.98 |
| AM-EXT   | 15    | 0      | 17     | 100.00 | 46.88 | 63.83 |
| AM-LOC   | 187   | 103    | 176    | 64.48  | 51.52 | 57.27 |
| AM-MNR   | 156   | 92     | 188    | 62.90  | 45.35 | 52.70 |
| AM-MOD   | 509   | 7      | 42     | 98.64  | 92.38 | 95.41 |
| AM-NEG   | 220   | 4      | 10     | 98.21  | 95.65 | 96.92 |
| AM-PNC   | 51    | 39     | 64     | 56.67  | 44.35 | 49.76 |
| AM-PRD   | 0     | 0      | 5      | 0.00   | 0.00  | 0.00  |
| AM-REC   | 0     | 0      | 2      | 0.00   | 0.00  | 0.00  |
| AM-TMP   | 782   | 156    | 305    | 83.37  | 71.94 | 77.23 |
| R-A0     | 198   | 12     | 26     | 94.29  | 88.39 | 91.24 |
| R-A1     | 116   | 19     | 40     | 85.93  | 74.36 | 79.73 |
| R-A2     | 6     | 0      | 10     | 100.00 | 37.50 | 54.55 |
| R-A3     | 0     | 0      | 1      | 0.00   | 0.00  | 0.00  |
| R-A4     | 0     | 0      | 1      | 0.00   | 0.00  | 0.00  |
| R-AM-ADV | 0     | 0      | 2      | 0.00   | 0.00  | 0.00  |
| R-AM-CAU | 0     | 0      | 4      | 0.00   | 0.00  | 0.00  |
| R-AM-EXT | 0     | 0      | 1      | 0.00   | 0.00  | 0.00  |
| R-AM-LOC | 9     | 1      | 12     | 90.00  | 42.86 | 58.06 |
| R-AM-MNR | 2     | 1      | 4      | 66.67  | 33.33 | 44.44 |
| R-AM-TMP | 21    | 7      | 31     | 75.00  | 40.38 | 52.50 |
| V        | 5207  | 60     | 60     | 98.86  | 98.86 | 98.86 |

Figura 59: Niveles de precisión, recall y F-Measure del sistema Pradhan, corpus tWSJ.

Number of Sentences : 2416  
 Number of Propositions : 5267  
 Percentage of perfect props : 45.28

|          | corr. | excess | missed | prec. | rec.  | F1    |
|----------|-------|--------|--------|-------|-------|-------|
| Overall  | 10269 | 2516   | 3808   | 80.32 | 72.95 | 76.46 |
| A0       | 3036  | 450    | 527    | 87.09 | 85.21 | 86.14 |
| A1       | 3559  | 901    | 1368   | 79.80 | 72.23 | 75.83 |
| A2       | 648   | 219    | 462    | 74.74 | 58.38 | 65.55 |
| A3       | 93    | 19     | 80     | 83.04 | 53.76 | 65.26 |
| A4       | 72    | 21     | 30     | 77.42 | 70.59 | 73.85 |
| A5       | 0     | 0      | 5      | 0.00  | 0.00  | 0.00  |
| AM-ADV   | 233   | 170    | 273    | 57.82 | 46.05 | 51.27 |
| AM-CAU   | 40    | 41     | 33     | 49.38 | 54.79 | 51.95 |
| AM-DIR   | 34    | 20     | 51     | 62.96 | 40.00 | 48.92 |
| AM-DIS   | 244   | 94     | 76     | 72.19 | 76.25 | 74.16 |
| AM-EXT   | 14    | 9      | 18     | 60.87 | 43.75 | 50.91 |
| AM-LOC   | 190   | 106    | 173    | 64.19 | 52.34 | 57.66 |
| AM-MNR   | 154   | 87     | 190    | 63.90 | 44.77 | 52.65 |
| AM-MOD   | 514   | 10     | 37     | 98.09 | 93.28 | 95.63 |
| AM-NEG   | 225   | 9      | 5      | 96.15 | 97.83 | 96.98 |
| AM-PNC   | 37    | 30     | 78     | 55.22 | 32.17 | 40.66 |
| AM-PRD   | 0     | 0      | 5      | 0.00  | 0.00  | 0.00  |
| AM-REC   | 0     | 0      | 2      | 0.00  | 0.00  | 0.00  |
| AM-TMP   | 798   | 210    | 289    | 79.17 | 73.41 | 76.18 |
| R-A0     | 196   | 35     | 28     | 84.85 | 87.50 | 86.15 |
| R-A1     | 111   | 37     | 45     | 75.00 | 71.15 | 73.03 |
| R-A2     | 6     | 4      | 10     | 60.00 | 37.50 | 46.15 |
| R-A3     | 0     | 0      | 1      | 0.00  | 0.00  | 0.00  |
| R-A4     | 0     | 0      | 1      | 0.00  | 0.00  | 0.00  |
| R-AM-ADV | 0     | 0      | 2      | 0.00  | 0.00  | 0.00  |
| R-AM-CAU | 0     | 0      | 4      | 0.00  | 0.00  | 0.00  |
| R-AM-EXT | 0     | 0      | 1      | 0.00  | 0.00  | 0.00  |
| R-AM-LOC | 17    | 8      | 4      | 68.00 | 80.95 | 73.91 |
| R-AM-MNR | 3     | 7      | 3      | 30.00 | 50.00 | 37.50 |
| R-AM-TMP | 45    | 29     | 7      | 60.81 | 86.54 | 71.43 |
| V        | 0     | 0      | 5267   | 0.00  | 0.00  | 0.00  |

Figura 60: Niveles de precisión, recall y F-Measure del sistema Surdeanu, corpus tWSJ.

Number of Sentences : 426  
 Number of Propositions : 804  
 Percentage of perfect props : 32.34

|          | corr. | excess | missed | prec.  | rec.  | F1    |
|----------|-------|--------|--------|--------|-------|-------|
| Overall  | 1370  | 497    | 807    | 73.38  | 62.93 | 67.75 |
| A0       | 488   | 69     | 78     | 87.61  | 86.22 | 86.91 |
| A1       | 432   | 184    | 244    | 70.13  | 63.91 | 66.87 |
| A2       | 63    | 44     | 84     | 58.88  | 42.86 | 49.61 |
| A3       | 0     | 2      | 12     | 0.00   | 0.00  | 0.00  |
| A4       | 3     | 0      | 12     | 100.00 | 20.00 | 33.33 |
| AM-ADV   | 48    | 23     | 95     | 67.61  | 33.57 | 44.86 |
| AM-CAU   | 2     | 2      | 6      | 50.00  | 25.00 | 33.33 |
| AM-DIR   | 20    | 12     | 33     | 62.50  | 37.74 | 47.06 |
| AM-DIS   | 9     | 20     | 13     | 31.03  | 40.91 | 35.29 |
| AM-EXT   | 0     | 5      | 5      | 0.00   | 0.00  | 0.00  |
| AM-LOC   | 28    | 19     | 57     | 59.57  | 32.94 | 42.42 |
| AM-MNR   | 40    | 31     | 70     | 56.34  | 36.36 | 44.20 |
| AM-MOD   | 88    | 19     | 3      | 82.24  | 96.70 | 88.89 |
| AM-NEG   | 44    | 4      | 6      | 91.67  | 88.00 | 89.80 |
| AM-PNC   | 6     | 3      | 11     | 66.67  | 35.29 | 46.15 |
| AM-PRD   | 0     | 0      | 1      | 0.00   | 0.00  | 0.00  |
| AM-TMP   | 58    | 46     | 54     | 55.77  | 51.79 | 53.70 |
| R-A0     | 23    | 3      | 2      | 88.46  | 92.00 | 90.20 |
| R-A1     | 11    | 9      | 10     | 55.00  | 52.38 | 53.66 |
| R-AM-CAU | 0     | 0      | 2      | 0.00   | 0.00  | 0.00  |
| R-AM-LOC | 1     | 1      | 3      | 50.00  | 25.00 | 33.33 |
| R-AM-MNR | 0     | 0      | 2      | 0.00   | 0.00  | 0.00  |
| R-AM-TMP | 6     | 1      | 4      | 85.71  | 60.00 | 70.59 |
| V        | 764   | 14     | 40     | 98.20  | 95.02 | 96.59 |

Figura 61: Niveles de precisión, recall y F-Measure del sistema Punyakanok, corpus tBrown.

Number of Sentences : 426  
 Number of Propositions : 804  
 Percentage of perfect props : 37.06

|          | corr. | excess | missed | prec.  | rec.   | F1    |
|----------|-------|--------|--------|--------|--------|-------|
| Overall  | 1423  | 603    | 754    | 70.24  | 65.37  | 67.71 |
| A0       | 501   | 69     | 65     | 87.89  | 88.52  | 88.20 |
| A1       | 448   | 218    | 228    | 67.27  | 66.27  | 66.77 |
| A2       | 79    | 66     | 68     | 54.48  | 53.74  | 54.11 |
| A3       | 1     | 4      | 11     | 20.00  | 8.33   | 11.76 |
| A4       | 5     | 1      | 10     | 83.33  | 33.33  | 47.62 |
| AM-ADV   | 47    | 35     | 96     | 57.32  | 32.87  | 41.78 |
| AM-CAU   | 3     | 4      | 5      | 42.86  | 37.50  | 40.00 |
| AM-DIR   | 19    | 13     | 34     | 59.38  | 35.85  | 44.71 |
| AM-DIS   | 9     | 20     | 13     | 31.03  | 40.91  | 35.29 |
| AM-EXT   | 0     | 1      | 5      | 0.00   | 0.00   | 0.00  |
| AM-LOC   | 30    | 39     | 55     | 43.48  | 35.29  | 38.96 |
| AM-MNR   | 37    | 43     | 73     | 46.25  | 33.64  | 38.95 |
| AM-MOD   | 90    | 18     | 1      | 83.33  | 98.90  | 90.45 |
| AM-NEG   | 43    | 3      | 7      | 93.48  | 86.00  | 89.58 |
| AM-PNC   | 3     | 7      | 14     | 30.00  | 17.65  | 22.22 |
| AM-PRD   | 0     | 0      | 1      | 0.00   | 0.00   | 0.00  |
| AM-TMP   | 56    | 55     | 56     | 50.45  | 50.00  | 50.22 |
| R-A0     | 25    | 2      | 0      | 92.59  | 100.00 | 96.15 |
| R-A1     | 16    | 3      | 5      | 84.21  | 76.19  | 80.00 |
| R-AM-CAU | 1     | 0      | 1      | 100.00 | 50.00  | 66.67 |
| R-AM-LOC | 1     | 1      | 3      | 50.00  | 25.00  | 33.33 |
| R-AM-MNR | 1     | 0      | 1      | 100.00 | 50.00  | 66.67 |
| R-AM-TMP | 8     | 1      | 2      | 88.89  | 80.00  | 84.21 |
| V        | 773   | 31     | 31     | 96.14  | 96.14  | 96.14 |

Figura 62: Niveles de precisión, recall y F-Measure del sistema Haghighi, corpus tBrown.



Number of Sentences : 426  
 Number of Propositions : 804  
 Percentage of perfect props : 36.44

|          | corr. | excess | missed | prec.  | rec.  | F1    |
|----------|-------|--------|--------|--------|-------|-------|
| Overall  | 1401  | 578    | 776    | 70.79  | 64.35 | 67.42 |
| A0       | 481   | 72     | 85     | 86.98  | 84.98 | 85.97 |
| A1       | 451   | 196    | 225    | 69.71  | 66.72 | 68.18 |
| A2       | 73    | 88     | 74     | 45.34  | 49.66 | 47.40 |
| A3       | 2     | 6      | 10     | 25.00  | 16.67 | 20.00 |
| A4       | 3     | 0      | 12     | 100.00 | 20.00 | 33.33 |
| AM-ADV   | 48    | 29     | 95     | 62.34  | 33.57 | 43.64 |
| AM-CAU   | 4     | 4      | 4      | 50.00  | 50.00 | 50.00 |
| AM-DIR   | 14    | 11     | 39     | 56.00  | 26.42 | 35.90 |
| AM-DIS   | 9     | 20     | 13     | 31.03  | 40.91 | 35.29 |
| AM-EXT   | 0     | 0      | 5      | 0.00   | 0.00  | 0.00  |
| AM-LOC   | 25    | 34     | 60     | 42.37  | 29.41 | 34.72 |
| AM-MNR   | 50    | 43     | 60     | 53.76  | 45.45 | 49.26 |
| AM-MOD   | 87    | 4      | 4      | 95.60  | 95.60 | 95.60 |
| AM-NEG   | 45    | 1      | 5      | 97.83  | 90.00 | 93.75 |
| AM-PNC   | 6     | 3      | 11     | 66.67  | 35.29 | 46.15 |
| AM-PRD   | 0     | 0      | 1      | 0.00   | 0.00  | 0.00  |
| AM-TMP   | 58    | 59     | 54     | 49.57  | 51.79 | 50.66 |
| R-A0     | 24    | 5      | 1      | 82.76  | 96.00 | 88.89 |
| R-A1     | 13    | 2      | 8      | 86.67  | 61.90 | 72.22 |
| R-AM-CAU | 0     | 0      | 2      | 0.00   | 0.00  | 0.00  |
| R-AM-LOC | 0     | 0      | 4      | 0.00   | 0.00  | 0.00  |
| R-AM-MNR | 0     | 0      | 2      | 0.00   | 0.00  | 0.00  |
| R-AM-TMP | 8     | 1      | 2      | 88.89  | 80.00 | 84.21 |
| V        | 776   | 28     | 28     | 96.52  | 96.52 | 96.52 |

Figura 63: Niveles de precisión, recall y F-Measure del sistema Marquez, corpus tBrown.

Number of Sentences : 426  
 Number of Propositions : 804  
 Percentage of perfect props : 36.44

|          | corr. | excess | missed | prec. | rec.  | F1    |
|----------|-------|--------|--------|-------|-------|-------|
| Overall  | 1339  | 477    | 838    | 73.73 | 61.51 | 67.07 |
| A0       | 471   | 59     | 95     | 88.87 | 83.22 | 85.95 |
| A1       | 451   | 176    | 225    | 71.93 | 66.72 | 69.22 |
| A2       | 74    | 75     | 73     | 49.66 | 50.34 | 50.00 |
| A3       | 1     | 9      | 11     | 10.00 | 8.33  | 9.09  |
| A4       | 5     | 2      | 10     | 71.43 | 33.33 | 45.45 |
| AM-ADV   | 42    | 17     | 101    | 71.19 | 29.37 | 41.58 |
| AM-CAU   | 4     | 2      | 4      | 66.67 | 50.00 | 57.14 |
| AM-DIR   | 13    | 10     | 40     | 56.52 | 24.53 | 34.21 |
| AM-DIS   | 4     | 12     | 18     | 25.00 | 18.18 | 21.05 |
| AM-EXT   | 0     | 0      | 5      | 0.00  | 0.00  | 0.00  |
| AM-LOC   | 24    | 23     | 61     | 51.06 | 28.24 | 36.36 |
| AM-MNR   | 32    | 34     | 78     | 48.48 | 29.09 | 36.36 |
| AM-MOD   | 73    | 5      | 18     | 93.59 | 80.22 | 86.39 |
| AM-NEG   | 40    | 2      | 10     | 95.24 | 80.00 | 86.96 |
| AM-PNC   | 4     | 7      | 13     | 36.36 | 23.53 | 28.57 |
| AM-PRD   | 0     | 0      | 1      | 0.00  | 0.00  | 0.00  |
| AM-TMP   | 59    | 36     | 53     | 62.11 | 52.68 | 57.00 |
| R-A0     | 23    | 2      | 2      | 92.00 | 92.00 | 92.00 |
| R-A1     | 13    | 4      | 8      | 76.47 | 61.90 | 68.42 |
| R-AM-CAU | 0     | 0      | 2      | 0.00  | 0.00  | 0.00  |
| R-AM-LOC | 0     | 1      | 4      | 0.00  | 0.00  | 0.00  |
| R-AM-MNR | 0     | 0      | 2      | 0.00  | 0.00  | 0.00  |
| R-AM-TMP | 6     | 1      | 4      | 85.71 | 60.00 | 70.59 |
| V        | 782   | 22     | 22     | 97.26 | 97.26 | 97.26 |

Figura 64: Niveles de precisión, recall y F-Measure del sistema Pradhan, corpus tBrown.

Number of Sentences : 426  
 Number of Propositions : 804  
 Percentage of perfect props : 29.48

|          | corr. | excess | missed | prec. | rec.   | F1    |
|----------|-------|--------|--------|-------|--------|-------|
| Overall  | 1299  | 495    | 878    | 72.41 | 59.67  | 65.42 |
| A0       | 473   | 79     | 93     | 85.69 | 83.57  | 84.62 |
| A1       | 383   | 187    | 293    | 67.19 | 56.66  | 61.48 |
| A2       | 64    | 44     | 83     | 59.26 | 43.54  | 50.20 |
| A3       | 0     | 4      | 12     | 0.00  | 0.00   | 0.00  |
| A4       | 3     | 1      | 12     | 75.00 | 20.00  | 31.58 |
| AM-ADV   | 43    | 31     | 100    | 58.11 | 30.07  | 39.63 |
| AM-CAU   | 5     | 3      | 3      | 62.50 | 62.50  | 62.50 |
| AM-DIR   | 17    | 8      | 36     | 68.00 | 32.00  | 43.59 |
| AM-DIS   | 9     | 19     | 13     | 32.14 | 40.91  | 36.00 |
| AM-EXT   | 0     | 2      | 5      | 0.00  | 0.00   | 0.00  |
| AM-LOC   | 24    | 33     | 61     | 42.11 | 28.24  | 33.80 |
| AM-MNR   | 35    | 26     | 75     | 57.38 | 31.82  | 40.94 |
| AM-MOD   | 84    | 4      | 7      | 95.45 | 92.31  | 93.85 |
| AM-NEG   | 44    | 2      | 6      | 95.65 | 88.00  | 91.67 |
| AM-PNC   | 6     | 1      | 11     | 85.71 | 35.29  | 50.00 |
| AM-PRD   | 0     | 0      | 1      | 0.00  | 0.00   | 0.00  |
| AM-TMP   | 61    | 37     | 51     | 62.24 | 54.46  | 58.10 |
| R-A0     | 23    | 5      | 2      | 82.14 | 92.00  | 86.79 |
| R-A1     | 14    | 4      | 7      | 77.78 | 66.67  | 71.79 |
| R-A2     | 0     | 1      | 0      | 0.00  | 0.00   | 0.00  |
| R-AM-CAU | 0     | 0      | 2      | 0.00  | 0.00   | 0.00  |
| R-AM-LOC | 0     | 0      | 4      | 0.00  | 0.00   | 0.00  |
| R-AM-MNR | 2     | 2      | 0      | 50.00 | 100.00 | 66.67 |
| R-AM-TMP | 9     | 2      | 1      | 81.82 | 90.00  | 85.71 |
| V        | 0     | 0      | 804    | 0.00  | 0.00   | 0.00  |

Figura 65: Niveles de precisión, recall y F-Measure del sistema Surdeanu, corpus tBrown.

ANEXO 3. ARTÍCULO ACEPTADO PARA LA CONFERENCIA TSD 2010.

# Borda-based voting schemes for Semantic Role Labeling

Vladimir Robles<sup>1,2</sup>, Antonio Molina<sup>2</sup>, and Paolo Rosso<sup>2</sup>

<sup>1</sup> Universidad Politécnica Salesiana, Cuenca, Calle Vieja 12-30, Ecuador  
vrobles@ups.edu.ec

<sup>2</sup> ELiRF, DSIC, Universidad Politécnica de Valencia, Valencia 46022, Camino de Vera  
s/n, Spain {amolina,prossso}@dsic.upv.es

**Abstract.** In this article, we have studied the possibility of applying Borda and Fuzzy Borda voting schemes to combine semantic role labeling systems. To better select the correct semantic role, among those provided by different experts, we have introduced two measures: the first one calculates the overlap between labeled sentences, whereas the second one adds different scoring levels depending on the verbs that have been parsed.

**Key words:** Semantic role labeling, Borda voting schemes.

## 1 Introduction

Previous studies shown that the semantic role labeling is a task that allows to improve the performance of many Natural Language Processing (NLP) applications. A semantic role is the underlying relationship between a syntactic constituent (consisting of a word or sequence of words) and the main verb of a sentence. The role is the function that assigns the predicate to its arguments. A clear example of what has been mentioned is shown in the following sentence: “*Hurricane-force winds demolished much of the town*”. If we review the sentence, it would have the following roles: [*Hurricane-force winds*]<sub>cause</sub> demolished [*much of the town*]<sub>theme</sub>. The syntactic constituent “*Hurricane-force winds*” is the cause that leads to a certain event, while “*much of the town*” constituent represents the argument that undergoes a change of state. The main thematic roles are: agent (argument that produces the action), experiencer (argument that is subjected to a sensory, cognitive or emotional experience), container (argument that is good or bad in a situation), location (argument representing sites), action (argument expressing some dimension) and item (argument that undergoes a change of state).

The task of semantic role labeling has been studied from several machine learning approaches, including the use of probabilistic and statistical techniques, such as Maximum Entropy or Conditional Random Fields and methodologies based on artificial intelligence such as Support Vector Machines. These methodologies depend on choosing the relevant characteristics, representing information of various kinds: lexical, syntactic and probabilistic, among other types [6].

In this paper we review the possibility of applying Borda and Fuzzy Borda voting schemes [5], to determine the feasibility of combining various systems of semantic role labeling. To accomplish this task we have worked with the data set published in the shared task of the conference CoNLL 2005 (Conference on Computational Natural Language Learning)<sup>3</sup>. We worked with the corpus tagged by the 5 best systems. We defined two measures of analysis: the level of role overlapping and the role scoring tables contained in each sentence.

The rest of the paper is organized as follows. In Section 2 we review the features of the used corpus. The Borda voting scheme and its variant Fuzzy, and the possibility of using it to combine two or more role labeling systems are described in Section 3. In Section 4 we review the steps we used to combine the results generated by the CoNLL 2005 systems. In Section 5 we show the results and analyze them. Conclusion and future work are described in Section 6.

## 2 Corpus CoNLL 2005

In CoNLL 2005, the corpus used is based on section 02 - 21 (training), section 24 (development) and section 23 (test) of the Wall Street Journal (WSJ). More precisely, the corpus is based on PropBank 1.0, which is a part of the Penn Treebank with enriched structures (predicate and argument). The corpus has different type of arguments, (i.e., Semantic Roles), Numbered Arguments (A0-A5, AA), Adjuncts (AM-), References (R-), and Verbs (V) [4].

In Table 1 we can see a list of the characteristics of the 5 best systems. The systems are ordered by F – Measure. The table lists the name of participation of each system, as well as precision, recall and F – Measure.

**Table 1.** The best five systems from the CoNLL 2005 competition.

| System      | Short Name | Precision | Recall | F-Measure |
|-------------|------------|-----------|--------|-----------|
| punyakankok | $S_1$      | 82.28%    | 76.78% | 79.44     |
| pradhan     | $S_2$      | 82.95%    | 74.75% | 78.63     |
| haghighi    | $S_3$      | 79.54%    | 77.39% | 78.45     |
| marquez     | $S_4$      | 79.55%    | 76.45% | 77.97     |
| surdeanu    | $S_5$      | 80.32%    | 72.95% | 76.46     |

## 3 Borda voting schemes

The Borda voting schemes is a technique that has been used in several NLP tasks: word sense disambiguation [3], geographical information retrieval [2], named entity recognition [1]. In this context, we consider that this methodology can improve the performance of semantic role labeling by combining different systems.

<sup>3</sup> <http://www.lsi.upc.edu/~srlconll/st05/st05.html>

For example, in this sentence of the tWSJ corpus: “*As a result, the link between the futures and stock markets ripped apart.*”, the best CoNLL 2005 three labeling systems produce the following results (Table 2):

**Table 2.** Comparison of labeling process performed by the systems  $S_1$ ,  $S_2$  and  $S_3$ .

| Constituent   | $S_1$     | $S_2$     | $S_3$     |
|---------------|-----------|-----------|-----------|
| As            | (AM-CAU*  | (AM-CAU*  | (AM-DIS*  |
| a             | *         | *         | *         |
| result        | *)        | *)        | *)        |
| ,             | *         | *         | *         |
| the           | (A1*      | (A1*      | (A1*      |
| link          | *         | *         | *         |
| between       | *         | *         | *         |
| the           | *         | *         | *         |
| futures       | *         | *         | *         |
| and           | *         | *         | *         |
| stock         | *         | *         | *         |
| markets       | *)        | *)        | *)        |
| <b>ripped</b> | (V*       | (V*       | (V*       |
| apart         | (AM-DIR*) | (AM-MNR*) | (AM-DIR*) |
| .             | *         | *         | *         |

If we want to apply a Borda voting scheme, each system should provide a determined amount of candidate roles for each sentence argument. In the example described in Table 3, the role AM-CAU is assigned to the argument “*As a result*”<sup>4</sup>. This argument must have been assigned to two or more candidate roles by each system. This allows the creation of the necessary matrices to apply the Borda voting scheme.

We calculate the general voting results considering role AM-CAU as candidate1, AM-LOC as candidate2 and AM-DIS as candidate3. For example, to calculate  $M_{S_1}$ , we fill with 1 in row 1 and column 2 which indicates that the system prefers candidate1 than candidate2. Doing so for all candidates and by filling 0 in the rest of positions, we obtain the matrix. The final vote is the sum of the rows of systems matrices.

$$M_{S_1} = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad M_{S_2} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad M_{S_3} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix} \quad Final_{Vote} = \begin{bmatrix} 3 \\ 4 \\ 2 \end{bmatrix}$$

Table 3 shows the preference candidate order for each system and the general order of the Borda voting scheme.

To apply a Fuzzy Borda voting scheme, we must add weights for each candidate role, as is shown in Table 4.

<sup>4</sup> To better illustrate our example, we add two candidate roles and we change the preference order for every role in the system  $S_2$ .

**Table 3.** Order of preference after applying Borda voting scheme.

| $S_1$  | $S_2$  | $S_3$  | Borda Preference |
|--------|--------|--------|------------------|
| AM-CAU | AM-LOC | AM-DIS | <b>AM-LOC</b>    |
| AM-LOC | AM-CAU | AM-LOC | <b>AM-CAU</b>    |
| AM-DIS | AM-DIS | AM-CAU | <b>AM-DIS</b>    |

**Table 4.** Preference order for roles labeled by each system, using weights.

| $S_1$              | $S_2$              | $S_3$              |
|--------------------|--------------------|--------------------|
| AM-CAU: <b>8.2</b> | AM-LOC: <b>7.3</b> | AM-DIS: <b>9.2</b> |
| AM-LOC: <b>7.2</b> | AM-CAU: <b>5.2</b> | AM-LOC: <b>3.2</b> |
| AM-DIS: <b>6.7</b> | AM-DIS: <b>4.7</b> | AM-CAU: <b>2.7</b> |

According to the Fuzzy Borda voting scheme, the element  $r_{j,k}^i$  (row  $j$ , column  $k$  of the matrix  $M_{S_i}$  for the role labelling systems  $S_i$ ) can be calculated using the following formula:

$$r_{j,k}^i = \frac{w_j^i}{w_j^i + w_k^i} \quad (1)$$

Using Formula 1, and the weights from Table 4, we calculate the preference matrix of Fuzzy Borda voting scheme:

$$M_{S_1} = \begin{bmatrix} 0.5 & 0.53 & 0.55 \\ 0.47 & 0.5 & 0.52 \\ 0.45 & 0.48 & 0.5 \end{bmatrix} \quad M_{S_2} = \begin{bmatrix} 0.5 & 0.42 & 0.53 \\ 0.58 & 0.5 & 0.6 \\ 0.47 & 0.39 & 0.5 \end{bmatrix} \quad M_{S_3} = \begin{bmatrix} 0.5 & 0.46 & 0.23 \\ 0.54 & 0.5 & 0.26 \\ 0.77 & 0.74 & 0.5 \end{bmatrix}$$

$$Final_{Vote} = \begin{bmatrix} 4.2 \\ 4.5 \\ 4.8 \end{bmatrix}$$

The resulting preference order, from the Fuzzy Borda scheme, is shown in Table 5.

**Table 5.** Preference order after applying Fuzzy Borda voting scheme.

| $S_1$  | $S_2$  | $S_3$  | Preference Order |
|--------|--------|--------|------------------|
| AM-CAU | AM-LOC | AM-DIS | <b>AM-DIS</b>    |
| AM-LOC | AM-CAU | AM-LOC | <b>AM-LOC</b>    |
| AM-DIS | AM-DIS | AM-CAU | <b>AM-CAU</b>    |

As we have seen, if we do not have the number of candidates or alternatives required by the Borda voting schemes, we can not apply them. We consider the following observations:

- In order to create the Borda matrix, each system must label roles as part of a single domain. All systems must assign the same candidate roles for each argument, ordered according to their preference or weights.



- The verb and its meaning are the parameters that help us to define candidate roles, which create the Borda matrix. Weights must be inferred from the level of precision, recall or F - Measure of a system.

## 4 Overlapping and scored verb analysis

### 4.1 Overlapping

The level of overlap is a measure that allows us to analyze the level of matching between two or more role labeling systems. Therefore, high value of overlapping indicates that the criteria of these systems is closer. This allows us to select those two systems that have the greatest value of matching. The system that has the highest score of verb analysis is selected.

To illustrate how we calculate the overlapping we took the sentence “*It screwed things up, said one major specialist.*” from tWSJ corpus and the roles proposed by the systems  $S_1$  and  $S_2$  (Table 6).

**Table 6.** Sentence of corpus tWSJ labeled by the systems  $S_1$  and  $S_2$ .

| Constituent | $S_1$        |           | $S_2$        |           |
|-------------|--------------|-----------|--------------|-----------|
|             | Verb screwed | Verb said | Verb screwed | Verb said |
| “           | *            | *         | *            | *         |
| It          | *            | (A1*      | (A0*)        | *         |
| screwed     | (V*)         | *         | (V*)         | (A1*      |
| things      | (A1*)        | *         | (A1*)        | *         |
| up          | *            | *)        | *            | *)        |
| ,           | *            | *         | *            | *         |
| “           | *            | *         | *            | *         |
| said        | *            | (V*)      | *            | (V*)      |
| one         | *            | (A0*      | *            | (A0*      |
| major       | *            | *         | *            | *         |
| specialist  | *            | *)        | *            | *)        |
| .           | *            | *         | *            | *         |
| Verb Score  | 0.8225       | 0.8523    | 0.8          | 0.8       |

As shown in Table 6, for the analysis of the verb “*screwed*”, the system  $S_2$  assigns A0 role to the constituent “*It*”, while the system  $S_1$  does not. For the constituent “*things*”, both systems agree to assign A1 as role. In this case, there is an overlapping in a single argument. For the verb “*said*” there is a partial overlapping in A1 role, because for the system  $S_1$  the argument is made up of “*It screwed things up*” constituents, whereas the system  $S_2$  is made up of “*screwed things up*”. For the A0 role both systems assign the same constituents. In this case, there is a partial overlap (A1) and a full overlap (A0).

To calculate the overlaps that occur in arguments consisting of a single constituent, we assign a value of 1 and we add the other arguments that have a single constituent. For the verb “*screwed*”, the overlapping value 1.

In the case of partial overlapping, we consider how many overlapping constituents of an argument (CS) and how many constituents make up that argument (CF). To calculate this value we have derived the following formula:

$$Overlap = \sum_1^N \left( \frac{CS_{S_1}}{CF_{S_1}} \right) \cdot \left( \frac{CS_{S_2}}{CF_{S_2}} \right). \quad (2)$$

Where:

- $CS_{S_1}$  and  $CS_{S_2}$  represent the constituents that overlap in the labeled argument by the systems  $S_1$  and  $S_2$ .
- $CF_{S_1}$  and  $CF_{S_2}$  represent the constituents that make up the argument of the systems  $S_1$  and  $S_2$ .
- $N$  is the total number of roles in the sentence.

The level of overlapping for the verb “*said*” is calculated as follows:

$$Overlap = overlap_{A0-Role} + overlap_{A1-Role}.$$

$$Overlap = \left[ \left( \frac{3}{3} \cdot \frac{3}{3} \right) + \left( \frac{3}{4} \cdot \frac{3}{3} \right) \right] = 1.75.$$

## 4.2 Scored verb analysis

To calculate this value we use a scoring system for each labeled role in a sentence. The basis of this metric is the overall level of precision of each system for role labeling (recall and F-Measure could also be used). For example, the system  $S_1$  labels A0 roles with a precision of 88.22%, recall of 87.88% and an F-Measure of 88.05.

Experiments were carried out using the precision values. By combining two or more role labeling systems, we are expanding the coverage level that the system has. To calculate the verb scores (Table 6), we obtain an average value of the precision that each system has to label the arguments of a specific verb. For example, in the system  $S_2$ , the score of the verb “*screwed*” is calculated as follows:  $[0.8(\text{precision of labeling the role A0}) + 0.8(\text{precision of labeling the role A1})] / 2 = 0.8$ .

For the verb “*screwed*”, the system  $S_2$  has labeled two roles, while the system  $S_1$  has labeled a single role. Our system selects the labeled roles of systems  $S_2$  and  $S_1$  for verbs “*screwed*” and “*said*”, respectively.

## 5 Experimental Results

In this section we show the main results that were obtained after applying our voting approach. We have used two schemes of overlapping and scoring, partial and complete overlapping. The first scheme does not discard the arguments that do not completely overlap. The second scheme discards those that do not have

a complete overlapping (all its constituents, similar to a simple voting scheme). In Figure 1 we see the number of arguments correctly classified by each of the combinations that we have tested. The best result is achieved combining all the systems with partial overlapping. Figure 1 shows the roles that were misclassified and also those that the system was not able to classify. The combination that produces the best results, considering the two values together (roles misclassified and non-tagged), is  $S_1$ - $S_3$ - $S_4$  with partial overlapping.

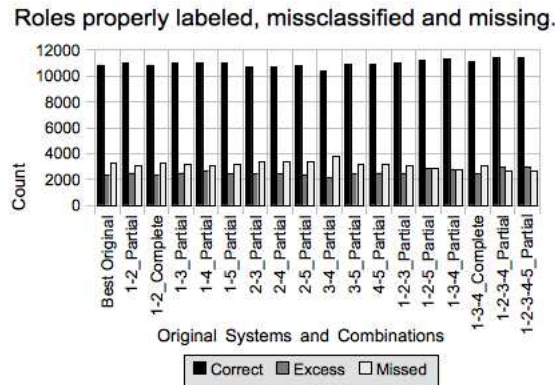


Fig. 1. Roles properly labeled, misclassified and missing.

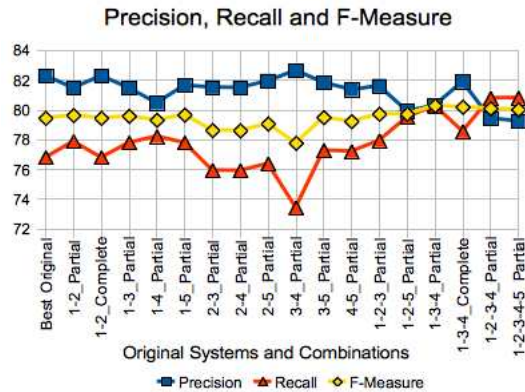
In Figure 2 we observe precision, recall and F - Measure for all system combinations. The combination that gets the best F-Measure value is  $S_1$ - $S_3$ - $S_4$  with partial overlapping. The precision is affected by the number of systems involved in the combination, because not all the systems have optimal values of this measure.

## 6 Conclusions

In this paper we have established an alternative measure of combinations between labeling systems, based on the Borda voting schemes. It has been shown that combining two or more systems together, better results can be achieved.

When we combine too many labeling systems, the precision become lower if these systems don not have similar values of precision. By contrast, the level of recall is enriched by the diversity of labeling schemes. One factor that improves the measurement of overlapping and especially the scored verb analysis, is to review the arguments that must have each verb. The implementation of this factor will help decrease the amount of roles that are misclassified or ignored.

As future work we propose to test scored verbs based on their level of matching with the arguments in PropBank and FrameNet, to apply Linear Integer Programming techniques which enrich the measurement process of overlapping



**Fig. 2.** Precision, recall and F-Measure of all system combinations.

and scored verb analysis, and include in the calculation of overlapping the values of precision, recall and F-Measure and verify their efficiency.

### Acknowledgements.

This work was partially funded by the TEXT-ENTERPRISE 2.0 TIN2009-13391-C04-03 and Telmosis PAID-06-08-3294 research projects.

### References

1. Benajiba, Y., Diab, M., Rosso, P.: Arabic Named Entity Recognition using Optimized Features Sets. EMNLP, Hawaii, USA (2008)
2. Buscaldi, D., Perea, J.M., Rosso, P. Ureña, L. A. Ferrés, D. Rodríguez, H.: Geo-TextMess: Result Fusion with Fuzzy Borda Ranking in Geographical Information Retrieval. In: CLEF, pp. 867-874. Revised Selected Papers (2008)
3. Buscaldi, D., Rosso, P.: UPV-WSD: Combining Different WSD Methods by Means of Fuzzy Borda Voting. In: Fourth International Workshop on Semantic Evaluations, pp. 434-437 (2007)
4. Carreras, X. Màrquez, L.: Introduction to the CoNLL-2005 shared task: Semantic role labeling. In: Proceedings of the Ninth Conference on Computational Natural Language Learning, pp. 152-164 (2005)
5. García-Lapresta, J.L., Martínez-Panero, M.: Borda count versus approval voting: A fuzzy approach. Public Choice, vol. 112, pp. 167-184 (2002)
6. Gildea, D., Jurafsky, D.: Automatic labeling of semantic roles. Computational Linguistics, vol. 28, pp. 245-288 (2002)

## COLOFÓN

Esta tesis se ha desarrollado usando  $\text{\LaTeX} 2_{\epsilon}$ , usando los tipos de letra Hermann Zapf's *Palatino* y *Euler*. Las listas emplean el tipo de letra *Bera Mono*, originalmente desarrollado por Bitstream, Inc. ("Bitstream Vera").

El estilo tipográfico está inspirado en el trabajo presentado en *The Elements of Typographic Style* [13]. El estilo de esta tesis esta disponible a través de  $\text{\LaTeX}$  via CTAN como "`classicthesis`".

*Versión Final:* 24 de junio de 2010 - 23:59.



## DECLARACIÓN

---

Declaro que todos los comentarios y opiniones vertidas en el presente trabajo de tesis son de exclusiva responsabilidad del autor.

*Valencia, Junio 2010*

---

Vladimir E. Robles Bykbaev