

Detección de menciones anidadas basada en expansión para el español*

Nested Mention Detection in Spanish based on Expansion

Marcel Puchol-Blasco Patricio Martínez-Barco

Natural Language Processing Research Group

University of Alicante

Alicante, Spain

{marcel,patricio}@dlsi.ua.es

Resumen: La detección de menciones es el primer módulo utilizado en los sistemas de resolución de la correferencia. Debido a que los errores se van acarreando de un módulo a otro, es importante que los resultados obtenidos por este módulo sean los mejores posibles. Dentro del campo de la detección de menciones, las menciones anidadas son las más difíciles de detectar. En este artículo presentamos un sistema de detección de menciones anidadas basado en expansión, un nuevo modelo de detección de elementos de Procesamiento de Lenguaje Natural anidados basado en aprendizaje automático. Los resultados obtenidos por nuestro sistema son superiores al 72% de medida-F en el corpus AnCora. No podemos comparar directamente nuestros resultados con otros sistemas debido a que no existen como tales, pero si comparamos la medida-F media obtenida por otros sistemas en la detección de todo tipo de menciones (no solo menciones anidadas), y que nosotros estamos tratando con las menciones más difíciles, conseguimos resultados favorables.

Palabras clave: Detección, menciones, resolución de correferencia

Abstract: Mention detection is the first module used in coreference resolution systems. Due to that, it is important that the results obtained by this module are as high as possible. Within the field of mention detection, nested mentions are the most difficult ones to detect. In this paper, we present a nested mention detection system based on expansion, a new model for detecting nested elements in NLP based on machine learning. The results obtained by our system are above the 72% in F-measure in AnCora corpus. We can not compare directly our results with other systems, since there are not exist, but if we consider that the average F-measure obtained by other systems for all mention (not only nested mentions), and that we are dealing with the most difficult mentions, we achieve good results.

Keywords: Detection, mentions, coreference resolution

1. Introduction

La resolución de la correferencia es un campo en el que se ha trabajado mucho últimamente dentro de la comunidad del Procesamiento del Lenguaje Natural (PLN). La mayor parte de áreas del PLN utilizan en alguna de sus fases información de la correferencia. Ejemplos de ello se encuentran en: Recuperación de la Información (González y Rodríguez, 2000), Extracción de la Información (Cardie, 1997), Generación de Resúmenes (Steinberger et al., 2007), Búsqueda de

Respuestas (Morton, 1999), ó incluso Extracción de Opiniones (Jakob y Gurevych, 2010).

El primer módulo utilizado dentro de cualquier sistema de resolución de la correferencia es el de detección de menciones/mentions/markables (debido a que el uso de mention es el más utilizado últimamente en inglés, a lo largo del artículo se utilizará la traducción de este término, es decir, mención). Una mención es un elemento que puede ser considerado como anáfora o como antecedente. A tal efecto, una mención puede ser: descripciones definidas, nombres demostrativos, nombres propios, aposiciones, pronombres, etc. La mayor parte de sistemas de resolución de la correferencia utilizan sistemas basados en reglas para la detección de

* Este artículo está parcialmente financiado por el Ministerio de Ciencia e Innovación - Gobierno de España (beca no. TIN2009-13391-C04-01), y Conselleria d'Educació - Generalitat Valenciana (beca no. PRO-METEO/2009/119).

menciones (Lappin y Leas (1994), Soon et al. (2001), ó Stoyanov et al. (2010)); o simplemente se basan en la extracción de menciones correctas dentro de un corpus. De hecho, hasta dónde conocemos, no existe ningún módulo ó sistema de detección de menciones basado en aprendizaje automático.

Debido a que la detección de menciones es el primer módulo utilizado en la resolución de la correferencia, es importante que los resultados obtenidos por dicho módulo sean lo más altos posibles. De hecho, Uryupina (2008) muestra en su artículo cómo hasta un 35% de errores de cobertura y un 20% de errores de precisión son debidos a errores producidos en el módulo de detección de menciones. Si profundizamos un poco más en el tema y tenemos en cuenta los resultados de los módulos de detección de menciones para el español obtenidos en Recasens et al. (2010) (la media de medida-F es del 72%), vemos que existe una clara necesidad de ahondar en la detección de menciones y mejorar los sistemas existentes.

Dentro del campo de la detección de menciones, las menciones anidadas son las más difíciles de detectar. Recientemente, problemas asociados con la detección de elementos anidados han sido estudiados (Byrne (2007) ó Finkel y Manning (2009) en reconocimiento de entidades nombradas).

En el estado de la cuestión de la detección de elementos de PLN, el modelo IOB es el más utilizado (mirar Abney (1996), Hacioglu et al. (2005), y Pradhan et al. (2008)). Sin embargo, creemos que este modelo posee ciertas deficiencias a la hora de detectar elementos anidados.

Por todas estas razones, en este artículo hemos propuesto un modelo alternativo para la detección de elementos anidados.

En la sección 2, describimos nuestro modelo. En la sección 3, las características utilizadas en el sistema de detección de menciones para español son descritas. La sección 4 muestra los resultados experimentales de nuestro sistema. Las conclusiones y trabajo futuro son mostradas en la última sección.

2. *Detección de menciones basada en expansión*

Para la detección de menciones, proponemos un modelo alternativo basado en 2 fases:

1. **Fase 1 - detección de elementos**

no anidados: detección de elementos no anidados utilizando el modelo clásico IOB; y

2. **Fase 2 - detección de elementos anidados:** recursivamente expandimos las menciones anteriormente anotadas hasta que no se detecta ninguna otra mención.

Consideramos que la detección de menciones no anidadas (fase 1) ya ha sido suficientemente explorada dentro del PLN (tal y como se ha indicado en la sección anterior). Debido a ello, este artículo se centra únicamente en la detección de menciones anidadas (fase 2).

2.1. **Aprendizaje**

El modelo propuesto para detectar menciones anidadas se basa en expansión, es decir, se basa en tomar los elementos anteriormente anotados y expandirlos (expandir sus límites) hasta crear un nuevo elemento. Para conseguir tal finalidad, representamos las anotaciones correctas de menciones de un corpus en forma de árbol (como si del resultado de un análisis sintáctico se tratase). Cada uno de los niveles de este árbol estará compuesto de tokens y de menciones anotadas en la fase 1 o en iteraciones anteriores (las cuales llamamos grupos).

Dado el esquema de representación especificado, el algoritmo de aprendizaje se basa en tomar cada uno de los niveles del árbol. Al inicio del procesamiento de cada nivel, clasificamos las menciones anotadas en el nivel anterior según la siguiente ontología de tipos de menciones¹: Verbal (G_VERBAL), Fecha (G_DATE), Número (G_NUMBER), Determinante Posesivo (G_DET_POSS), Determinante demostrativo (G_DET_DEM), Pronombre Relativo (G_PRONOUN_REL), Pronombre Posesivo (G_PRONOUN_POSS), Pronombre Personal (G_PRONOUN_PERS), Descripción Definida (G_DEF), Descripción Indefinida (G_INDEF), Nombre Propio (G_PROPER) y Nombre Común (G_COMMON).

Posteriormente, tomamos cada una de las nuevas menciones (para mayor comprensión trataremos a esta mención a lo largo de la sección como MENT) de los que vamos a aprender. Para una nueva mención, seleccionamos

¹Esta clasificación ha sido obtenida estudiando varios corpora, y la representación y forma de expansión de cada uno de los diferentes tipos de menciones.

cada uno de los grupos que contiene y los tratamos de manera individual para la expansión (para mayor comprensión trataremos a este grupo como GRP_{BASE}). Tomando como límite de aprendizaje aquellos elementos (los cuales pueden ser tokens o grupos) que conforman la nueva mención y una ventana de 4 elementos tanto por la izquierda como por la derecha, dividimos el aprendizaje en: aprendizaje de elementos por la izquierda y aprendizaje de elementos por la derecha. Para cada elemento, asignaremos una etiqueta IOB (Inside - Dentro, Outside - Fuera y Begin - Inicio) a los elementos adyacentes al grupo seleccionado como base (GRP_{BASE}). Cuando un elemento se encuentra entre GRP_{BASE} y el límite de la nueva mención (MENT), asignamos una etiqueta I a dicho elemento. Cuando un elemento coincide con el límite de la nueva mención (MENT), asignamos una etiqueta B a dicho elemento. Finalmente, cuando un elemento está fuera de los límites de la nueva mención (MENT), asignamos una etiqueta O a dicho elemento.

Acorde con este modelo, una instancia de aprendizaje se representa como el par [GRP_{BASE} , elemento adyacente]. Esta representación permite utilizar características de aprendizaje automático más sofisticadas que el modelo clásico IOB, ya que en dicho modelo sólo tendríamos un elemento (el elemento en sí) sobre el que aplicar características.

Para decrementar el número de instancias de aprendizaje de elementos etiquetados con O, sólo aquellos elementos que se encuentren dentro de una ventana determinada, son entrenados (no todos los elementos encontrados dentro de una misma oración que se encuentren fuera de la nueva mención).

Finalmente, cuando todo el árbol ha sido procesado y entrenado el sistema de aprendizaje automático, las menciones encontradas en el último nivel son clasificadas (de igual modo que si se tratasen de elementos dentro del árbol). Para estas menciones, se realizará el proceso de aprendizaje, pero etiquetando los elementos adyacentes con valor de O (indicando, así, que esos elementos no poseen a otra mención que los contenga).

Para entender mejor el proceso de aprendizaje, vamos a realizar los pasos explicados anteriormente en el siguiente ejemplo de oración:

[₁ El doble [₂ que [₂] se enfren-

tará a [₃ la pareja Sandon_Stolle-Mark.Woodforce [₃] [₁] estará formado por [₄ [₅ Guga [₅] y [₆ Jaime.Oncins [₆] [₄].

La figura 1 muestra el proceso de aprendizaje seguido para este ejemplo (se toma en cuenta un tamaño de ventana de 4 elementos). Inicialmente, en la fase 1, se detectarían los elementos no anidados de la oración (“la pareja Sandon_Stolle-Mark.Woodforce”, “Guga” y “Jaime.Oncins”). Los tokens contenidos en las menciones no anidadas son agrupados y clasificados, obteniendo la oración mostrada en la figura. Una vez detectadas las menciones no anidadas, procedemos con el proceso del primer nivel del árbol (menciones 1 y 4). Para la mención 1, tomamos los grupos que lo componen, es decir G_REL y G_PROPER₁, y aprendemos la expansión de cada uno de ellos. La expansión para G_REL se muestra en la primera oración de ejemplo de la figura. Como instancias de aprendizaje del lado izquierdo, tendremos los pares [G_REL, El]=B y [G_REL, doble]=I; mientras que las instancias del lado derecho serían: [G_REL, se]=I, [G_REL, enfrentará]=I, [G_REL, a]=I, [G_REL, G_PROPER₁]=B, [G_REL, será]=O, [G_REL, el]=O, [G_REL, formado]=O y [G_REL, por]=O. Para el resto de grupos del mismo nivel se realizaría el mismo proceso. Una vez completado el primer nivel, se agrupan los elementos detectados en el nivel anterior, obteniendo el resultado mostrado en la figura. Como ya no existen nuevas menciones sobre estos grupos (hemos alcanzado el nivel máximo del árbol de menciones), se etiquetan los elementos adyacentes como O y se realiza el mismo aprendizaje.

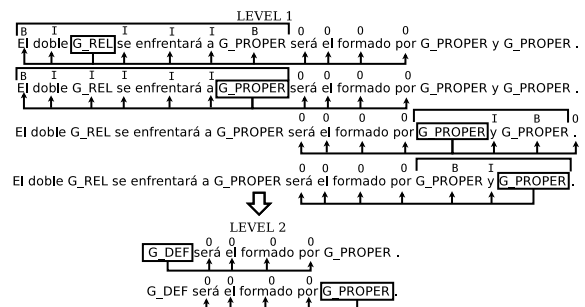


Figura 1: Proceso de aprendizaje de las menciones anidadas.

2.2. Anotación

Inicialmente, al igual que en la fase de aprendizaje, tomamos una oración y la anotamos con las menciones no anidadas (fase 1). Estas menciones son agrupadas y clasificadas. Posteriormente seleccionamos aquél grupo que se encuentra más bajo en el árbol sintáctico de la oración como grupo base sobre el que expandir. Una vez seleccionado, tomamos los elementos adyacentes y los clasificamos con el modelo aprendido en la fase de entrenamiento. Para cada lado del grupo, tomamos el último elemento anotado con la etiqueta I ó con la etiqueta B como elemento que representa el final de la expansión para cada lado. Es posible que el elemento más adyacente al grupo haya sido etiquetado como O. En caso de que tal etiquetación se de en ambos lados, se tomará como que no existe una nueva mención de la expansión del anterior. Si esta etiquetación sólo sucede en uno de los lados, para ese otro lado se tomará como límite el propio grupo base, obteniendo una nueva mención. Este proceso se ejecutará de manera recursiva hasta que todos los grupos existentes en una iteración dada sean no expansibles.

3. Características de aprendizaje automático

Debido a que las instancias de aprendizaje automático son representadas como el par $[GRP_{BASE}, \text{elemento adyacente}]$, las características de las instancias pueden estar relacionadas con el grupo base, con el elemento adyacente o con ambos.

Las siguientes características son las que se han utilizado en el sistema de español. Sin embargo, las mismas pueden ser extrapoladas a otros idiomas.

- Características asociadas a los elementos adyacentes
 - **Patrones léxicos:** información de mayúsculas/minúsculas y de tipo (numérico, alfanumérico, letras u otro).
 - **Etiquetas morfológicas**
 - **Etiquetas de lista:** el elemento actual pertenece o no a una lista de palabras (las listas disponibles son: geográfica, personas, corporaciones y temporal).

- **Información del núcleo sintáctico:** Información morfológica y morfosintáctica del núcleo sintáctico del actual elemento.
 - **Información del cabeza del núcleo sintáctico:** Información morfológica y morfosintáctica del núcleo sintáctico del núcleo sintáctico del actual elemento.
 - **Compatibilidad de WordNet:** en caso de que el elemento actual sea un nombre común, compatibilidad entre dicho elemento y núcleo sintáctico del mismo (siempre que el núcleo sintáctico también sea un nombre común).
 - **WordNet:** en caso de que el elemento actual sea un nombre común, nombre de la clase de WordNet para dicho elemento. El nombre de la clase corresponde al nivel más alto de la jerarquía de WordNet.
 - **Tokens:** si el elemento actual es un símbolo, su etiqueta de símbolo; si el elemento actual es un adverbio o preposición, la palabra que conforma al token.
- Características asociadas al grupo base o a los elementos adyacentes
 - **Posición:** el elemento se encuentra al principio o al final de la oración; el elemento se encuentra al inicio, dentro o al final de unos paréntesis o corchetes.
 - **Información del grupo:** si el elemento es un grupo o es un token; etiqueta de la ontología que describe el tipo de mención.
 - **Información morfosintáctica:** en caso de que el elemento sea un grupo, indica si el grupo contiene a un nombre dentro del árbol sintáctico; información morfosintáctica del elemento.
 - Características asociadas tanto al grupo base como a los elementos adyacentes
 - **Es núcleo sintáctico:** indica si el elemento adyacente actual es el núcleo sintáctico del grupo base o viceversa.

- **Características en línea:** resultados anotados en los resultados anteriores.
- **Posición:** distancia entre el elemento adyacente actual y el grupo base.
- **Información morfosintáctica:** el elemento adyacente actual está dentro de la información morfosintáctica del grupo base.

Para representar las características anteriores, se utilizan trigramas. A modo de ejemplo, la característica del núcleo sintáctico se aplicará para el token actual con una ventana de tres elementos, es decir, que entrarían los elementos de la posición -3 a la posición 3 (incluidos).

4. Evaluación

4.1. Corpus

El único corpus existente que tiene anotadas las menciones en español es el corpus AnCora. El corpus AnCora para español (Recasens y Martí, 2010) está extraído de periódicos y agencias de noticias y consiste en 500k tokens. Además, se ha anotado manualmente: argumentos y roles temáticos, clases semánticas, entidades nombradas, sentidos de WordNet y relaciones de correferencia.

Debido a que nuestra intención es evaluar nuestro sistema de datos obtenidos de manera automática, utilizamos el corpus ofrecido en la tarea de correferencia del SemEval-2010 (Recasens et al., 2010), el cuál está basado en el corpus AnCora, anotando de manera automática: la información morfológica y morfosintáctica mediante la herramienta FreeLing²; y la información de roles semánticos y de dependencias mediante la herramienta JoinParser³. Debido a que las entidades nombradas no están anotadas automáticamente en este corpus, hemos decidido no utilizar características relacionadas con las entidades nombradas en nuestro sistema. Estadísticas relacionadas con las menciones para este corpus se muestran en la tabla 1.

4.2. Resultados

Para evaluar nuestro sistema, hemos utilizado el clasificador J48 para el sistema de aprendizaje automático. Debido a que solo queremos evaluar el sistema de detección

| | Ment. | Anid. | No anid. |
|------|--------|--------------|--------------|
| Ent. | 78.779 | 25.521 (32%) | 53.258 (68%) |
| Des. | 12.232 | 3.962 (32%) | 8.270 (68%) |
| Test | 14.133 | 4.469 (31%) | 9.664 (69%) |

Cuadro 1: Estadísticas sobre las menciones en el corpus AnCora. Nota: las abreviaturas ent. y des. son el corpus de entrenamiento y el de desarrollo del corpus AnCora.

de menciones anidadas, hemos utilizado las menciones no anidadas correctas existentes en el corpus para los resultados de la fase 1. A partir de estas menciones, se ha ejecutado el sistema para la obtención de las menciones anidadas.

Para entrenar nuestro sistema, los corpus de entrenamiento y de desarrollo han sido utilizados; mientras que el corpus de test es el utilizado para evaluar el sistema.

En la tabla 2 se muestran los resultados obtenidos por nuestro sistema. Estos resultados son extraídos de la herramienta de scorer proporcionada por la tarea de correferencia del SemEval-2010. Cabe destacar de esta herramienta que, en caso de no coincidencia exacta entre el corpus correcto y la anotación realizada, se tendrá en cuenta que dicha anotación será correcta un 50% siempre y cuando en la anotación se incluya el núcleo sintáctico del corpus correcto. Hemos separado los resultados de detección globales (Overall) de aquellos en los que concuerdan los límites exactamente (Exact matching). Esta división se ha realizado debido a que muchas de las diferencias entre el corpus correcto y las anotaciones realizadas por nuestro sistema se debían a un solo determinante o signo de puntuación que estaba anotado en el corpus correcto y que no anotaba nuestra herramienta. Creemos que esta diferencia no es significativa para un sistema posterior de resolución de la correferencia, con lo que es necesario mostrar ambos resultados.

Nos es imposible comparar estos resultados con otros obtenidos por diferentes modelos por dos distintas razones: 1) no existen resultados para sistemas que utilizan modelos de aprendizaje automático para la detección de menciones; y 2) ninguno de los sistemas existentes separa los resultados obtenidos entre menciones anidadas y no anidadas. No obstante, consideramos que los resultados obtenidos por nuestro sistema son in-

²<http://www.lsi.upc.es/nlp/freeling>

³<http://www.lsi.upc.edu/xlluis/?x=cat:5>

| | Overall | Exact matching |
|------------|-------------|----------------|
| Correcto | 4022 | 3306 |
| Incorrecto | 0 | 716 |
| Olvidado | 447 | 447 |
| Espurio | 730 | 730 |
| Precisión | 0.85 | 0.70 |
| Cobertura | 0.90 | 0.74 |
| F1 | 0.87 | 0.72 |

Cuadro 2: Resultados obtenidos en la detección de menciones anidadas.

directamente comparables a los obtenidos en Recasens et al. (2010). Esto se debe a que tanto en el caso de Recasens et al. (2010) como en el nuestro, utilizamos el mismo corpus y el mismo sistema de evaluación. Si tenemos en cuenta que la media de la medida-F de estos sistemas es del 71 % para todos los tipos de menciones (tanto las anidadas como las no anidadas) y que nosotros obtenemos un 72 % en las menciones más difíciles de detectar (las menciones anidadas), podemos concluir que nuestros resultados son acordes al estado de la cuestión. Además, estos sistemas son basados en reglas y específicos del español, mientras que los nuestros pueden ser, mediante una fácil adaptación, extrapolados a otros idiomas.

Después de analizar los errores obtenidos por nuestro sistema, nos hemos dado cuenta de la complejidad presentada por el corpus utilizado. En este corpus, no se han anotado: predicados, aposiciones, improperios, sintagmas nominales atributivos y modismos. Debido a que detectar algunos de los citados elementos requiere de conocimiento del mundo, es muy difícil diferenciar estos casos de los que sí deben ser anotados; sobre todo, debido a la ambigüedad existente en algunos de los mismos.

Las debilidades que se han hallado después de comprobar los resultados son, básicamente: 1) la independencia entre los lados izquierdo y derecho genera, algunas veces, anotaciones espurias debido a que, en determinados casos, si no se anotan ambos lados, no deberá anotarse ninguno (ejemplo de ello se encuentra en el pronombre relativo “que”); y 2) los errores en niveles tempranos se acumulan hasta los niveles superiores.

Debido a todos los problemas encontrados, una solución plausible podría ser la de utilizar un módulo de detección de la anaforicidad

(mirar Ng (2004) para mayor información) y así descartar aquellas menciones que seguro que no son anafóricas; reduciendo el número de resultados espurios obtenidos por el sistema.

5. Conclusiones

Hemos presentado un nuevo modelo basado en aprendizaje automático y expansión para la detección de menciones anidadas, describiendo el sistema con detalle en todas las fases de detección. Nuestros resultados no pueden compararse directamente a otros sistemas de detección de menciones, pero consideramos que, ya que estamos tratando con las menciones más difíciles de detectar (las menciones anidadas), hemos conseguido resultados favorables.

Actualmente, estamos desarrollando un modelo basado en IOB para poder realizar comparaciones entre ambos sistemas. Como trabajo futuro, queremos probar este modelo para otros idiomas distintos del español, mejorando las características utilizadas y analizando qué características son comunes a todos los idiomas y cuáles no. Además queremos expandir este modelo a otras tareas de PLN para la detección de elementos anidados, tales como la detección de entidades nombradas.

Bibliografía

2008. *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2008, 26 May - 1 June 2008, Marrakech, Morocco*. European Language Resources Association.
- Abney, Steven. 1996. Partial parsing via finite-state cascades. *Natural Language Engineering*, 2(4):337–344.
- Byrne, Kate. 2007. Nested named entity recognition in historical archive text. *International Conference on Semantic Computing*, 0:589–596.
- Cardie, Claire. 1997. Empirical methods in information extraction. *AI magazine*, 18:65–79.
- Finkel, Jenny Rose y Christopher D. Manning. 2009. Nested named entity recognition. En *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume*

- 1, EMNLP '09, páginas 141–150, Stroudsburg, PA, USA. Association for Computational Linguistics.
- González, José Luis Vicedo y Antonio Ferrández Rodríguez. 2000. Applying anaphora resolution to question answering and information retrieval systems. En *Proceedings of the First International Conference on Web-Age Information Management*, WAIM '00, páginas 344–355, London, UK. Springer-Verlag.
- Hacioglu, Kadri, Benjamin Douglas, y Ying Chen. 2005. Detection of entity mentions occurring in english and chinese text. En *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, páginas 379–386, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Jakob, Niklas y Iryna Gurevych. 2010. Using anaphora resolution to improve opinion target identification in movie reviews. En *Proceedings of the ACL 2010 Conference Short Papers*, ACLShort '10, páginas 263–268, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Lappin, Shalom y Herbert J. Leass. 1994. An algorithm for pronominal anaphora resolution. *Computational Linguistics*, 20(4):535–561.
- Morton, Thomas S. 1999. Using coreference in question answering. En *In Proceedings of the 8th Text REtrieval Conference (TREC-8)*, páginas 85–89.
- Ng, Vincent. 2004. Learning noun phrase anaphoricity to improve coreference resolution: issues in representation and optimization. En *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, ACL '04, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Pradhan, Sameer S., Wayne Ward, y James H. Martin. 2008. Towards robust semantic role labeling. *Comput. Linguist.*, 34:289–310, June.
- Recasens, Marta, Lluís Màrquez, Emili Sapena, M. Antònia Martí, Mariona Taulé, Véronique Hoste, Massimo Poesio, y Yannick Versley. 2010. Semeval-2010 task 1: Coreference resolution in multiple languages. En *Proceedings of the 5th International Workshop on Semantic Evaluation*, SemEval '10, páginas 1–8, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Recasens, Marta y M. Antònia Martí. 2010. Ancora-co: Coreferentially annotated corpora for spanish and catalan. *Lang. Resour. Eval.*, 44:315–345, December.
- Soon, Wee Meng, Hwee Tou Ng, y Daniel Chung Yong Lim. 2001. A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics*, 27(4):521–544.
- Steinberger, Josef, Massimo Poesio, Mijail A. Kabadjov, y Karel Jeek. 2007. Two uses of anaphora resolution in summarization. *Inf. Process. Manage.*, 43:1663–1680, November.
- Stoyanov, Veselin, Claire Cardie, Nathan Gilbert, Ellen Riloff, David Buttler, y David Hysom. 2010. Coreference resolution with reconcile. En *Proceedings of the ACL 2010 Conference Short Papers*, ACLShort '10, páginas 156–161, Stroudsburg, PA, USA. Association for Computational Linguistics.