INSTITUTO POLITÉCNICO NACIONAL

CENTRO DE INVESTIGACIÓN EN COMPUTACIÓN

Laboratorio de Lenguaje Natural y Procesamiento de Texto

Coreference Resolution using Methods of Word Sense Disambiguation

TESIS

QUE PARA OBTENER EL GRADO DE:

MAESTRÍA EN CIENCIAS DE LA COMPUTACIÓN

P R E S E N T A:

Ing. Cristina Alicia Díaz Jiménez

Director de tesis:
Dr. Alexander Gelbukh

Mexico, D.F. Diciembre 2014

SIP-14

INSTITUTO POLITÉCNICO NACIONAL

SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

ACTA DE REVISIÓN DE TESIS

En la Ciudad de México, D.F. siendo las 12:00 horas del día 02 del mes de diciembre de 2014 se reunieron los miembros de la Comisión Revisora de la Tesis, designada por el Colegio de Profesores de Estudios de Posgrado e Investigación del:

Centro de Investigación en Computación

para examinar la tesis titulada:

"Coreference resolution using methods or Word sense disambiguation"

Presentada por la alumna:

DÍAZ	**JIMENEZ**	**CRISTINA ALICIA**
Apellido paterno	Apellido materno	Nombre(s)

Con registro:

B	1	2	1	0	3	2

aspirante de: **MAESTRÍA EN CIENCIAS DE LA COMPUTACIÓN**

Después de intercambiar opiniones los miembros de la Comisión manifestaron ***APROBAR LA TESIS***, en virtud de que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.

LA COMISIÓN REVISORA

Director de Tesis

Dr. Alexander Gelbukh

Dr. Sergio Suárez Guerra

Dr. Grigori Sidorov

Dra. Olga Kolesnikova

Dra. Sofía Natalia Galicia Haro

Dr. Francisco Hiram Calvo Castro

PRESIDENTE DEL COLEGIO DE PROFESORES

Dr. Luis Alfonso Villa Vargas

ESTADOS UNIDOS MEXICANOS
INSTITUTO POLITECNICO NACIONAL
CENTRO DE INVESTIGACION EN COMPUTACION
DIRECCION

INSTITUTO POLITÉECNICO NACIONAL

SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

CARTA CESIÓN DE DERECHOS

En la Ciudad de México el día 05 diciembre del 2014 , el (la) que suscribe Cristina Alicia Díaz Jiménez alumno (a) del Programa de Maestría en Ciencias de la Computación con número de registro B121032 , adscrito a Laboratorio de Procesamiento de Lenguaje Natural , manifiesta que es autor (a) intelectual del presente trabajo de Tesis bajo la dirección de Dr. Alexander Gelbukh y cede los derechos del trabajo intitulado Coreference Resolution using Word Sense Disambiguation Methods , al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo a la siguiente dirección cdiaz_b12@sagitario.cic.ipn.mx . Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.

Cristina Alicia Díaz Jiménez
Nombre y firma

Dedicado a el chip,
a mi familia y
a mis amigos.

Resumen

El problema de resolución de correferencia, junto a la desambiguación de sentidos, es una de las tareas centrales que el PLN busca resolver. A pesar de las aplicaciones e importancia de estos dos problemas, en otras áreas del PLN, es muy poco común que se estudien a la par.

En esta tesis, se propone una técnica inspirada en la desambiguación de sentidos que se combina con algoritmos del aprendizaje automático. El resultado final muestra que la propuesta es prometedora e inspira a seguir desarrollando este enfoque.

Abstract

The corefence resolution problem is one of the main tasks to resolve in NLP, as well as the word sense disambiguation task. Despite their importance and numerous applications in different areas of NLP, they are rarely studied together.

A new technique for coreference resolution is proposed in this thesis. Our proposed technique is inspired by words sense disambiguation and machine learning. The final results are promising and encourage to further develop our approach.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Nowadays, large amounts of information are kept in different ways as it has been stored through the history, the usage of the computer has allowed to human beings to save as much information as possible. This had lead into the problem of knowing the meaning of all the data stored. Actually, every day new ways of interaction between human beings and computers are being developed. These interactions have even changed the way people interacts with other people using computers or electronic devices.

Natural Language Processing (NLP) is a multidisciplinary task involving Computer Science, Linguistics and other disciplines. Its main goal is to determine models that allow the comprehension of text by a computer using the human language in order to get information from unstructured text.

Two of the core problems of NLP are Coreference Resolution and Word Sense Disambiguation (WSD). The first one aims to identify nominal phrases that recall the same entity in the real word, while the second one aims to find the correct sense of a word which owns several meanings.

- Information Extraction
- Question Answering
- Summarization
- Spam Filtering
- Machine Translation

Given the importance of these two problems, new methods to solve them have been proposed through the last years, aiming to find different ways to solve these problems.

1.1 Objectives

1.1.1 General Objectives

The general objective of this work is to explore the possibility to develop an algorithm developed in the Word Sense Disambiguation that finds coreference chains in Spanish.

1.1.2 Particular Objectives

The particular objectives of this work include:

- Determine coreference chains in the Corpus AnCora-Esp
- Avoid the usage of straight syntactic restrictions.
- Apply semantic similarity (Wu and Palmer distance) as the main criterion for classification.
- evaluate the obtained results and decide if this is a promising approximation

1.2 Contributions

The main contributions of this thesis can be summarized as follows:

- A novel algorithm for coreference resolution based on the ideas from the word sense disambiguation field,
- Tuning of the parameters of the proposed algorithm,
- Evaluation of its performance.

1.3 Scope

A new method to figure out coreference chains in Spanish, based on a WSD algorithm is presented, we aim to see if this method is promising, especially in the AnCora Corpus.

Seven features to train a classifier are defined, including a new one, the Wu and Palmer distance between the elements of each noun phrase, and others proposed by Sonn are modified or not used.

Our future work includes following with this approach and trying with different semantic similarity measures in order to get better results.

1.4 Organization of the Document

In the following chapters, the developed work will be presented.

- Chapter 1 contains an introduction to the topic and general description of the document.
- Chapter 2 discusses to previous knowledge that is required to understand this work.
- Chapter 3 shows some of the most important or recent contributions to the Coreference Resolution and Word Sense Disambiguation fields.
- Chapter 4 explains the methodology that was followed.
- Chapter 5 shows the results obtained in this work.
- Chapter 6 includes some discussion.
- Chapter 7 concludes the document.

Chapter 2

Antecedents

2.1 Introduction

Coreference is a linguistic phenomenon with many interesting properties. First of all, we could mention it as the hyperlink of natural language, where different linguistic elements (nouns, pronouns, etc.) evoke the same entity. Second, is an element of cohesion between the ideas to be transferred. And third, it tells us about the complex structure language and the goal that is for the mechanical understanding of language to comprehend it such as human brains does it.

In the following chapter, proper definitions of coreference, its types, and its difference with another phenomena are mentioned, as well different approaches to solve it.

2.2 Coreference

Coreference resolution is a linguistic problem which aims to select the mentions of a noun phrase **(NP)** and determine which of them refers to the same real-world entity. It is related to the *Anaphora Resolution problem*, whose goal is to determine the antecedent of an anaphoric.

In 1997, Hirschman et al. defined the relation of coreference as the one held between two noun phrases if they refer to the same entity. In other words, let us assume that α_1 and α_2 are occurrences of noun phrases **(NP)** and let us assume that both have a unique reference in the context in which they occur, so they are unambiguous in the same context, then:

Definition 1. *α_1 and α_2 corefer if and only if Reference(α1) = Reference(α2), where*

Reference(α_n) means 'the entity referred to by α_n'.

2.2.1 Anaphora and Coreference

An *Anaphora* ($\alpha\nu\alpha$: back, behind and $\varphi o\rho\alpha$: carry) is a linguistic phenomenon very close to the coreference. Hallidyay and Hasan (1976) gave the classic definition of anaphora based on the notion of cohesion: anaphora is cohesion (presupposition) which points back to some previous item.

An *anaphor* is the pointing back element, the entity to witch an anaphor refers to is called *antecedent*. The process of determining the antecedent of an anaphor is called anaphora resolution. For example:

Mary fue al cine el jueves. A ella no le gustó la película...
Mary went to the cinema on Thursday. She didn't like the film...

In the previous sentence, "ella" *(she)* is the anaphor and "Mary" is the antecedent, both refer to the same entity in the real-world. It is important to notice that the anaphor (Mary) is not a single noun, it is a NP.

Most of the time, Coreference Resolution is thought as an Anaphora problem. Actually they are very similar phenomena, but not exactly the same. Coreference occurs when both the antecedent and the anaphor are used as referring 2 expressions and having the same referent in the real-world.

The following examples give an idea of the difference between both phenomena:

a. Llegaron con buenos resultados hasta los torneos de la final, pero en ellos perdieron.
They got with good results to the final competitions, but they lost in them.

b. Los mejores equipos de la NBA son mejores que los nuestros.
The best teams of the NBA are better than ours.

c. La capital de Francia...en París...
The capital of France...in Paris...

The main difference between Coreference and Anaphora is that not all anaphoric relations are coreferential (b), nor are all coreferential relations anaphoric (c).

Coreference is a symmetrical and transitive relation, this means that all the mentions that corefer to the same entity form a simple partition of a set of the available NP. In the other hand, Anaphora is a nonsymmetrical and no transitive relation: if NP_1 is anaphoric to NP_2 then usually, NP_2 is not anaphoric to NP_1.

Another difference lays in the context interpretation, this means that an anaphoric pronoun cannot be interpreted without information about where it occurs.

It is possible that anaphoric and coreferential relations coincide, for example:

W.J. Clinton tomó el juramento; luego él tomó una respiración profunda.
W.J. Clinton took the oath; then he took a deep breath.

2.2.2 MUC Definition

The first time the Coreference solution was presented as a task of NLP was during MUC-6 in 1995, increasing the numbers of projects related to coreference resolution in Information Extraction (Bangalore and Baldwin 1996, Gaizauskas et al. 1998, Kameyama 1997). MUC defines Coreference in the following way:

Definition 2. *An identity-of-reference relation between two textual elements known as Markables.*
Markables can consist of definite noun phrases, demonstrative NP, proper names, and appositives.

Defining coreference as a formal task is not trivial. It is said it turns into a formal task since it can be evaluated. This definition was provided by the MUC-6 coreference task which:

1. Only consider references by noun phrases to other noun phrases and not reference to events.
2. Only consider identity relations and not part-whole relations.
3. Exclude split antecedents.
4. Include predicate nominals and apposition.

However, the two notions, coreference and anaphora are not properly distinguished in the MUC data, which has led to a TD difficult to understand and apply. The works of van Deemter and Kibble criticize the MUC Task Definition for violating the relation of coreference and mixing it with anaphora.

2.2.3 Kinds of Coreference

There are different ways Coreference presents, in the following section we discuss this classification. As mentioned before, MUC does not consider all of the following.

2.2.3.1 Noun Phrase Coreference

This is the most common way to find coreference. In this kind of coreference the antecessor is substituted by a personal pronoun.

For example:

Lo ví, era Gilberto.
I saw him, it was Gilberto

The NP Coreference is found in *lo* and *Gilberto*

2.2.3.2 Demonstrative Noun Phrase Coreference

This kind of coreference occurs when a demonstrative pronoun is used. Such a demonstrative pronoun may refer to a person, object or idea.

Mientras esto sucedía, la princesa dormía en el palacio *While this happened, the princess was sleeping in the palace.*

In this sentence, the coreference ocurrs between *esto* and the event *la princesa dormía en el palacio.*

This kind of coreference also uses possessive pronoun and quantifiers.

2.2.3.3 Lexical Coreference

This kind of coreference uses synonyms, hypernyms and hyponyms. Synonyms are words who have similar meaning, hypernyms are words that include several specific meanings while hyponyms include similar semantic meaning words.

Ese adolescente era una promesa. El joven era muy creativo.
That teenager was a promise. The young man was very creative

In this sentence *Ese adolescente* and *El joven* evoke the same entity.

Había peras, manzanas y naranjas, las frutas más comunes.
There were pears, apples and oranges, the most common fruit.

In this case, *peras, manzanas y naranjas* corefers to *fruit.* All of them are members of the semmantic class *fruit.*

According to Sidorov and Gelbukh, there are two cases where the coreference appears:

1. Direct coreference, as shown in the discourse
   "he visto una nueva casa ayer. Su cocina era excepcionalmente grande" (su = de la casa)
   I have seen a new house yesterday. Its kitchen was exceptionally big (its = of the house)

2. Indirect coreference as shown in the discourse
   "he visto una nueva casa ayer. La cocina era excepcionalmente grande" (la cocina = de la casa)
   I have seen a new house yesterday. The kitchen was exceptionally big (the kitchen = of the house)

In Spanish, the direct coreference appears when personal pronouns are used, while the indirect coreference appear when using determinate articles.

2.3 NP Coreference Resolution

NP Coreference might seem a simple problem, but it is not. It has been shown to be one of the most difficult problems in Natural Language Processing (NLP). Despite its difficulty NP Coreference has a large number of applications in NLP, some of them are: Information Extraction, Questions Answering, Summarization, and Similarity Analysis, summarization etc. The importance of this linguistically phenomenon is such, that it has crossed borders with other disciplines, turning into a main tool for bioinformatics i.e. find the expression of genes for cancer research.

The techniques used for NP Coreference Resolution include complex inference procedures and sophisticated knowledge sources. Several theories have been proposed too, such as the *centering* and *focusing* which try to determine an order to find the perfect match between NP. All these theories have led to the definition of many centering algorithms.

Since the 1990's the proliferation of statistical NLP made that researches changed their point of view, from a heuristic perspective, to machine learning approaches. Nowadays, one of the most common models has reduced the coreference resolution problem to a classification-clustering task, divided into two stages:

1. Classification stage: Determines if a couple of mentions are coreferential.

2. Clustering stage: Groups the mentions into entities based on the first step.

Another import event that has marked the NP coreference research is the existence of annotated corpora such as MUC-6 and MUC-7 [1]

In the following sections, the most common approaches for the supervised NP coreference are explained:

2.3.1 Annotated Corpora

As mentioned before, the existence and easy distribution of annotated coreference corpora: MUC-6 and MUC-7 corpora are considered small (60 documents each) and homogeneous, but have been extensively used for training and evaluating coreference model.

Nowadays, another common and popular corpora are the ones produced by the Automatic Content Extraction (ACE). This corpora have become more important since last decade. The earlier ACE corpora contented only English documents (newswire and broadcast articles) the latest version include Chinese and Arabic documents taken from broadcast conversation, weblog, etc.

Coreference annotations are also publicly available in treebanks. These include the English Penn Treebank (Marcus et al., 1993) which is part of the Onto Notes project, Tubingen Treebank (Telljohann et al., 2004), which is a collection of German news articles; the Prague Dependency Treebank (Hajic et al., 2006), which is extracted from the Czech National Corpus; (4) the NAIST Text Corpus (Iida et al.,2007b), which consists of 287 Japanese news articles; (5) the AnCora Corpus (Recasens and Marttí, 2009), which consists of Spanish and Catalan journalist texts; and (6) the GENIA corpus (Ohta et al.,2002), which contains 2000 MEDLINE abstracts.

The Ruslan Mitkov's research group have made available another two annotated corpora, one is a 55,000-word corpus which refers to security and terrorism and the second is training dataset released as part of the 2007 Anaphora Resolution Exercise (Orasan et al., 2008), a Coreference Resolution shared task.

Future works include SemEval-2010 has shared the task on Conference Resolution in Multiples Languages (Recasens et al., 2009) expecting to gather data from six different languages.

[1]The Message Understanding Conferences (MUC) is a project financed by DARPA (Defense Advanced Research Projects Agency) to encourage the development of new and better methods of information extraction.

2.3.2 Learning-Based Coreference Models

The following models have been developed during the last 18 years and are considered as ones of the most important.

2.3.2.1 Mention-Pair Model

The mention-pair model is a classifier that determines whether two NPs are coreferential. This model was proposed by Aone and Benett (1995) and McCarthy and Lehnert (1995), nowadays it is one of the most influential learning-based coreference model.

Although it is a very popular model, the binary approach during the classification is a not very good property. This means that it is possible to determine that A and B are coreferential, B and C are son, by A and C are not coreferential, then an additional clustering mechanism is needed to build a partition and take the classification decision.

The model needs to be trained on a data set where each instance represents two NPS and possesses a class value that indicates whether the NP are coreferential or not. As it is very difficult to get the data set with this conditions, and most NP pairs in a text are not coreferential, this turns into a much skewed class distribution.

So this model requires of a learning algorithm for training the model, the linguistic features that represent the instance, a method to create a good training data set and a clustering algorithm to build a partition.

To create proper training instances that reduce the class skewness, many heuristic methods have been developed, the most popular one is Soon et al.'s (1999-2001). For a given anaphoric noun phrase, this method creates a positive instance between NP_k and its closets preceding antecedent NP_j and an negative instance bye paring NP_k which the complete subset of NP involved.

Other kinds of methods include the usage of a filtering mechanism on the top of the instance creation method, for example avoid the creation of training instances that cannot be coreferential, this criteria include gender and number agreement as discrimination criteria.

Uryupina (2004) and Hoste (2005) presented learning-based methods based on the idea that there are relations harder to identify.

After the creation of the training set, the modeled must be trained with a specific algorithm. Most common algorithms are the Decision tree induction systems (Quinlan 1993), memory based learners (Cohen, 1995) such as machine learning. The statistical current had preferred entropy models (Berger et al., 1996), perceptrons (Freund and Schapire, 1999) and support vector machines (Joachims, 1999).

After the training, the result model can be applied to text, this must be done with the help of a clustering algorithm to generate the NP partition. The most common a simple algorithms for clustering are *closest-first clustering* (Soon, et al., 2001) *best-first clustering* (Ng and Cardie, 2002).

The closest-first clustering algorithm selects as the antecedent for an NP, NP_k, the closest preceding noun phrase that is classified as coreferential with it. If such preceding noun phrase does not exist, no antecedent is selected for NP_k. The best-first clustering algorithm aims to improve the precision of closest-first clustering, by selecting the antecedent of NP_k the most probable preceding NP that is classified as coreferential with it.

The main problem with this algorithms is that they are too greedy. The resulting clusters are formed by a small number of pairwise decision that unjustifiably favored the positive pairwise decisions. More over the NPs might end in the same cluster having enough or strong evidence that A does not corefer to B.

As an answer to this failure of the greedy algorithms other works that aim to fix their debilities. For example *Correlation clustering* (Bansal et al., 2002) whose final partition respects as possible many pairwise decisions.

Graph partitioning algorithms are applied on a weighted undirected graph where a vertex corresponds to a NP and an edge is weighted by the pairwise coreference scores between two NPs (e.g., McCallum and Wellner (2004), Nicolae and Nicolae (2006).

Some other algorithms aim to represent the mentions in a closer way as a human create coreference clusters, so instead of process the NPs in a left-to-right sequential order, the earlier ones (Cardie and Wagstaff, 1999; Klennerand Ailloud, 2008). Luo et al.'s (2004) Bell-tree-based algorithm is another clustering algorithm where the later coreference decisions are dependent on the earlier ones.

The coreference clustering algorithms will try to resolve each NP in the document, but most of them will not be anaphoric elements. In order to get better results the knowledge of a NP can improve the performance of a coreference solver. Uryupina in 2003 and Ng and Cardie in 2002 have shown that the use of an anaphoric classifier to filter non-anaphoric NPs before the Coreference Resolution can improve a learning-base solver.

An important disadvantage of this model is the two steps that form it. And no matter the increase in one of both steps, the improvement of a single step does not affect the performances of the Coreference Resolution.

The other important disadvantage is that the model can only determine how good a candidate antecedent is relative to the anaphoric NP, but not how a good a candidate antecedent is relative to other candidates.

The final disadvantage is that, sometimes the information extracted from two isolated NP might not be enough for making a correct coreference decision. This happens specially when trying pronouns.

2.3.2.2 Entity-Mention Model

To understand the motivation of the Entity-Mention model, let's think of the following example taken from McCallum and Wellner (2003).

"Sr. Clinton", "Clinton", "ella"
"Mr. Clinton", "Clinton", "she"

In the same cluster we would find *Sr. Clinton* and *Clinton*, since we are using string-matching features, as there are not gender restrictions *Clinton* is referred to *ella*. Then, by transitivity, *ella* and *Sr. Clinton* belong to the same cluster.

As this is not possible, the motivation of this model is to think of different levels of clustering or semi clustering. This model tries to classify an NP is coreferential with a preceding cluster, so each of its training instances NP comes from a NP_k, and a preceding cluster, C_j which is labeled either positive or negative. This is how each feature is represented by a level in the cluster. Features may be defined over an arbitrary subset of the NPs in any cluster. Logical predicates and the mention-pair model can help modeling a level of the cluster. A common feature represented is the number agreement, so for a given number, we are able to determine if two NPs agree in number and we would be creating the cluster-level.

Unfortunately, this approach has not given yet encouraging results. Yang et al. (2004) investigated this model, obtaining some improvements over the mention-pair model.

2.3.2.3 Ranking Models

Ranking models allow us to determine for a given a NP which of the candidate antecedents has higher probability to be resolved. This is a different approach to classification, which seems to be a more natural reformulation of the coreference solution problem. The main idea of the model is to resolve a NP to the candidate that has the higher rank. This is advantage to the classification approaches which, sometimes are unable to determine the best pairwise classification decision.

The very first ideas of ranking candidate antecedents came from the centering algorithms, which used grammatical roles to rank forward using centers (Walker et al., 1998). In this model, each training instances corresponds to a NP to be resolved.

Each NP_k has to NP to be referred to (NP_i and NP_j), and only one of it is antecedent to NP_k. This model was called tournament by Iida et al. (2003) and twin-candidate model by Yang et al(2003; 2008b)

Last advances are related to Machine learning, making possible to train a mention ranker which ranks all the candidate antecedents simultaneously. Although this capability, they are not more expressive than the mention-pair model.

In 2009, Rahman and Ng proposed the cluster-ranking mode. Which ranks preceding clusters and not candidate antecedents. This model has shown to improve mention rankers and are conceptually alike to Lappin and Leass's (1994) heuristic pronoun solver.

2.3.2.4 Knowledge Sources

It is very important to remark the importance of the different linguistic features which help the Coreference Resolution.

String-matching features It include from simple string matching operations such as exact string match, substring match, to more complicated operations as longest common sequence (Castaño, 2002) and so on.

Syntactic features Computed based on a syntactic parse tree. Ge et al. (1998) implemented a Hobbs distance feature. This measurements encodes the rank assigned to a candidate pronoun according to Hobb's algorithm (1978).

Grammatical features

Encode the grammatical properties of the NPs involved in an instance. This features, are often used as constrains for coreference. The most common example is that two NPs that corefer need to match in number and gender.

Semantic features are a very important constrain. Nowadays, the semantic knowledge has been extracted from WordNet, an unannotated corpora that measures the similarity between two nouns (Harabagiu et al., 2001)

2.3.3 Coreference Resolution in Spanish

The coreference solution in Spanish has been restricted to the resolution of third person anaphoric and zero pronouns. The third person anaphoric pronouns include *él, ella, ellos, su* and the zero pronouns are those where the subjects is omitted.

> John y Jane llegaron tarde al trabajo porque ∅ se durmieron
> *John and Jane were late for work because [they] ∅ over-slept.*

Another case of study has been the resolution of descriptions introduced by the definite article or a demonstrative that corefer with another NP.

The common techniques to figure out the problem is to apply heuristics on shallowly parsed texts and evaluate them on corpora.

2.4 Resources

2.4.1 Ancora

Ancora is a corpus in Catalan (Ancora-Cat) and Spanish (Ancora-Es) which contains 500,000 words and it is mainly formed by journalist papers. It was developed by the Centre de Llenguatge i Computació (CLiC) de la Universitat de Barcelona (UB) in Spain. It has been used in several competitions such as SemEval 2007, SemEval 2010, CoNLL 2007 and CoNLL 2009.

Some of Ancora's properties are:

1. Lemma and Part of Speech
2. Syntactic constituents and functions
3. Argument structure and thematic roles
4. Nouns related to WordNet synsets
5. Named Entities
6. Coreference relations

This corpus is available at http://clic.ub.edu/corpus/ in XML format.

2.4.2 WordNet

WordNet is a large lexical database of English developed by the Princeton University. Its development was inspired by psycholinguistic theories of human lexical memory, turning it into an interdisciplinary project. The first version was released in 1985 and the latest version for Unix/Linux systems (WordNet3.0) was released in March, 2005. Nowadays WordNet 3.1 is available only on line at the following web address: http://WordNetweb.princeton.edu/perl/webwn.

WordNet joins up nouns, verbs, adjectives and adverbs in sets called synsets. Each member of a synset is synonym of the other elements in the set, this means

```
<?xml version="1.0" encoding="UTF-8"?>
<article lng="es">
  <sentence>
    <sn arg="arg0" func="suj" tem="agt">
      <sadv adjunct="yes">
        <grup.adv>
          <r lem="casi" pos="rg" wd="Casi"/>
        </grup.adv>
      </sadv>
      <sn entity="entity1" entityref="nne" homophoricDD="yes">
        <spec gen="m" num="p">
          <d gen="m" lem="todo" num="p" pos="di0mp0" postype="indefinite" wd="todos"/>
          <d gen="m" lem="el" num="p" pos="da0mp0" postype="article" wd="los"/>
        </spec>
        <grup.nom gen="m" num="p">
          <n gen="m" lem="experto" num="p" pos="ncmp000" postype="common" sense="16:06947056" wd="expertos"/>
          <S clausetype="relative">
            <relatiu arg="arg0" coreftype="ident" entity="entity1" entityref="nne" func="suj" tem="agt">
              <p gen="c" lem="que" num="c" pos="pr0cn000" postype="relative" wd="que"/>
            </relatiu>
            <grup.verb>
              <v lem="haber" mood="indicative" num="p" person="3" pos="vaip3p0" postype="auxiliary" tense="present" wd
              <v gen="m" lem="analizar" lss="A21.transitive-agentive-patient" mood="participle" num="s" pos="vmp00sm"
            </grup.verb>
            <sadv arg="argM" func="cc" tem="mnr">
              <grup.adv>
                <r lem="sistemáticamente" pos="rg" wd="sistemáticamente"/>
              </grup.adv>
            </sadv>
            <sn arg="arg1" entityref="nne" func="cd" homophoricDD="yes" tem="pat">
              <spec gen="m" num="s">
                <d gen="m" lem="el" num="s" pos="da0ms0" postype="article" wd="el"/>
              </spec>
              <grup.nom gen="m" num="s">
                <n gen="m" lem="contenido" num="s" pos="ncms000" postype="common" sense="16:04949838" wd="contenido"/>
                <sp>
                  <prep>
                    <s lem="de" pos="sps00" postype="preposition" wd="de"/>
                  </prep>
                  <sn entity="entity6" entityref="nne" homophoricDD="yes">
                    <spec gen="m" num="p">
                      <d gen="m" lem="el" num="p" pos="da0mp0" postype="article" wd="los"/>
                    </spec>
```

Figure 2.1: Sample of Ancora corpus

synsets represent concepts. Synsets are linked by semantic and lexical relations with the other member of its synset. Every synset own a gloss which is its definition, and an offset which is the ID of the synset in the Database.

The possible relationships between synsets are the following:

1. Hypernyms: Y is hypernym of X if Y includes the meaning of X. Vehicle is hypernym of car.

2. Hyponyms: Y is hyponym of X if the meaning of X is included in the meaning of X. Car is hyponym of vehicle.

3. Meronym: Y is a meronym of X if Y is a part of X. Brick is part of wall

4. Holonym: Y is a holonym of X if X is a part of Y. Wall is formed by bricks.

Figure 2.2: WordNet 3.1 on line demo

WordNet has become a really useful tool in NLP which helps to establish the relatedness, similarity, or distance between words and concepts. Several authors have defined measurements based on the WordNet database trying to define how similar a pair or words is according to the positions of their synsets in WordNet.

Some of the concepts to understand the similarity measures are the following:

1. Root: A global root is set, to ensure that WordNet is a connected graph, therefore there will be a path between any two nodes.

2. The length of the shortest path: Given two words in synsets a and b, the similarity between them will be the length of the path that connects both synsets with the smallest number of nodes between them.

3. The depth of a node: Length of the path from a specific node to the global root.

4. Lest Common Subsummer (LSC): Is the first common parent a and b have.

One of the first ideas to measure the similarity between two words belonging to two different synsets was just to measure the length of the shortest path between them, hence:

Definition 3. *Similarity(a,b)= min(len path(a,b))*

The shortest the path is, the more related the words will be, so an inversely proportional can be deduced between the length of the path and the similarity

between a and b. The maximum similarity value is 1, which mean we are calculating Similarity(a,a), and both words are in the same synset or are exactly the same.

Another idea was proposed by Wu and Palmer in 1994.

Definition 4. $WuP(a,b) = \frac{2depth(LCS(a,b)}{depth(a)+depth(b)}$

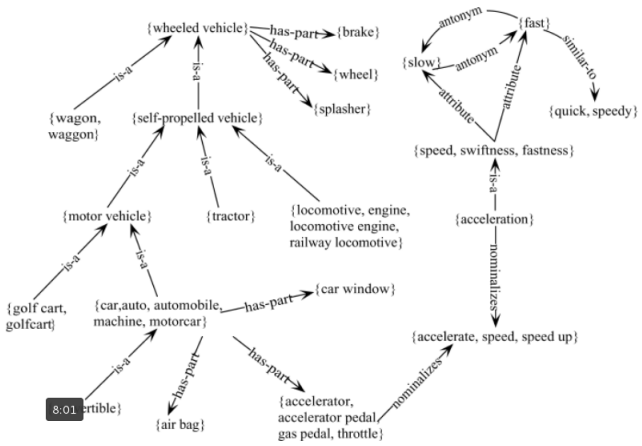

Figure 2.3: WordNet taxonomy

According to Figure 2.2, the similarity between a=*car* and b=*tractor* is:

1. $Sim_{shortestpath} = 2$

2. $Sim_{Wup} = \frac{2*1}{3+3} = 0.3333$

Nowadays WordNet is a very important resource used in NLP but its usage is exclusive of the English language that is why several similar projects for different languages have been developed. Although none of this have been as successful or complete as WordNet.

Despite this problem, the Ancora corpus has used WordNet to tag nouns, just in case that a noun exists both in English and Spanish it is tagged with the offset of its corresponding synset in WordNet 1.6. It is important to tell that the developers have been very careful in this task, and the offset presented has been disambiguated between all the possible synsets where the noun could be included.

```
Data: Words w in sentence S, senses s of each word
Result: Best sense
BestScore=0, BestSense=null
for every word w[i] in S do
    for every sense s[i] in w[i] do
        Score=0 for every other word w[k] in the phrase, k != i do
            for every sense s[l] of w[k] do
                Score = Score + number of words that occur in the gloss of
                both s[j] and s[l]
            end
        end
        if Score > BestScore then
            BestScore=Score
            BestSense=w[i]
        end
    end
    if BestScore > 0 then
        output BestSense
    end
end
```

Algorithm 1: Lesk Algorithm

2.5 Word Sense Disambiguation & the Lesk Algorithm

Word sense disambiguation is a very important task in NLP. The goal of this task is to automatically choose the meaning of a word in its context.

In 1986, Michael E. Lesk introduced the classical algorithm for Word Sense Disambiguation that was named after him.

A simplified version of this algorithm was presented in 2004 by Vasilescu et al.

These algorithms are part of the so-called Lesk approach, based on determine the maximum overlap of words between the definition of the senses and the text that surrounds the word to be disambiguated.

```
Data: Words w in sentence S, senses s of each word
Result: Best sense
BestScore=0, BestSense=null
for every word w[i] in S do
    for every sense s[j] of w[i] do
        Score=0 for every other word w[k] in the phrase, k != i do
            Score=Score +number of words that occur in the gloss of both
            sense[j]
        end
        if Score > BestScore then
            BestScore=Score
            BestSense=w[i]
        end
    end
    if BestScore > 0 then
        output BestSense
    end
end
```

Algorithm 2: Simplified Lesk Algorithm

Chapter 3

State of the art

3.1 A Machine-Learning Approach to Coreference Resolution of Noun Phrases

Soon et al. present a corpus-based, machine learning approach to solve the noun phrase coreference over a small corpus of training documents previously annotated with coreference chains of noun phrases. To get the possible markables in the training documents a pipeline of language-processing modules is used, later feature vectors are generated for possible pairs of markables. These vectors work as training examples which were given to classifier to be trained.

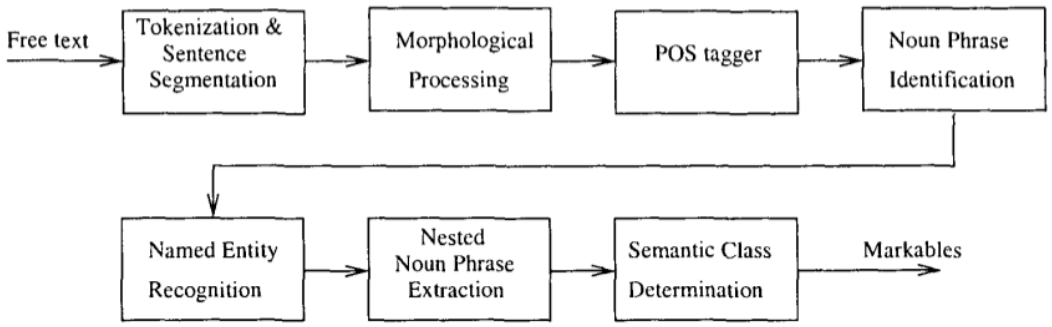

Figure 3.1: Architecture of the pipeline used in Soon's et al. work

The first step of the pipeline is to find the tokens and to segment sentences, later morphological processing is performed and part-of-speech tagging using Hidden

Markov Models. The following steps include noun phrase identification and named entity recognition also based in Hidden Markov Models, nested noun phrase extraction as well semantic class determination.

Once the markables are selected, a pair of them is selected and it is determined if they corefer or not by the extraction of a set of features that describe them. These features are generic so they can be used across different domains.

The vector defined by Soon et al. is constructed of 12 features described below, and is derived from the description of two extracted markables, i and j, where i is the potential antecedent and j is the anaphor.

Feature	Description
Distance Feature	Distance between markables i and j. Possible values are 0, 1, 2 ...
i-Pronoun Feature	Represents if i is a pronoun. Possible values are true or false.
j-Pronoun Feature	Represents if j is a pronoun. Possible values are true or false.
String Match Feature	After removing stopwords, if $i \subseteq j$ or $j \subseteq i$ possible values are true or false.
Definite Noun Phrase Feature	If the noun phrase starts with *the*, then it is a Definite Noun Phrase. Possible values are true or false.
Demonstrative Noun Phrase Feature	If the noun phrase starts with *that, this, those, these* it is a Demonstrative Noun Phrase. Possible values are true or false.
Number Agreement Feature	If i and j match in number (both singular and both plural). Possible values are true or false.
Semantic Class Agreement Feature	Expresses if i and j belong to the same class *person, organization, date, location, money, object, etc..* Possible values are true or false.
Gender Agreement Feature	Expresses if i and j agree in number. Possible values are true or false.
Both-Proper-Names feature	Expresses if i and j are both proper names. This determination is based on capitalization.

After the generation of the vectors, these are given to a machine learning algorithm to learn a classifier. The algorithm used was C5, which is a tree learning algorithm. Only the elements which are in a immediately adjacent chain are used to create positive training examples, in this pair of markables, modeled in the vector, the first

Table 3.1: Features proposed by Soon et al.

Feature	Description
Alias Feature	Expresses if i and j are acronyms or substrings. Possible values are true or false
Apposition Feature	Expresses if i and j are in an apposition structure. Possible values are true or false

noun phrase is considered as antecedent and the second one is considered as the anaphor.

Negative examples also must be supplied to the algorithm: there are other markables extracted that do not belong to any coreference chain or appear in different chains. One of this noun phrases are paired with an anaphor to get a negative example.

Soon et al.'s contribution was to define the most general and basic features between two possible markers in a coreference chain, and give them to a classifier. Small, annotated corpus are required, obtaining encouraging results, that is why this work has been the base for many other author who have trained different classifiers with this features, some of the including some others or modifying them.

3.2 Robust Pronoun Resolution with Limited Knowledge

Most of the traditional approaches to anaphora resolution do not include any deep linguistic knowledge. The reason for this is clear: this is very labor-intensive and time-consuming task, caused by the nature of the computational linguistics: It takes a long time for a human to get all the required data as well, it will consume a large amount of computational resources to process that amount of data, turning it into a really expensive computational task.

In Mitkov's work, he figures out anaphoric entities in technical manuals which have been previously pre-processed by a part-of-speech tagger. Each element in the input is checked against the agreement and for a number of antecedent indicators. The candidates will get a score by each indicator and the candidate with the highest score will returned as the antecedent.

As mentioned before, representing and manipulating the various types of linguistic

and domain knowledge is a really expensive, so different approaches have been proposed. Some of these approaches include neural networks, semantics framework, or the principles of reasoning which still are not enough.

In order to get a cheaper method than contains main linguistic information, Mitkov's looks for basic but powerful linguistic knowledge avoiding complex syntactic, semantic and discourse analysis, as well, leaving out parsing sentences and focusing in the results thrown by of a part-of-speech tagger. The next step is to identify the noun phrases which precede the anaphor within a distance of 2 sentences each of these noun phrases are checked for gender and with the anaphor and then applies the genre-specific antecedent indicators to the remaining candidates.

The value of the antecedent indicators is the core of this method because they allow to finally identify the correct antecedent for a set of possible candidates. All the candidates will have a score for each indicator, the possible values of that score include all the integers in the interval [-1, 0, 1, 2] the candidate with the highest aggregate score is proposed as the antecedent.

The antecedent indicators are the following which are gotten empirically:

1. Definiteness: Definite noun phrases score 0 and indefinite ones are penalized by -1.

2. Givenness: Noun phrases in the previous sentences that represent the "given information" score 1 and candidates not representing it score 0.

3. Indicating verbs: The first noun phrase following verbs like: discuss, present, illustrate, identify, etc... scores 1 and 0.

4. Lexical reiteration: Repeated synonymous noun phrases preceded by definite articles or demonstratives as well sequence of noun phrases with the same elements score 1, otherwise score 0.

5. Section heading preference: Noun phrase occurring in the heading of a section, part of which is the current sentence, then we consider it as the preferred candidate may score 1, 0

6. "Non-prepositional" noun phrases: Non-prepositional phrase is given a higher preference than a noun phrase which is part of a prepositional phrase, possible values are 0, -1.

7. Collocation pattern preference: This is restricted to the patterns "noun phrase (pronoun), verb" and "verb"; if the pattern matches the score is 2, otherwise 0.

8. Immediate reference: Is as a modification of the collocation preference, possible values are -2 or 0.

9. Referential distance: Noun phrases in the previous clauses are the best candidate for the antecedent of an anaphor in the following clause. Values are (2, 1, 0, -1) for anaphors in complex sentences and (1, 0, -1) for simple sentences...

10. Term preference: NPs representing terms in the field are more likely to be the antecedent than NPs which are not terms (score 1 if the NP is a term and 0 if not).

With this, Mitkov shows that his approach , using brief but powerful lingustinc information is able to solve anaphoric pronouns correctly. The gotten succes rate was 89.7%

In order to get a cheaper method than contains main linguistic information, Mitkov's looks for basic but powerful linguistic knowledge avoiding complex syntactic, semantic and discourse analysis, as well, leaving out parsing sentences and focusing in the results thrown by of a part-of-speech tagger. The next step is to identify the noun phrases which precede the anaphor within a distance of 2 sentences, each of these noun phrases are checked for gender and with the anaphor and then applies the genre-specific antecedent indicators to the remaining candidates.

3.3 End-to-End Coreference Resolution via Hypergraph Partitioning

Jai and Strube describe a novel approach to coreference resolution which implements a global decision via hypergraph partitioning. The main difference between all the previous approaches is that they perform coreference resolution globally in one step.

They implement a hypergraph-based global model in an end-to-end coreference resolution system. The system outperforms two strong baselines (Soon et al., 2001; Bengtson and Roth, 2008) using system mentions only.

The new approach to coreference resolution avoids the division into two steps and instead performs a global decision in one step. Each document is represented as a hypergraph, where the vertices denote mentions and the edges denote relational features between mentions. Then, the coreference resolution is performed globally in one step by partitioning the hypergraph into subhypergraphs so that all mentions in one subhypergraph refer to the same entity.

This approach materializes in a system called COPA which consists of modules, the learning module receives the hyperedge weights from the training data, and the resolution module create a hypergraph representation for the testing data and perform partitioning to produce subhypergraphs, each of which represents an entity.

3.3.1 Example

An example analysis of a short document involving the two entities:

```
[US President Barack Obama] came to Toronto today.
[Obama] discussed the financial crisis with [President Sarkozy].
[He] talked to [him] about the recent downturn of the European
markets.
[Barack Obama] will leave Toronto tomorrow.
```

The hypergraph in Figure 3.1 is created, based on three features. Two hyperedges denote the feature partial string match, US President Barack Obama, Barack Obama,Obama and US President Barack Obama, President Sarkozy. One hyperedge denotes the feature pronoun match, he, him. Two hyperedges denote the feature all speak, Obama, he and President Sarkozy, him

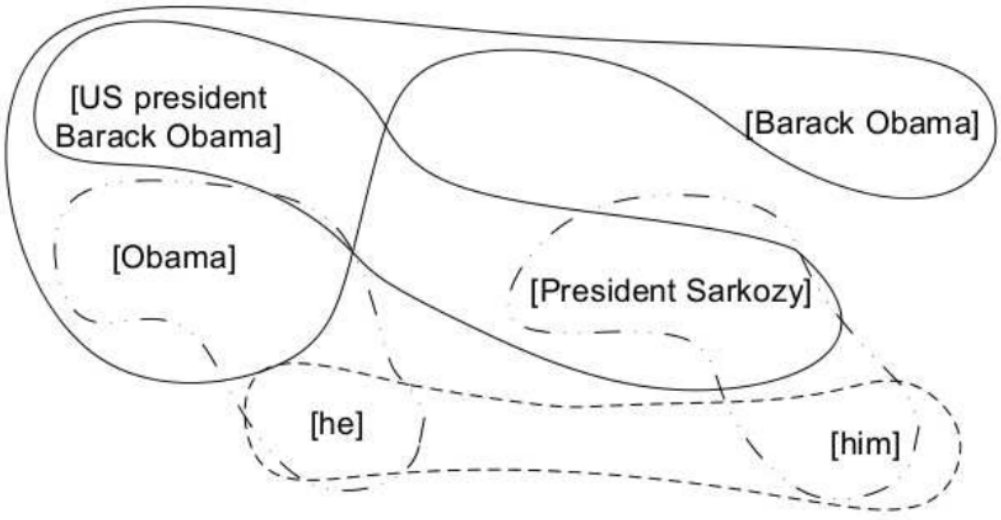

Figure 3.2: Hypergraph data representation

3.4 BART: A Modular Toolkit for Coreference Resolution

BART is a toolkit for developing coreference applications. This toolkit uses lexical and encyclopedic knowledge for entity disambiguation, a set of features similar to the ones used by Soon .

Bart was built in a maximum entropy learner with the set of features previously named. Its properties are the following: The process starts with a chunking pipeline, which mixes the Stanford POS tagger, the YamCha Chunker ante the Stanford Name Entity Recognizer, later a parsing pipeline which uses a Charniak and Johnson's re ranker parser to assign POS tags. The Berkley parse and ACE mention tagger are part of this process too. For the feature extraction, all the mentions are analyzed, looking for the possible antecedents based on Soon's set of features. Every pair of anaphor and candidate is enriched whit this features is provided to a learning-based classifier that decides if the mention pairs is corefered. The features vectors are given to Weka or to a specialized algorithm as SVMlight or a Maximum entropy classifier. Once the classifier is trained an encoder/decoder module decides where the equivalence classes where the pairs of mentions belong. Each mention in a coreference chains is tagged with the name of the chain where it belongs, mentions which own the same tag are members of the same coreference chain.

BART's open source version is available from http://www.sfs.uni-tuebingen.de/~versley/BART. This versions presents a tool for researchers who want to use which enables to add knowledge sources or use the toolkit and have access to some of the state of the art algorithms that BART implements.

3.5 Incorporating Coreference Resolution into Word Sense Disambiguation

Word Sense Disambiguation is a very important problem in NLP, it seeks to find the correct sense of a word that has multiple meanings. Some of the applications of the WSD are machine translation, knowledge acquisition, and information retrieval.

Hu and Liu say the WSD and the Coreference Resolution should be studied together more frequently, and they present an idea to incorporate the coreference resolution techniques to figure out Word Sense Disambiguation in order to improve the disambiguation precision. They present an Instance Knowledge Network approach that is helped by the coreference resolution method, which is how the dependency graphs of the candidate are connected.

In the WSD task, it is required to understand contextual knowledge which can be searched from the previous similar contents. This knowledge if formed by instances of word senses and the relationship between them. This is called Instance Knowledge Network (IKN). An IKN is a knowledge representation with three levels:

1. Word
2. Type Synset
3. Instance Level

The first two levels are gotten form WordNet, while the Instance Level is gotten by parsing a sentence in a sense tagged corpus into a dependency graph. Each word node has a dependency graph with a unique identifier, every word node will also become an instance node, and using the tagged sense of WordNet, so the dependency graph of the word turns into an Instance Graph Pattern(IGP). As result of this process, each word may have multiple senses, as well, they may be tagged in multiple positions in the corpus, and so, an instance node is created for each tagged word.

The previous steps only describe the creation of an IKN. Hu and Liu propose extending this structure with coreference resolution techniques. According to these authors, first the dependency graph is created using the Stand for dependency parser, this graph contains nodes that represent words and relations between the words as edges. An edge will not exist if there are different groups of different sentences. When two words are clarified and they belong to different sentences, their dependency graphs are connected.

Once the extended graph is gotten, the coreference resolution is performed by BART, as mentioned before, BART results are represented as tags, phrases or elements with the same tag are in the same coreference chain. Once BART has resolved the coreference chain, a base word is defined. This base word will be the pronoun itself, or in the noun phrase, the last of the multiple nouns or adjectives is chosen. The base word of the first phrase is taken, this will be the prime base word. The final step is to connect all the nodes of the base words in the group of the prime base word, each edge that is added represents an existing coreference relation. As a result, the final graph includes a bigger number or dependency graphs. This graph is called Joint Dependency Graph (JDG)

Given a JDG , all the matching sub-graphs in the instance level of the IKN with coreference information are found. What the algorithms goes after to find an each pair of matching edges between a candidate dependency graph and an Instance Graph Path, and then maximize the connection between edges in the candidate dependency graph and the IGP.

Once the matches have been performed, a probabilistic training algorithm and a WSD algorithm are developed. The first one attempts to get the conditional probability for each part of instances nodes in each IGP. Finally, the WSD algorithm attempts to find the Instances Matching Subgraph (IMSG)for a given candidate dependency graph.

With the application of the coreference resolution system, the size of the context was enlarged by growing the IKN, in an Extended Instance Knowledge Network (EIKN) which was trained with 500K instance nodes, and to compare the EIKN precision, the usage of different known WSD algorithms at the final stage.

The performance of the EIKN with different WSD algorithms has showed that enlarging the contexts helps getting higher precision than in a simple IKN.

This work shows that trying to mix the different approaches of two of the main problems in WSD may give good results when trying to solve any of them.

3.6 First-Order Probabilistic Models for Coreference Resolution

The common approach to coreference resolution using machine learning methods is the mention-pair model, where the classifier must decide if a mentions a refers to a b. Culotta et al's, argue this approach has several difficulties. The first one is that most of the times it is not very clear how to get a set of classification into disjoint clusters of noun phrases. The second reason is that the pairwise decomposition restricts the feature set to evidence about the pairs.

The existence of a set of noun phrases that were captured by a combination of pair features is what author propose as a possible solution. This means, to augment the pairwise model to enable classification over set of noun phrases.

$$p(y_j|x^j) = \frac{1}{Z_{x^j}} exp \sum_k \lambda_k f_k(x^i, y_j),$$

where:
$x^j = x_i$: Set of noun phrases,
$y_j =$: binary variable, $y_j = 1$ when all the noun phrases in $x_i \in x_j$ are coreferential,
$f_k =$: set of features,
λ_k: weight of a feature,
$\frac{1}{Z_{x^j}}$: normalizing coefficient.

To enrich the model, Markov logic networks are used, which allows to build a formulation of a first-order logic in order to characterize the noun coreference task

and can clear weights for instances of this formula. The problem of this model is its complexity, while the classic pairwise model is O ($|x^2|$), the First-Order Logic Model is O ($2^{|x|}$).

The features used were:

- Match features: Gender, number, head text
- Mention type: Pronoun, name or nominal
- Alias
- Apposition
- Relative pronoun:
- WordNet features: Hypernyms, Synonym or Antonym
- Both speak: If both contain a synonym of the verb "said"
- Substring
- Modifiers match: If a noun modifies another one
- Enumeration of each pair of noun phrases:
- Cluster size
- Number of phrases in the cluster that are of each mention type

3.7 Reconcile: A Coreference Resolution Research Platform

Reconcile is a platform for learning based noun phrase, which is characterized by the enlargement of the set of features and different approaches for the coreference resolution. The most current state of the art algorithms (2007) are included in Reconcile, and as most of the works developed, includes the preprocessing of the input documents, feature generation, classification, and clustering.

This platform is was built in Java, what ensures portability, and it is very flexible for including new sources of knowledge. For the preprocessing stage, the sentence splitting was performed with the help of OpenNLP and the University of Illinois Urbana-Champaing sentence segmentation tool, which mixes a heuristic

and maximum entropy approach. Tokens, POS and NER tags were gotten form Openly too. The identification of coreference elements is performed by an extractor developed by the authors, based on the parsing and NER identification.

For classification the pairwise model is used, and an 88 features vectors is created, this features are the properties of each coreference element.

The classifications algorithms were provided by Weka, with some extra algorithms such as SVMlight. Clustering algorithms performed were Best First Clustering, Most Recent Clustering and Single Link algorithms.

Reconcile's general architecture can easily resolve coreference chains, supporting NLP research incorporating coreference resolution into larger systems either having a reference of state of the art.

3.8 Coreference Resolution across Corpora: Languages, Coding Schemes, and Preprocessing Information

In this work, Recanses and Hovy aim to find a relationship between the variation of some parameters of a corpus and the performance of a coreference resolution system using the MUC and B^3 measures .These parameters are language, annotation scheme, and preprocessing information. Later, the corpus is given to a coreference resolution system.

For the experiments the AnCora, ACE, and OntoNotes corpora were used and the Soon's and Ng and Cardies' features were used to create vectors to train a classifier. And the systems used is called CISTELL.

When varying the language, the English corpus (OntoNotes) had a just a little better performance than the Spanish corpus (Ancora), which means that language-specific issues do not have an important impact in the system, at least there is no big difference between this mentioned languages.

The second parameter, the annotation scheme was different for the corpora. ACE and OntoNotes were used. They were reduced to a simpler scheme, the ACE scheme. This reduction allowed to focus on differences in the corpora. For example, the type of mentions. For this experiment a bigger number of documents (shared by ACE too) were used. Despite the bigger number of OntoNotes used, the performances of the classifier was slightly better.

The third parameter was to determine the importance of the source and preprocessing in the system's performance. The idea of the variation of this parameter comes with

the theory that coreference resolution requires many levels. To prove this ACE and Onto Notes, which differ in the amount and correctness of such information were used. In this experiment, ACE score was a little worse than OntoNotes.

Corpus	P	R	F
AnCora	45.73	65.16	53.75
OntoNotes	47.46	66.72	55.47

Table 3.2: MUC measures for CISTELL using Ancora and OntoNotes corpora

Corpus	P	R	F
AnCora	68.50	87.71	76.93
OntoNotes	70.36	88.05	78.22

Table 3.3: B^3 measures for CISTELL using Ancora and OntoNotes corpora

3.9 SUCRE: A Modular System for Coreference Resolution

SUCRE is a coreference resolution tool which separately performs noun, pronoun and full coreference resolution. The features used by SUCRE come from a relational database model and a regular feature definition language.

SUCRE provides a more flexible method to get features for coreference resolution. This is made by relational databases that are able to model unstructured text corpus in a structured model. A regular feature definition language was developed for SUCRE in order to extract different features in a very efficient way.

SUCRE'S architecture is the following:

1. The Text Corpus is turned into a relational database.

2. Link generation: Creation of positive and negative training samples.

3. Links features extraction : Features as POS tag, genre, semantic class... and definition of a language with keywords to select combinations of markables.

4. Learning: A Naive Bayes' classifier is used.

5. Decoding: It searches for the best predicted antecedent from a specific search space.

SUCRE participated in the SemEval-2010 Task 1 competition, and got the best results in regular closed Annotation track. The best obtained results were in the English and German corpora, where it got the highest score for all metrics.

3.10 UBIU: A Language-Independent System for Coreference Resolution

UBIU is a language independent coreference detector. It is able to find chains formed bye named entities, pronouns, and noun phrases. UBIU participated fin SemEval-2010 Task 1.

To be a language independent tool, UBIU combines language independent features and machine learning. The last one, implements Memory-Based Learning (MBL). The first step is to change the format of the data, this is required to get the language dependent feature extraction. The extraction of the features is developed by a specific module called language dependent modules. In this modules, there are finite state expressions that are able to identify the heads based on the linguistic annotations. It takes approximately one hour to the module to adapt the regular expressions to any language. Later, the syntactic heads of the possible markables are found the feature extraction is performed. These features will be used to train a classifier. The features to be obtained are the ones proposed by Rahman and Ng (2009). The only possible values that each features may takes is yes (Y) or no (N). Some of these features are:

1. m_j is the antecedent
2. m_k is the mention to be resolved
3. Y if m_j is pron.; else N
4. Y if m_j is subject; else N

After the features are extracted, a MBL is trained, and anon-exhaustive parameter Optimization across all languages. This optimization only is performed with the k closest neighbors. The value of k varies.

During UBIU's participation in SemEval-2010, this had a very good performance. It was one of the two systems that could supply a result for all the languages of the evaluation. The main goal to improve is the mention detection.

Chapter 4

Methods

The proposed method tries to mix the approaches called in previous chapter, this means, using a WSD method to find out coreference chains. The designed algorithm was implemented in Python 2.7

The main idea of the proposed algorithm is to see each nominal group as an entity, which has some specific features (genre, number,). Pairs of nominal groups in a specified search space are formed. This were gotten by comparing the nominal group A with all its neighbors. For every comparison a score is resulted. The pair will be formed by A and X, where score(A,X) is the highest. The score is gotten from the number of features they both share.

Once the pair is formed, it is verified if the pair is member of a coreference chain and turned into a feature vector for the training of an algorithm. After the training, a cluster determines the elements that belong to the same coreference chain.

In the following sections, these procedures are explained in a clearer way.

4.1 Preprocessing

As the Ancora corpus already is tagged, there is no need to identify POS, NER tags, syntactic dependencies or coreference entities. However, this information must be retrieved form the corpus.

The information required from Ancora is:

1. SN: *Sintagma Nominal* or Noun Phrase, contains the tag of the coreference entity if they are in a coreference chain.

2. GN: *Grupo Nominal* Groups of nouns and modifiers.

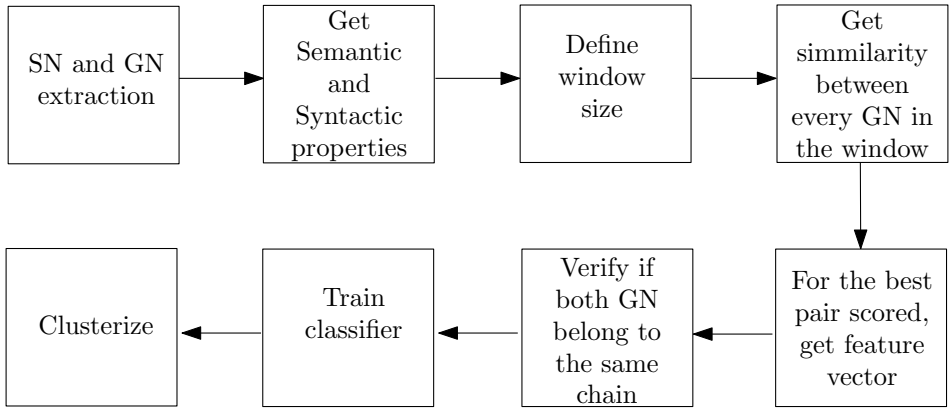

Figure 4.1: General architecture of the process

3. Nouns: *Sustantivo*Nouns that contain the POS and NER tags and WorldNet's synsets, follow the EAGLES format.

4. Pronouns: *Pronombre* Pronouns that contain POS tags and follow the EAGLES standard.

5. Sentence: *Sentence* Number of the sentence where GN or SN are located.

Each XML document of the corpus is read to get a dictionary of all the SN in the file and to know if a specific SN is part of a coreference chain. Every single GN is retrieved too with its features (number, genre) and all the nouns or pronouns that form it. The number, genre and synset of every noun that forms a specific GN are retrieved too.

4.2 Analysis

Once all the previous elements have been gotten, a window size is defined, which is the search space. For this case, a search space of three sentences was set.

All the GN that have been found in this window, they will be compare among each other, generating the pair (i,j). When i and j match in one of the following features, a score will be assigned.

1. Number: 0 when they mismatch in genre, 1 when genre is exactly the same, 0.5 when genre(i) or genre(j) is "Common"

Data: XML document
Result: SN dictionary, GN dictionary
BestScore=0, BestSense=null
for *every node sn <SN> in the document* **do**
  Get all the words in sn, the sentence where sn is in and if sn is part of a coreference chain
end
for *every node p <p> in the document* **do**
  Get p's number, genre and the where sentence p is in.
end
for *every node gn <grup.nom> in the document* **do**
  Get the sentence where gn is Get every noun in gn **for** *every noun N in gn* **do**
    Get N's number, genre, and synset
  end
end

Algorithm 3: Preprocessing algorithm

2. Genre: 0 when they mismatch in genre, 1 when genre is exactly the same, 0.5 when genre(i) or genre(j) is "Invariable"

Then, for every noun in i and in b, a comparison is performed, calculating the Wu and Palmer distance between (a,b). If the result equal zero, then the Wu and Palmer distance is calculating according the semantic class of a and b. The pair of nouns a and b that gets a bigger distance will be doubled and saved as the Semantic Similarity(SemSim), $0 \leq$ SemSim ≤ 2.

The sum of these three features is saved, and for the GN i, will be possibly in the same coreference chain that the GN j_k where Score(i, j_k) is maximum. After selecting the pair (i,j_k), both GN will be looked up at the SN dictionary, to verify if they belong to the same SN (in case of a possible apposition), if this is true the value of SameSN is 1.

When getting the semantic similarity, a first version of a vector that will be provided to Weka is created and will be enriched in a future step. This vector only contains the three previous features explained. In case of the pronouns, for all the pronouns that may be candidates as anaphor a vector is created. The vectors for pronouns are produced are one of the ways to create negative examples for the classifier.

It has already been already mentioned how the negative examples for the classifier are selected, but it has not been said the whole process to get them. Whenever a

```
Data: List of GN, List of pronouns
Result: Vectors of the best scored pairs
Window=3
for every i in GN and in window do
    for every j=i+1 in GN and in window do
        if i !=j then
            if (i⊆ j or j⊆i) then
                Vec= Get vector with i features
            end
            else
                SemSim=ProcessNoun(i,j)
                SameSN(i,j)
            end
        end
    end
    for every p in Pronouns and in window do
        Candidates=ProcessPronoun(i,p) for every c in Candidates do
            Same(i,p)
            Vec=Get vector with i,j features
        end
    end
end
```

Algorithm 4: Main algorithm

pair of GN or GN and pronoun have been detected to be possible corefered, a specific routine checks if they are in the same SN and if this SN belong to a reference chain.

4.3 Feature Vector Generation

For every pair (i,j) where i, j got the maximum score among the other GN in the window, a vector is formed. The vector includes the following features:

1. Genre: Possible values are 0 if there was no matching in genre, 1 the genre perfectly matches or 0.5 if any of i or j owns common genre.

2. Number: Possible values are 0 if there was no matching in number, 1 the number perfectly matches or 0.5 if any of i or j owns common number.

```
Data: GN i, Gn j
Result: Best noun
Max=0, Tot=0, Higher=0, Best=0
for every noun a in do
    for every noun b in j do
        Sim=WuP(i,j)x2
        if Sim=0 then
            Sim=Wup(NE(i), NE(j))2
            )
        end
        Total = ∑ (Matches between genre, number and Sim)
        Vec= Get Vector with (i,j)properties
        if Max > Total then
            Max=Total
            Possible.add( i + j)
        end
    end
end
for every element in Possible do
    if Higher > Total(element) then
        Higher=Total(element)
        Best=element
    end
end
Return Best and its vector
```

Algorithm 5: ProcessNoun

3. Sim: Real number that represents the Wu and Palmer distance as similarity between i and j. 0⩽Sim ⩽ 2

4. Distance: Number of sentences between i and j positions

5. TypeI / Type J: Possible values 0 if both are GN or 1 if a pronoun is involved

6. SN: Possible values 0 of 1 if both share the same syntactic father (belong to the same SN)

7. Class: They are part of a coreference chain or not. Possible values are yes(S) or no(N).

Data: GN i, List of pronouns Candidates
Result: SemSim
ListofVector=null
for *every p in Candidates* **do**
  if *number matches or is common* **then**
    if *genre matches or is common* **then**
      Vec= Get vector with (i,p) features
      ListofVectors.add(Vec)
    end
  end
  Return ListofVectors
end

Algorithm 6: ProcessPronoun

8. Name: The two noun phrases that form the pair.

9. File: File to be analyzed.

4.4 Classification

Once the vectors have been generated, they are given to Weka, the used version of Weka is 3.6. A Naive Bayes classifier decides whether a pair of mentions are in a coreference chain or they are not. Experiments were performed in both Weka API for java and Weka GUI (Explorer)

In order to run the classifier, a filter must be used. The cause of this is that the Naive Bayes algorithm does not support string features when classifying. The function of the filter is to reduce the data set, using just the numeric values.

```
Remove filter = new Remove ();
filter.setAttributeIndices("7"); //Removes the pair feature
filter.setInputFormat(data);
Instances data2 = Filter.useFilter(data, filter);
```

The class index is set in 6, which is the features with values “S” (if the pair is in a coreference chain) or “N” (if the pair is not in a coreference chain). As the vector includes a string attribute, a filter must be used in order to the run classifier.

Data: Gn i, Gn j, Dictionary of SN
Result: Sharing of SN and Coreference existences in SN
for *every element in Dictionary of SN* **do**
  if $\subseteq$ *element and j* $\subseteq$ *element* **then**
    if *element is in coreference chain* **then**
      Return 1,1
    end
    else
      return 1,0
    end
  end
  else
    return 0,0
  end
end

Algorithm 7: Verify SN

```
data.setClassIndex(data.numAttributes()-2);
//Feature 6 is set as the class index for training
```

Then the model is built and evaluated

```
Evaluation eTest = new Evaluation(data2);
eTest.evaluateModel(cModel, data2);
```

After the evaluation of a cluster, the allocation of classes is gotten. All the elements that belong to the class "0.0" are part vectors that represent pairs of noun phrases in coreference chain. The assignation of classes is shown in the following figure:

With all the vectors tagged as "0.0", a new arrf is generated, this only with all the vectors that have been classified with the class "0.0" which means this vectors are in any coreference chain.

4.5 Clustering

With the second arrf file, a clustering algorithm will be performed. The algorithm selected for this process is the Expectation-Maximization (EM) algorithm. EM aims to find the maximum likelihood between a collection of elements using statistical parameters.

```
%
 %
@relation coreference
@attribute gen numeric
@attribute num numeric
@attribute sim numeric
@attribute dis numeric
@attribute tipi numeric
@attribute tipj numeric
@attribute sn numeric
@attribute class {S,N}
@attribute par string
@attribute archivo string
@data
0,1,1,0,0,0,1, S , demanda$de$euros*euro,3_20020103.4.tbf.xml
1,1,1,0,0,0,1, S , Consejo_de_Camaras*presidente$del$Consejo_de_Camaras,3_200
1,1,1,0,0,0,1, S , que$nos$permite$pronosticar$que$en$los$proximos$15$o$20$di
1,0,0,0,0,0,0, N , enorme$interes*presidente$del$Consejo_de_Camaras,3_2002010
1,1,1,0,0,0,1, S , presidente$del$Consejo_de_Camaras*Jose_Manuel_Fernandez_No
1,0,0,1,0,0,0, N , Jose_Manuel_Fernandez_Norniella$$presidente$del$Consejo_de
0,1,1,0,0,1,0,N , usuarios*conjunto$de$Espana,3_20020103.4.tbf.xml
0,1,1,0,0,0,1, S , usuarios*llamadas$de$usuarios$en$las$que$sobre_todo$se$exp
$previsiones,3_20020103.4.tbf.xml
0,1,1,0,0,1,0,N , euros*conjunto$de$Espana,3_20020103.4.tbf.xml
1,0,1,0,0,0,1, S , euros*euro,3_20020103.4.tbf.xml
0,1,1,0,0,1,0,N , previsiones*conjunto$de$Espana,3_20020103.4.tbf.xml
1,1,1,0,0,0,1, S , previsiones*llamadas$de$usuarios$en$las$que$sobre_todo$se$
$previsiones,3_20020103.4.tbf.xml
1,0,0,1,0,0,0, N , euro*dias,3 20020103.4.tbf.xml
```

Figure 4.2: Arff file generated

The cluster was performed in Weka's GUI, obtaining a new arff file with the assignments of each vector in a cluster according its properties. The number of clusters was provided for every document.

Once more, the string attribute is removed because the clustering algorithms cannot work in vectors with string attributes. For Weka's GIU , in the Cluster section, there is an option for ignoring specific features.

4.6 Evaluation

For every file of the corpus, a prime file was generated, containing the real number of chains and all the GN that form it.

First, the results of the classifier and cluster performances are evaluated thanks to the statics provided by Weka. Later 3 different measures are used to see how well the selection of clusters and classification (according to the gold standard provided by the corpus) was.

For evaluation the following measures were obtained:

```
%
 %
@relation coreference
@attribute gen numeric
@attribute num numeric
@attribute sim numeric
@attribute dis numeric
@attribute tipi numeric
@attribute tipj numeric
@attribute sn numeric
@attribute class {S,N}
@attribute par string
@attribute archivo string
@data
0,1,1,0,0,0,1, S , demanda$de$euros*euro,3_20020103.4.tbf.xml,0.0
1,1,1,0,0,0,1, S , Consejo_de_Camaras*presidente$del$Consejo_de_Camaras,3_20020103.4.tbf.xml,0.0
1,1,1,0,0,0,1, S , que$nos$permite$pronosticar$que$en$los$proximos$15$o$20$dias$la$peseta$habra$desapare
$mercado*mercado,3_20020103.4.tbf.xml,0.0
1,1,1,0,0,0,1, S , presidente$del$Consejo_de_Camaras*Jose_Manuel_Fernandez_Norniella$$presidente$del$Cor
0,1,1,0,0,0,1, S , usuarios*llamadas$de$usuarios$en$las$que$sobre_todo$se$expresaron$quejas$porque$la$de
$previsiones,3_20020103.4.tbf.xml,0.0
1,0,1,0,0,0,1, S , euros*euro,3_20020103.4.tbf.xml,0.0
1,1,1,0,0,0,1, S , previsiones*llamadas$de$usuarios$en$las$que$sobre_todo$se$expresaron$quejas$porque$la
$previsiones,3_20020103.4.tbf.xml,0.0
1,1,1,0,0,0,1, S , oficinas$del$euro$instaladas$por$la$red$de$camaras$de$comercio*camaras$de$comercio,3_
0,1,1,0,0,0,1, N , comercio*red$de$camaras$de$comercio,3_20020103.4.tbf.xml,0.0
```

Figure 4.3: Classes belonging to a coreference chain

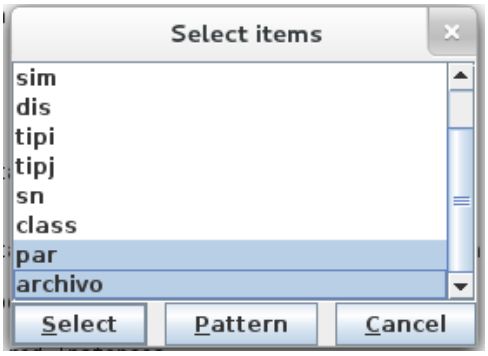

Figure 4.4: “Select feature to remove” dialog

1. Precision
2. Recall
3. F1-score

The complete procedure of evaluation includes getting this three measures for testing how good the identification of elements in a coreference chain was, and also to know how good the elements were clustered in order to match the entities tagged in the corpus that will be used as gold standard.

After each of the classification a clustering processes, the arff file obtained is examined according to the gold standard to get this scores.

It is important to say, that all the secondary files used as gold standard have the same name as the original file in the corpus, in order to use the file feature as an index an verify the coreference chain in the correspondent gold standard file.

Chapter 5

Results

Despite Ancora corpus contains more than 1000 documents, not all of them have been properly formed according the XML node structure. As mentioned before, the corpus is tagged with marks as SN or Grup.Nom, and using the XML there must be an opening and closing mark for every tag. For example:

$< sn >$
All members of the SN
$< \backslash sn >$

Or

$< grup.nom >$
All members of the GN
$< \backslash grup.nom >$

This means there must be the same number of opening $< sn >$ tags (or any other tag) than $< \backslash sn >$.But most of the documents do not follow this rule, for that reason a subset of the corpus was gotten. There were approximately 140 correct documents approximately 140 correct documents throughout the corpus, but only 41 documents were selected for the experiments.

Every document in the corpus has between 2-14 sentences and the number of vectors generated for a total number X of GN' is approximately x^2. This is mainly caused by all the GN's that are substrings of other GN's, generation repeated

iterations and comparison and because of the generation of negative examples between pronoun-GN pairs. (As mentioned before, a vector is formed between each pronoun and GN despite they corefer or not)

5.1 Classification

The classification process is performed in both, the Weka's GUI and the API. The GUI is used to obtain graphics that describe the results of this classification. Here, it is possible to save the model built and the graphs, but it is not possible to get the result of the classification as an arff file. This is the reason of the usage of the API, where the model is built, evaluated and results saved in the arff file. For both cases, the 10-fold validation testing option was used.

The performance of the EM algorithm is shown in Table 5.1. These statistics show the values values gotten during the cross validation process.

Table 5.1: Naive Bayes' performance statistics

Correctly Classified Instances	93.4066%
Incorrectly Classified Instances	6.5934%
Kappa statistic	0.8345
Mean absolute error	0.0782
Root mean squared error	0.2376
Relative absolute error	19.4833
Root relative squared error	53.1453
Total Number of Instances	87969

This result only shows that from the vectors created and marked as corefent and non coreferent, a good classification was done, the question is: Were those vectors really correctly created following the main algorithm it has been described?

To answer this question, another process for evaluation is performed, this time the classified vectors in the new intermediate arff file, this means, the three features previously named.

The measures where gotten using the following formula:

$$P = \frac{PiC}{TPiC} \tag{5.1}$$

Figure 5.1: Graphic representation of the similarity feature

$$R = \frac{PiC}{TP} \tag{5.2}$$

$$F = \frac{2P * R}{P + R} \tag{5.3}$$

where:
PiC: is the number of pairs that really belong to a coreference chain,
TPiC: is the total number of pairs that belong to a coreference chain,
TPF: is the total number of pairs found.

The obtained results for the evaluation of the pairs really in coreferent chains are showed in Table 5.2.

5.2 Clustering

For the clustering process, the intermediate arff file with formed by those vectors that the classifier has select as members of a coreference chain was used as input.

Figure 5.2: Graphic representation of the number feature

Table 5.2: Evaluation of pairs correctly selected as coreferent

Precision	0.24
Recall	0.36
F-score	0.288

By default, this algorithm decides the number of clusters to be formed by cross validation. Weka translates this statement in the option "Number of clusters" which is set in -1. If the algorithm itself attempts to find the clusters that represent the real life entities its precision is the following:

$$P = \frac{FC}{CC} \tag{5.4}$$

where:
FC is the number of found clusters,
CC is the correct number of clusters that should have been found.

The highest precision on a document was 0.6, where, from 8 defined coreference chains 3 were found. Although this is a very good result, it was complicated to get

Figure 5.3: Graphic representation of classification results between coreferent and non-coreferent pairs

the Recall value. This results tells us that the clustering process found some entities, but, are the members of each clusters in the correct cluster?

That is why another measurements where obtained. The three measurement applied in the following way:

$$P = \frac{CE}{TCR} \tag{5.5}$$

$$R = \frac{CE}{TC} \tag{5.6}$$

$$F = \frac{2 * P * R}{P + R} \tag{5.7}$$

where:
CE: Number of correct elements in its correct coreference chain,
TCR: Total number of clusters retrieved,
TC: Real number of clusters found.

The result of these scores are shown in Table 5.3.

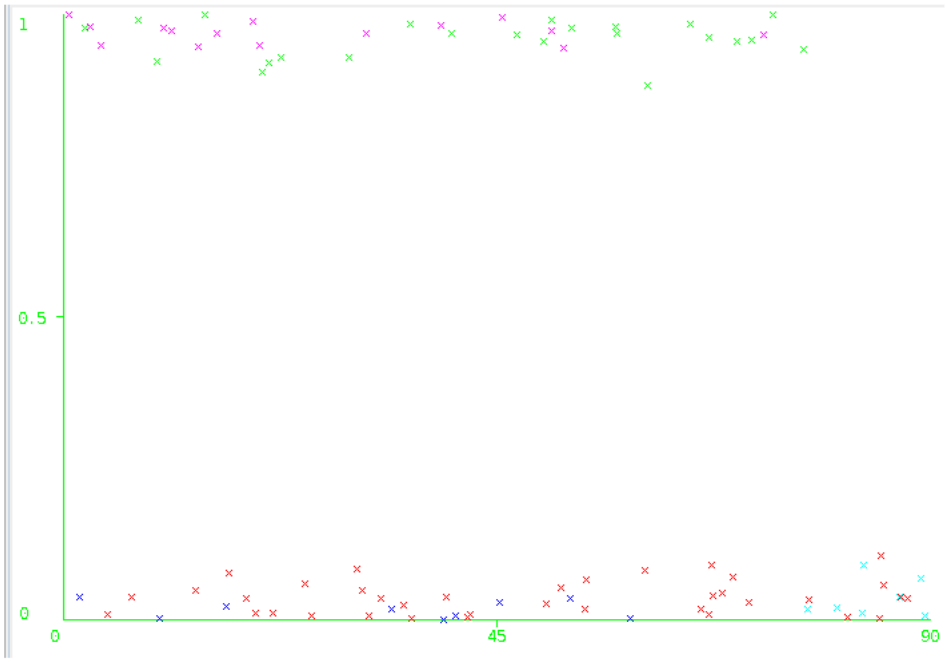

Figure 5.4: Cluster assignment

Table 5.3: Evaluation of pairs correctly selected as coreferent

Precision	0.4
Recall	0.23
F-score	0.292

5.3 Comparison with previous methods

The AnCora corpus has been used in several occasions. The most important, during SemEval-2010 Task 1, where the highest F score was gotten. The following chart includes a brief description of the results obtained. The last two were results obtained during this competition. This results where obtained during the classification stage are the following:

Author	P	R	F
Recasens, M. and Hovy, E. (2009)	45.73	65.16	53.75
Kobdani, H. and Schütze, H. (2010)	74.9	66.3	70.3
Zhekova, D. and Kübler, S. (2010)	51.1	72.7	60.0
Our method	**24.0**	**36.0**	**28.8**

Chapter 6

Discussion

As seen before, the detection of words and GN has a satisfactory result, this is caused by the good syntactical annotation the Ancora corpus has.

As well, the detection of pairs that appear in a corefence chain has been performed with very good results. This means that the features that have been proposed are powerful enough to model the relationships between a pair of mentions. This also means that the algorithm that has been proposed correctly selects the pair that matches the most as a possible candidate in coreference chain with a little amount of features. The Wu and Palmer distance seems to be a good criterion for training the classification algorithm, this is quite important because there was no need to model synonyms and antonyms relationships indeed.

A possible factor that limits this result is the annotation of the corpus, specifically, the tagging of the senses for each word, as not all the Spanish words may be found in WordNet, and a very few amount of tokens may have different senses or NE values according the context where they were used. For example word Spain used as location and organization.

On the other hand, the detection of clusters that represent the entities still have to be improved. The main goal is to separate those cluster that represent different entities but they got melted in one. This means that the vectors generated model with low efficiency the common sets, including new features may help to separate this mixed groups.

Despite this results, the comparison with other methods show that this method still have to be studied in order to get higher scores. It is important to say that the main advantage obtain is the usage of very few amount lexical resources. With this little information it was possible to build the classifier and to get promising scores. Finally, the final results are good if we think there the partial matching of noun

phrases was not considered and only complete noun phrases were retrieved.

Chapter 7

Conclusions and Future Work

7.1 Contributions

In this thesis, the following contributions have been made:

- A novel algorithm for coreference resolution based on word sense disambiguation techniques,
- Feature selection for the proposed machine learning technique,
- Training the language model for the machine learning method,
- Evaluation of the performance of the proposed algorithm.

The results show that the proposed technique is promising.

7.2 Conclusions

A new promising technique that resolves coreference chains has been developed. This new approach seems to be a possible different way to attack this problem, and as it has been said, is quite useful in nowadays applications.

This technique was able to correctly find elements of a coreference chain using the proposed algorithm. Words that belonged to a specific chain were successfully selected and isolated from words that did not belong to a coreference chain.

For those GN that were classified as members of a coreference chain, was also possible to identify the cluster where they belonged. The performance of the clustering algorithm allowed to get a good approach of the number of correct number of clusters,

and when the number of desired clusters was specified, the experiment threw better results.

Although this encouraging results, the cluster performance was not that good. The resulted clusters were formed by elements of a chain concatenate with other chains. This requires to adequate the features to enable the cluster algorithm to distinguish between different chains.

7.3 Future work

In order to improve this work, experiments with a bigger number of texts are suggested, in order to see the behavior of the proposed algorithm.

Also is suggested to try or increase the number of features, this may help to figure out the mixing of clusters, creating a wider separation between those elements that are considered as similar.

Finally, it is recommended to try the usage of different sensitivity threshold to control the formation of more refined groups, balancing with the number of identified word as part of the same group. Or, in other words try to modify the precision and recall values.

Chapter 8

Appendices

8.1 Corpus Annotations

The corpus annotation is as follows:

Sentence	<sentence>	Group of words that follow a specific order and have meaning.
Noun phrase *(Sintagma Nominal)*	<sn>	Phrase that contains nouns, pronouns, which are its nucleous.
Nominal group *(Grupo nomial)*	<grup.nom>	Group of nouns used to form a noun phrase intermediate between noun phrase and noun.

8.2 EAGLE Standard

This standard are used for tagging nouns, pronouns, adverbs, etc. In this work, only the noun and pronouns elements were used. The description of the noun tag is the following:

Position	Feature	Value	Code
1	Category	Name	N
2	Type	Common	C
		Proper	P
3	Genre	Masculine	M
		Femenine	F
		Common	C

4	Number	Singular	S
		Plural	P
		Invariable	N
5-6	Semantic class	Person	SP
		Place	G0
		Organization	O0
		Other	V0
7	Degree	Aumentative	A
		Diminutve	D

The pronoun tag description is:

Position	Feature	Value	Code
1	Category	Pronoun	P
2	Type	Personal	P
		Demonstrative	D
		Undefined	I
		Possesive	X
		Interrogative	T
		Relative	R
		Exclamative	E
3	Person	First	1
		Second	2
		Third	3
4	Genre	Masculine	M
		Femenine	F
		Neutral	N
5	Number	Singular	S
		Plural	P
		Invariable	N
6	Case	Nominative	N
		Acusative	A
		Dativ	D
		Oblicuous	O
7	Owner	Singular	S
		Plural	P
8	Politeness	Polite	P

Bibliography

[1] Alonso-Rorís, V. M., Gago, J. M. S., Rodríguez, R. P., Costa, C. R., Carballa, M. A. G., and Rifón, L. A. Information extraction in semantic, highly-structured, and semi-structured web sources. *Polibits 49* (2014), 69–75.

[2] Aone, C., and Bennett, S. W. Evaluating automated and manual acquisition of anaphora resolution strategies. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics* (1995), pp. 122–129.

[3] Aone, C., and Bennett, S. W. Applying machine learning to anaphora resolution. In *Connectionist, Statistical, and Symbolic Approaches to Learning for Natural Language Processing*, S. Wermter, E. Riloff, and G. Scheler, Eds., vol. 1040 of *Lecture Notes In Computer Science*. Springer-Verlag, London, UK, 1996, pp. 302–314.

[4] Bagga, A., and Baldwin, B. Algorithms for scoring coreference chains. In *Proceedings of the Linguistic Coreference Workshop at the First International Conference on Language Resources and Evaluation* (Granada, Spain, May 1998), pp. 563–566.

[5] Baldwin, B. Cogniac: High precision coreference with limited knowledge and linguistic resources. In *Proceedings of the ACL'97/EACL'97 Workshop on Operational Factors in Prctical, Robust Anaphora Resolution* (Madrid, Spain, 1997), pp. 38–45.

[6] Banerjee, S. Adapting the Lesk algorithm for word sense disambiguation to WordNet. Master's thesis, University of Minnesota, 2002.

[7] Bangalore, S., and Baldwin, B. Exploiting supertag representation for fast coreference resolution. In *Proceedings of the International Conference*

on Natural Language Processing and Industrial Applications (NLP+ IA'96) (Moncton, Canada, 1996).

[8] Bansal, N., Blum, A., and Chawla, S. Correlation clustering. In *Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science* (2002), pp. 238–247.

[9] Barbu, C., and Mitkov, R. Evaluation tool for rule-based anaphora resolution methods. In *Proceedings of the Association for Computational Linguistics* (Toulouse, France, 2001).

[10] Baumann, S., and Riester, A. Coreference, lexical givenness and prosody in German. *Lingua 136* (2013), 16–37.

[11] Bean, D., and Riloff, E. Unsupervised learning of contextual role knowledge for coreference resolution. In *Proceedings of 2004 North American Chapter of the Association for Computational Linguistics* (2004).

[12] Berger, A. L., Pietra, S. A. D., and Pietra, V. J. D. A maximum entropy approach to natural language processing. *Computational Linguistics 22*, 1 (1996), 39–71.

[13] Bergler, S., Witte, R., Khalife, M., Li, Z., and Rudzicz, F. Using knowledge-poor coreference resolution for text summarization. In *Proceedings of the HLT-NAACL 2003 Text Summarization Workshop and Document Understanding Conference (DUC 2003)* (Edmonton, Alberta, 2003), pp. 85–92.

[14] Bergsma, S. Automatic acquisition of gender information for anaphora resolution. In *Proceedings of the Eighteenth Canadian Conference on Artifical Intelligence* (2005), pp. 342–353.

[15] Björkelund, A., Eckart, K., Riester, A., Schauffler, N., and Schweitzer, K. The extended dirndl corpus as a resource for coreference and bridging resolution. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)* (Reykjavik, Iceland, 2014).

[16] Budanitsky, A., and Hirst, G. Evaluationg wordnet-based measures of lexical semantic relatedness. *Computational Liguistics* (2005), 13–47.

[17] Cai, J., and Strube, M. End-to-end coreference resolution via hypergraph partitioning. In *23rd International Conference on Computational Linguistics* (2010), pp. 143–151.

[18] Carbonell, J., and Brown, R. Anaphora resolution: A multi-strategy approach. In *12th International Conference on Computational Linguistics* (1988), pp. 96–101.

[19] Cardie, C., and Wagstaff, K. Noun phrase coreference as clustering. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Very Large Corpora* (1999), ACL, pp. 92–89.

[20] Castaño, J., Zhang, J., and Pustejovsky, J. Anaphora resolution in biomedical literature. In *Proceedings of the 2002 International Symposium on Reference Resolution* (2002).

[21] Cohen, W. Fast effective rule induction. In *Proceedings of the 12th International Conference on Machine Learning* (1995), pp. 115–123.

[22] Dagan, I., and Itai, A. Automatic processing of large corpora for the resolution of anaphora references. In *Proceedings of the 13th International Conference on Computational Linguistics (COLING '90)* (1990), pp. 330–332.

[23] Das, N., Ghosh, S., Gonçalves, T., and Quaresma, P. Comparison of different graph distance metrics for semantic text based classification. *Polibits 49* (2014), 51–57.

[24] Evans, R. Refined salience weighting and error analysis in anaphora resolution. In *Proceedings of the 2002 International Symposium on Reference Resolution for Natural Processing* (Alicante, Spain, Jun 2002), University of Alicante.

[25] Evans, R., and Orasan, C. Improving anaphora resolution by identifying animate entities in texts. In *Proceedings of the Discourse Anahpora and Reference Resolution Conference* (2000), pp. 154–162.

[26] Finley, T., and Joachims, T. Supervised clustering with support vector machines. In *Proceedings of the 22nd International Conference on Machine Learning* (2005), pp. 217–224.

[27] Freund, Y., and Schapire, R. E. Large margin classification using the perceptron algorithm. *Machine Learning 37*, 3 (1999), 277–296.

[28] Gaizauskas, R., and Wilks, Y. Information extraction: Beyond document retrieval. *Journal of Documentation 54*, 1 (1998), 70–105.

[29] Ge, N., Hale, J., and Charniak, E. A statistical approach to anaphora resolution. In *Proceedings of the Sixth Workshop on Very Large Corpora* (Montreal, Canada, 1998), pp. 161–170.

[30] Hajic, J., Panevova, J., Hajicova, E., Panevova, J., Sgall, P., Pajas, P., Stepanek, J., Havelka, J., and Mikulova, M. *The Prague Dependency Treebank 2.0.* Linguistic Data Consortium, 2006.

[31] Hajičová, E., Panevová, J., and Sgall, P. Coreference in Annotating a Large Corpus. In *Proceedings of the Second LREC Conference* (Athens, Greece, 2000), pp. 497–500.

[32] Hall, K. B. A statistical model of nominal anaphora. Master's thesis, Brown University, 2001.

[33] Halliday, M. A. K., and Hasan, R. *Cohesion in English.* Longman, London, 1976.

[34] Harabagiu, S. M., Bunescu, R. C., and Maiorano, S. J. Text and knowledge mining for coreference resolution. In *Proceedings of the 2nd Meeting of the North American Chapter of the Association of Computational Linguistics* (Carnegie Mellon University, Pittsburg, PA, USA, June 2001), ACL, pp. 55–62.

[35] Hirschman, L., and Chinchor, N. Muc-7 coreference task definition, July 1997.

[36] Hoste, V. *Optimization Issues in Machine Learning of Coreference Resolution.* PhD thesis, University of Antewerp, Belgium, 2005.

[37] Hu, S., Liu, C., Zhao, X., and Kowalkiewicz, M. Building instance knowlegde network for word sense disambiguation. In *Thirty-Fourth Australasian Computer Science Conference* (2011), pp. 3–10.

[38] Iida, R., Inui, K., , and Matsumoto, Y. Capturing salience with a trainable cache model for zero-anaphora resolution. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP* (2009), pp. 647–655.

[39] Jiang, J., and Conrath, D. Semantic similarity based on corpus statistics and lexical taxonomy. In *International Conference on Research of Computational Linguistics* (1997), pp. 1–15.

[40] Kameyama, M. Recognizing referential links: An information extraction perspective. In *Proceedings of a Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts* (1997), Association for Computational Linguistics.

[41] Kehler, A. Probabilistic coreference in information extraction. In *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing* (Providence, Rhode Island, USA, August 1997), Brown University, ACL, pp. 163–173.

[42] Kennedy, C., and Boguraev, B. Anaphora for everyone: Pronominal anaphora resolution without a parser. In *Proceedings of the 16th International Conference on Computational Linguistics* (Copenhagen, Denmark, 1996), pp. 113–118.

[43] Kibble, R., and van Deemter, K. Coreference annotation: Whither? In *Proceedings of the 2nd International Conference on Language Resources and Evaluation* (Athens, Greece, 2000), pp. 1281–1286.

[44] Kobdani, H., and Schütze, H. Sucre: A modular system for coreference resolution. In *Proceedings of the 5th International Workshop on Semantic Evaluation* (2010), Association for Computational Linguistics, pp. 92–95.

[45] Lappin, S., and Leass, H. J. An algorithm for pronominal anaphora resolution. *Computational Linguistics 20*, 4 (1994), 535–561.

[46] Lee, H., Chang, A., Peirsman, Y., Chambers, N., Surdeanu, M., and Jurafsky, D. Deterministic coreference resolution based on entity-centric, precision-ranked rules. *Computational Linguistics*, Just Accepted (2013), 1–54.

[47] Lee, H., Peirsman, Y., Chang, A., Chambers, N., Surdeanu, M., and Jurafsky, D. Stanford's multi-pass sieve coreference resolution system at the conll-2011 shared task. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task* (2011), Association for Computational Linguistics, pp. 28–34.

[48] Liang, T., and Wu, D.-S. Automatic pronominal anaphora resolution in english texts. *Computation Linguistics and Chinese Language Processing 9*, 1 (February 2004), 21–40.

[49] Luo, X. On coreference resolution performance metrics. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing* (Vancouver, British Columbia, Canada, October 2005), Association for Computational Linguistics, pp. 25–32.

[50] Luo, X., Ittycheriah, A., Jing, H., Kambhatla, N., and Roukos, S. A mentionsynchronous coreference resolution algorithm based on the Bell tree. In *Proceedings of the 42nd AnnualMeeting of the Association for Computational Linguistics* (2004), pp. 135–142.

[51] Luo, X., and Zitouni, I. Multi-lingual coreference resolution with syntactic features. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing* (Vancouver, British Columbia, Canada, October 2005), Association for Computational Linguistics, pp. 660–667.

[52] Marcus, M. P., Santorini, B., and Marcinkiewicz, M. A. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics 19*, 2 (1993), 313–330.

[53] Markert, K., and Nissim, M. Comparing knowledge sources for nominal anaphora resolution. *Computational Linguistics 31*, 3 (September 2005), 367–401.

[54] McCallum, A., and Wellner, B. Conditional models of identity uncertainty with application to noun coreference. In *Advances in Neural Information Proceesing Systems* (2004).

[55] McCarthy, J. F., and Lehnert, W. G. Using decision trees for coreference resolution. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence* (1995), C. Mellish, Ed., pp. 1050–1055.

[56] Mitkov, R. Towards a more consistent and comprehensive evaluation of anaphora resolution algorithms and systems. In *Proceedings of the Discourse, Anaphora and Reference Resolution Conference* (Lancaster, UK, 2000), pp. 96–107.

[57] Mitkov, R. Outstanding issues in anaphora resolution. In *Proceedings of the Second International Conference on Computational Linguistics and Intelligent* (Mexico City, Mexico, February 2001), A. F. Gelbukh, Ed.

[58] Mitkov, R. *Anaphora Resolution.* Pearson Education, Edinburgh, Great Britain, 2002.

[59] Mitkov, R., Boguraev, B., and Lappin, S. Introduction to the special issue on computational anaphora resolution. *Computational Linguistics 27*, 4 (2001), 473–477.

[60] Modjeska, N. N. *Resolving Other-Anaphora.* PhD thesis, University of Edinburgh, 2003.

[61] Morton, T. S. Coreference for nlp applications. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics* (Hong Kong, 2000), Association for Computational Linguistics, pp. 173–180.

[62] Ng, H. T., Zhou, Y., Dale, R., and Gardiner, M. A machine learning approach to identification and resolution of one-anaphora. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI 2005)* (Edinburgh, Scotland, UK, 2005), pp. 1105–1110.

[63] Ng, V., and Cardie, C. Combining sample selection and error-driven pruning for machine learning of coreference rules. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing* (2002), pp. 55–62.

[64] Ng, V., and Cardie, C. Identifying anaphoric and non-anaphoric noun phrases to improve coreference resolution. In *Proceedings of the 19th International Conference on Computational Linguistics* (Howard International House, Taipei, Taiwan, August 2002), ACL.

[65] Ng, V., and Cardie, C. Improving machine learning approaches to coreference resolution. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics* (University of Pennsylvania, Philadelphia, PA, USA, August 2002), ACL, pp. 104–111.

[66] Ng, V., and Cardie, C. Bootstrapping coreference classifiers wtih multiple machine learning algorithms. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing* (Sapporo, Japan, July 2003), ACL, pp. 113–120.

[67] Nicolae, C., and Nicolae, G. Best-Cut: A graph algorithm for coreference resolution. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing* (2006), pp. 275–283.

[68] Ohta, T., Tateisi, Y., and Kim, J.-D. The GENIA corpus: An annotated research abstract corpus in molecular biology domain. In *Proceedings of the Second International Conference on Human Language Technology Research* (2002), pp. 82–86.

[69] Olsson, F. A survey of machine learning for reference resolution in textual discourse. Tech. rep., Swedish Institute of Computer Science, 2004.

[70] Orasan, C., Cristea, D., Mitkov, R., and Branco, A. H. Anaphora resolution exercise: an overview. In *LREC* (2008).

[71] Pakray, P., Poria, S., Bandyopadhyay, S., and Gelbukh, A. Semantic textual entailment recognition using UNL. *Polibits 43* (2011), 23–27.

[72] Peris, A., M., T., and H., R. Semantic annotation of deverbal nominalizations in the Spanish AnCora corpus. In *Treebanks and Linguistic Theories* (2010), pp. 187–198.

[73] Poria, S., Agarwal, B., Gelbukh, A., Hussain, A., and Howard, N. Dependency-based semantic parsing for concept-level text analysis. In *Proceedings of the 15th International Conference on Intelligent Text Processing and Computational Linguistics, CICLing 2014, Part I* (2014), vol. 8403 of *Lecture Notes in Computer Science*, Springer, pp. 113–127.

[74] Poria, S., Gelbukh, A., Cambria, E., Hussain, A., and Huang, G.-B. Emosenticspace: A novel framework for affective common-sense reasoning. *Knowledge-Based Systems 69* (October 2014), 108–123.

[75] Preiss, J. Anaphora resolution with memory based learning. In *Proceedings of CLUK5* (University of Leeds, UK, January 2002), Computational Linguistics UK, pp. 1–9.

[76] Preiss, J. A comparison of probabilistic and non-probabilistic anaphora resolution algorithms. In *Proceedings of the student workshop at ACL* (Philadelphia, Pennsylvania, USA, 2002), pp. 42–47.

[77] Quinlan, J. R. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA, 1993.

[78] Raghunathan, K., Lee, H., Rangarajan, S., Chambers, N., Surdeanu, M., Jurafsky, D., and Manning, C. A multi-pass sieve for coreference resolution. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing* (2010), Association for Computational Linguistics, pp. 492–501.

[79] Rahman, A., and Ng, V. Supervised models for coreference resolution. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing* (2009), pp. 968–977.

[80] Rahman, A., and Ng, V. Supervised models for coreference resolution. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2* (2009), Association for Computational Linguistics, pp. 968–977.

[81] Recasens, M., and Hovy, E. A deeper look into features for coreference resolution. In *Anaphora Processing and Applications*. Springer, 2009, pp. 29–42.

[82] Recasens, M., and Hovy, E. Coreference resolution across corpora: Languages, coding schemes, and preprocessing information. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics* (2010), Association for Computational Linguistics, pp. 1423–1432.

[83] Recasens, M., Màrquez, L., Sapena, E., Martí, M. A., and Taule, M. Semeval-2010 task 1: Coreference resolution in multiple languages. In *Proceedings of the 5th International Workshop on Semantic Evaluation* (2010).

[84] Recasens, M., Màrquez, L., Sapena, E., Martí, M. A., Taulé, M., Hoste, V., Poesio, M., and Versley, Y. Semeval-2010 task 1: Coreference resolution in multiple languages. In *Proceedings of the 5th International Workshop on Semantic Evaluation* (2010), Association for Computational Linguistics, pp. 1–8.

[85] Rich, E., and LuperFoy, S. An architecture for anaphora resolution. In *ACL Conference on Applied Natural Language Processing* (1988), pp. 18–24.

[86] Schnitzer, S., Schmidt, S., Rensing, C., and Harriehausen-Mühlbauer, B. Combining active and ensemble learning for efficient classification of web documents. *Polibits 49* (2014), 39–45.

[87] Sidorov, G. Syntactic dependency based n-grams in rule based automatic english as second language grammar correction. *International Journal of Computational Linguistics and Applications 4*, 2 (2013), 169–188.

[88] Sidorov, G. Should syntactic n-grams contain names of syntactic relations? *International Journal of Computational Linguistics and Applications 5*, 1 (2014), 139–158.

[89] Sidorov, G., and Gelbukh, A. La resolución de correferencia con un diccionario de escenarios. In *Proc. CIC-99, Simposium Internacional de Computación* (Mexico D.F., November 15–19 1999), CIC, IPN, pp. 382–391.

[90] Sidorov, G., Gelbukh, A., Gómez-Adorno, H., and Pinto, D. Soft similarity and soft cosine measure: Similarity of features in vector space model. *Computación y Sistemas 18*, 3 (2014), 491–504.

[91] Soon, W., Ng, H. T., and Yong Lim, D. A machine learning approach to coreference resolution of noun phrases. *Computational Liguistics* (2001), 521–544.

[92] Soon, W. M., Ng, H. T., and Lim, D. C. Y. A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics 27*, 4 (2001), 521–544.

[93] Stoyanov, V., Cardie, C., Gilbert, N., Riloff, E., Buttler, D., and Hysom, D. Reconcile: A coreference resolution research platform. In *Proceedings of the Annual Conference of the Association for Computational Linguistics (ACL 2010)* (2010), Association of Computational Liguistics, pp. 1–14.

[94] Taulé, M., Martí, M., and Recasens, M. AnCora: Multilevel annotated corpora for catalan and spanish. In *6th International Conference on Language Resources and Evaluation* (2008), pp. 521–544.

[95] Taulé, M., Martí, M., and Recasens, M. AnCora-CO: Coreferentially annotated corpora for Spanish and Catalan. *Language Resources and Evaluation 44*, 4 (2010), 315–345.

[96] Telljohann, H., Hinrichs, E. W., Kübler, S., Zinsmeister, H., and Beck, K. *Stylebook for the Tübingen Treebank of Written German (TüBa-D/Z)*. University of Tübingen, November 2009.

[97] Uryupina, O. Linguistically motivated sample selection for coreference resolution. In *Proceedings of the 5th Discourse Anaphora and Anaphor Resolution Colloquium* (2004).

[98] van Deemter, K., and Kibble, R. On coreferring: Coreference in muc and related annotation schemes. *Computational Linguistics 27*, 4 (December 2000), 629–637.

[99] Versley, Y., Ponzetto, S., and Poesio, M. BART: A modular toolkit for coreference resolution. In *HLT-Demonstrations'08, Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Demo Session* (2008), Association of Computational Liguistics, pp. 9–12.

[100] Vilain, M., Burger, J., Aberdeen, J., Connolly, D., and Hirschman, L. A model-theoretic coreference scoring scheme. In *Proceedings of the 6th Message Understanding Conference* (San Mateo, CA, USA, 1995), Morgan Kaufmann, pp. 45–52.

[101] Walker, M., Joshi, A., and Prince, E., Eds. *Centering Theory in Discourse*. Oxford University Press, 1998.

[102] Yang, X., Su, J., and Tan, C. L. Improving noun phrase coreference resolution by matching strings. In *Proceedings of the First International Joint Conference on Natural Language Processing* (2004), pp. 22–31.

[103] Yang, X., Su, J., and Tan, C. L. A twin-candidate model for learning-based anaphora resolution. *Computational Linguistics 34*, 3 (2008), 327–356.

[104] Yang, X., Su, J., Zhou, G., and Tan, C. L. Coreference resolution using competition learning approach. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics* (July 2003), ACL.

[105] Yang, X., Zhou, G., Su, J., and Tan, C. L. Coreference resolution using competitive learning approach. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics* (2003), pp. 176–183.

[106] Zhekova, D., and Kübler, S. Ubiu: A language-independent system for coreference resolution. In *Proceedings of the 5th International Workshop on Semantic Evaluation* (2010), Association for Computational Linguistics, pp. 96–99.

[107] Zhila, A., and Gelbukh, A. Automatic identification of facts in real Internet texts in Spanish using lightweight syntactic constraints: Problems, their causes, and ways for improvement. *Revista Signos. Estudios de Lingüística* (2015).