Extraction of Family Relations between Entities

Daniel Santos, Nuno Mamede, Jorge Baptista

IST – Instituto Superior Técnico
Universidade do Algarve
L^2F – Spoken Language Systems Laboratory – INESC ID Lisboa
Rua Alves Redol 9, 1000-029 Lisboa, Portugal
{daniel.santos,nuno.mamede}@ist.utl.pt,jbaptis@ualg.pt

Abstract. Nowadays, there is a growing need to automatically extract information from texts. In this active research field much effort has been invested to improve the identification and classification of named entities, the detection of time expressions, and the identification of semantic relations between text entities. This paper presents a system that identifies and classifies family relations. The directives, the options used, the implementation and the results obtained are here presented.

Resumo. Existe uma necessidade crescente em extrair informação a partir de um texto, nomeadamente na identificação e classificação de entidades mencionadas e na identificação de relações semânticas entre essas entidades. Este artigo apresenta um sistema que identifica relações familiares. Descrevem-se as directivas de identificação e anotação, as opções adoptadas, a implementação do sistema e os resultados obtidos.

1 Introduction

Automatic extraction of semantic knowledge is one of the goals of Natural Language Processing. The extraction of semantic relations between entities represented in a text can improve the performance of systems that rely on this type of information, such as question/answering systems and text summarization systems. Family relations are a particularly well defined set of semantic relations. Historical and biographical documents are examples of texts that are every rich in Family Relations.

Although the evaluation of these type of systems for the English language has produced very good results, for the Portuguese language the extraction of semantic relations is still in an early phase and results are not as good yet. In the late 80's, the first major evaluation campaigns for relation extraction in the English language took place in MUC[1] (Message Understanding Conference). In the late 90's, the ACE (Automatic Content Extraction) conference promoted a joint evaluation almost every year.

For the Portuguese language, the only joint evaluation contest ever held for the extraction of semantic relations took place in 2008, as a specific track of the

[1] http://www.itl.nist.gov/iad/894.02/related_projects/muc/proceedings/ie_task.html

INForum 2010 - II Simpósio de Informática, Luís S. Barbosa, Miguel P. Correia (eds), 9-10 Setembro, 2010, pp. 549–560

HAREM conference (Avaliação e Reconhecimento de Entidades Mencionadas) [Mota & Santos, 2008]. Results reported in this task were not as good as those achieved in the Named Entities Recognition task.

This paper describes the implementation of a system built to identify *family* relations (like parenting, sibling, etc.), for the Portuguese language and reports on an evaluation of that system. The main goal is to get a good f-measure minimizing the number of incorrect classifications, ie, maximizing precision.

The remainder of this paper is organized as follows. Section 2 presents the state of the art and the methodologies used. Section 3 describes the architecture and the implementation of this system. Section 4 presents the strategy used to extract the semantic relations, while Section 5 contains the evaluation. Finally, in Section 6 some conclusions are drawn in order to structure future work.

2 Related Work

The *Family* relation type has been present in every ACE edition, although it underwent some changes over time. In the earlier editions[2], this category was associated to several subcategories, but after the fourth edition[3] there has been only one major category *Family*, which includes all different types of family relations. In HAREM, the Portuguese joint evaluation contest, the *Family* category was also present as one of the subcategories of "Outras" [Freitas et. al., 2009].

Both evaluations consider all family relations in a single category without specifying the type of the relation. We decided that family relations should be differentiated from other types of semantic relations. In order to do that, we created a set of features to identify this particular type of semantic relation.

Two major groups of methods are usually adopted for the semantic relations extraction task: rule-based and machine learning approaches. All systems evaluated in HAREM are based on rules, i.e, these systems analyze the syntactic structure of sentences looking for patterns and then, based on the information that is present on every sentence and the patterns extracted, they deduce relations among entities. For example, in the sentence *O João é primo do José* "João is José's cousin" if the system matches the pattern consisting in a noun phrase (NP) whose head is a human noun (e.g. *João*), the verb *ser* "to be", a family relation noun (e.g. *primo* "cousin") and a prepositional phrase (PP) introduced by *de* "of" with another human head noun (e.g. *José*); and if a subject relation has been established between the first NP and the verb, while a modifier relation has been found to exist between the head of the last NP and the family relation noun, then a *Family* relation between the two human nouns should be established. The best system in the global task of relation extraction in HAREM was REMBRANDT [Cardoso, 2008] which is based on a set of rules used to infer

[2] More information about ACE02 in *"ACE Evaluation plan version 06"* is available in `http://www.itl.nist.gov/iad/mig/tests/ace/2002/doc/`

[3] *"Version (7) of the 2004 ACE evaluation plan"* in `http://www.itl.nist.gov/iad/mig/tests/ace/2004/`

new relations. The result achieved was 45% f-measure, with 60% precision and 35% recall.

Culotta and Sorensen [Culotta & Sorensen, 2004] use the same categories defined in the first two ACE editions, and they try different approaches for relation extraction. They created a system based on machine learning with dependency tree kernels. The first step consists in building a parse tree using a maximum entropy statistical parser. This tree is then converted to a dependency tree that represents the grammatical relations between words in a sentence. The final step consists on the application of kernel methods, defined by the authors, for the extraction of relations among entities. They report a best result of 63.2% f-measure, with 81.2% precision and 51.8% recall.

3 Architecture

In this paper, a rule-base approach is adopted, since there is still no available corpus for the Portuguese language annotated for family semantic relations. The corpus used in HAREM cannot be used because: the annotations did not include different types of family relations, the number of annotations in the data set is too small for an efficient use of machine learning methods, and finally, the annotations in this corpus are made between named entities and in this paper we extract relations between any type of entities.

Furthermore, because semantic relations can be viewed as another layer of information over syntactic dependencies, already being extracted by the syntactic parser here used, it would be easier to extend the rule-based grammar already implemented in this system [Mamede, 2007] to encompass also semantic relations.

The identification of relations between entities is thus performed in XIP (Xerox Incremental Parsing), one of the modules of the L2F[4] NLP processing chain, whose structure is sketched in Figure 1, and that will be briefly presented below.

The first module (Segment Splitter) splits the text into tokens, while Palavroso [Medeiros, 1995] assigns to each token all the possible part-of-speech (POS) tags, depending on the token ambiguity.

The Sentence Splitter splits the text into sentences. Every time the system finds one of the following characters ".", "!" and "?" it considers it as the end of the current sentence. The result is converted into an adequate format and piped into RuDriCo [Pardal, 2007], which uses several heuristics to remove or select some of the POS that were given by the morphosyntactic labeling module. This type of rules is based on previously known cases and they choose or eliminate some specific POS for a given token given its neighboring context. Another functionality of RuDriCo is the joining of strings of words forming compounds as single tokens and splitting of contracted word forms into their component words; for instance, the words *Coreia*; *do*; *Norte* becomes a single token *Coreia do Norte*

[4] Laboratório de Sistemas de Língua Falada do Instituto de Engenharia de Sistemas e Computadores - Investigação e Desenvolvimento

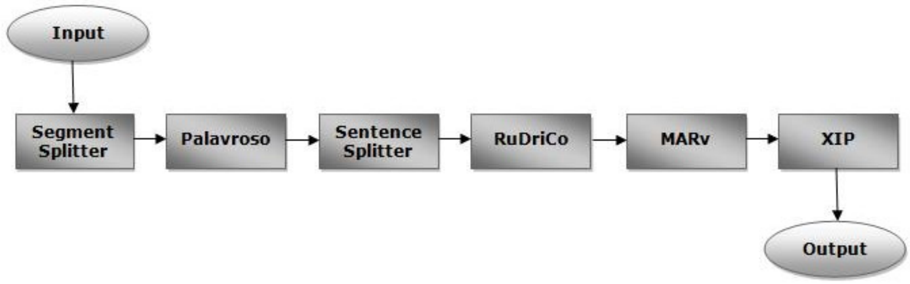

Fig. 1. L2F NLP Processing Chain.

("North Corea"), while the contraction *disso* is split in to preposition *de* ("of") and pronoun *isso* ("that"). The MARv module [Ribeiro et. al., 2003] performs the remaining disambiguation. It uses a statistical model and the Viterbi algorithm to choose the most probable category for each given token in the sentence. The result is again converted and piped into XIP [Xerox, 2003], where a set of complex operations is carried out, namely the structuring of the sentences into chunks, the extraction of syntactic dependencies, named entities recognition, and anaphora resolution. It is also at this stage that the extraction of semantic relations takes place.

4 Methodology

Only one major category, named *FAMILY*, has been defined, covering all family relations. Each relation will be associated to a feature that expresses the relation type (uncle, parenting, sibling, etc.) and the gender of the relation arguments. For example, in the sentence *O João é primo do José* "João is José's cousin", already presented above, the following semantic relation is extracted:

```
FAMILY_cousin_1M_2M(João,José)
```

To look for relevant patterns, the words (mainly nouns) that express family relations are very important lexical clues. For example, the word *pai* ("father") is a good lead, but its mere presence does not mean that we have a family relation in that sentence, for example in sentences like *São Pacómio, pai da vida monástica cenobítica*[5] "St. Pachomius, the father of cenobitic monastic life", where *pai* is used in the sense of "founding father". In order to solve this and similar problems, rules have to be rendered much more precise to ensure that a family relation is captured only if its arguments are human nouns. These include proper nouns, professions, titles, or generic human nouns like, man, woman, children, etc.

In order to identify family relations, a survey of the syntactic patterns that these relation nouns determine was carried out. Some of these patterns are similar regardless of the semantic relation they express.

[5] http://hagiaecclesia.blogspot.com/2009/05/sao-pacomio-c.html.

For example, the following sentences: *O João é pai do Pedro* "John is Peter's father", *O João é tio do Pedro* "John is Peter's uncle", and *O João é irmão do Pedro* "John is Peter's brother" all have the same syntactic structure: *João* is always the subject, the verb *ser* "to be" links the subject to the noun referring the type of relation, and the `PP` with *Pedro* is governed by the relation noun. Instead of making a specific rule for each relation, a general rule is constructed in order to capture all these cases, while other rules specify the relation type present in each sentence.

Next, before detailing how these rules were built, a brief overview of the XIP syntax and of the dependency types used to extract family relations in presented. XIP syntax is based on regular expressions. For clarity, XIP rules are split in three parts (all of these three parts are optional):

```
|pattern| if <condition> <dependency terms>.
```

- The `pattern` part regards the nodes of a given sentence. A node or a chunk is composed by one or more words, like a noun, a verb, an article, a preposition, etc. The most common nodes are `NP` (noun phrase), `VP` (verb phrase), `PP` (prepositional phrase). It is also possible to verify the presence of some word features, for example the gender (masculine or feminine), and the number (singular or plural). The features used in the extraction of family relations are related to gender, number, the lemma of a word, the feature "relative" that is present in every word related to a family relation (father, mother, uncle, brother, sister, etc.) and finally we verify if a noun may represent a person through the presence one of these features: people, individual, human or profession.
- The `<condition>` part is an if clause, which is used to verify some conditions, like the presence of a dependency that conveys a some specific meaning in the sentence. For example the dependency `SUBJ` identifies the subject of the verb; for the sentence *O João é irmão do Pedro* "John is Peter's brother" the subject dependency is created: `SUBJ(é,João)`.
- Several dependencies are used in the relation extraction task:
  - The `PREDSUBJ` dependency links a copula verb like *ser* "to be" to a predicative noun, an adjective or adverb; for example, in the sentence, *O João é irmão do Pedro* "João is Pedro's brother", we have a `PREDSUBJ` dependency between the verb *ser* "to be" and the noun *irmão* "brother".
  - The `APPOSIT` dependency links the noun with an apposite; in the sentence *O João, o irmão do Pedro, fez isso* "João, Pedro's brother, did that", the following dependency is extracted: `APPOSIT(João,irmão)`.
  - The coordination dependency `COORD` links elements in a coordination chain. For instance, if a verb operates on a noun and that noun has a coordination dependency with another noun then the second noun is also operated upon by the same verb (and is in the same syntactic relation to the verb as the first noun).
  - The `HEAD` dependency relates the nucleus of a chunk with the chunk itself; in the previous sentence, the noun *João* is the head of the nominal chunk *O João*.

– Finally, the `<dependency terms>` determines the action of the rule. In the relation extraction task, this part of the rule creates the family dependency.

The rule presented in Figure 2 is used to determine if a sentence has a family relation. The first part imposes restrictions on the tree structure of the sentence: this must have a `NP` that is composed by something (?*) and a noun presenting semantic, human-related features (people and individual, human, people or profession); this `NP` can be followed by some optional `NP` or `PP` (these elements often indicate the age or the profession of the first `NP`; for the kind of relations here targeted, that information is not relevant); after those optional chunks there must be a `VF` with a verb whose lemma is *ser* "to be"; Then we have another `NP` with a noun that has the "relative" feature (like "father", "uncle", "brother"); finally, we have again a `PP` with a noun referring to a human.

Afterwards, we verify some conditions by way of an if clause. The noun in the first `NP` has to be the head of the chunk, the same thing happens in the `PP`. The head of the first `NP` must be the subject of the verb, and the noun of the second `NP` has to be in a `PREDSUBJ` relation with the verb.

At last, and if no previous family relation has been detected, we extract the `FAMILY` relation between the two human nouns and in this first phase we keep the type of the relation that is present in the second `NP`.

```
| NP#1{?*, noun#2[people, individual]; noun#2[human];
          noun#2[people]; noun#2[profession]},
  PUNCT*, NP*, PP*, PUNCT*,
  VF{verb#3 [lemma:ser]}, NP{noun#4[relative]},
  PP#5{?*,noun#6[people, individual]; noun#6[human];
         noun#6[people]; noun#6[profession]} |

 if( HEAD(#2,#1) & HEAD(#6,#5) & PREDSUBJ(#3,#4) &
      SUBJ[PRE](#3,#2) &
      ~FAMILY(#2,#6) & ~FAMILY(#4,#2,#6))

  FAMILY(#4,#2,#6)
```

Fig. 2. XIP rule: creates a new Family relation

Next, as shown in Figure 3, a set of rules is used to remove the type of relation from the first argument of the `FAMILY` dependency and to add the relation type as a feature of the dependency, now with only two arguments. In the if clause we erase the previous dependency created and verify which is the lemma for the first argument, e.g.*primo* "cousin"; if that is the case, then we create a new dependency `FAMILY` with the feature `cousin` included. For each different lemma referring a family relation, a similar rule has to be made.

It is also necessary to identify the gender of both arguments in a family relation. In order to do that four features have been created: `1M`; `2M`; `1F` and `2F`. These features indicate whether the first or the second argument is either

```
if( ^FAMILY(#1,#2,#3) & #1[lemma:primo])
   FAMILY[cousin=+](#2,#3)
```

Fig. 3. XIP rule: remove the type of relation from the arguments.

masculine or feminine. Figure 4 contains an example of this type of rules. In this rule the dependency `FAMILY` is checked to determine if it does not have the gender feature already, and if the first argument has the feature `masc` (masculine) and does not have the feature `fem` (feminine). If these conditions are satisfied, then the feature `1M` is added, indicating that the first argument is masculine.

```
if( ^FAMILY(#1,#2) & ~FAMILY[1M](#1,#2) & ~FAMILY[1F](#1,#2) &
     #1[masc] & ~#1[fem])
    FAMILY[1M=+](#1,#2)
```

Fig. 4. XIP rule: add the gender feature to the relation.

Although these rules identify most of the cases, some given names are ambiguous and all family names are not marked for gender [Baptista et. al., 2006]. In those cases we decided not to include the gender of that argument.

To solve some of the ambiguity present, some rules were further refined. These take into account the fact that often the noun expressing the relation type indicates the gender of one of its arguments. For example in the sentence *Saraiva é tio de Silva* “Saraiva is Silva’s uncle” we know that the family proper name *Saraiva* represents a male person because of the relation noun here used.

We have also removed some ambiguity in nouns by changing its gender based on the article that usually precedes the noun. Notice that in Brazilian Portuguese the article is usually not used, while in European Portuguese the presence or absence of the article is meaningful and its use is related to the degree of notoriety or the familiarity of the speaker with the individual.

Several idiomatic expressions also convey familiar relations, for example the sentence *O João e a Joana deram o nó* (literally, “João and Joana have tied the knot”) means that these two people got married. These cases are different from the ones we presented before because, instead of a global rule, each one must have a specific rule since it conveys a single relation type and have a specific syntactic structure.

Many relation nouns can be used in combination with other lexical elements in order to distinguish or to define in a more specific way a basic family relation; for example adoptive father, foster father, twin brother, etc. To handle these type of relations we use regular expressions in the lemmas. For example, if the lemma: “pai” is used, this expression will only match the word *pai* “father”, however if we use as lemma: “pai(%c*)” then it will also match *pai adoptivo* “adoptive father”.

Symmetric relations, like *irmão* "sibling", *cunhado* "brother-in-law" and *primo* "cousin" also require specific rules to deal with sentences such as *João e o Pedro são primos* "João and Pedro are cousins", which are not captured by the general rules. The symmetry property consists in the arguments of the relation noun being able to appear coordinated in the subject position. Non-symmetric relation nouns cannot enter this syntactic pattern: *João e o Pedro são pais* "João and Pedro are fathers" (this sentence would be acceptable but the "father" relation would not hold between the two human nouns).

So far, all relations presented here have two arguments, but some expressions may appear to have more than two, for example, the sentence *O João e o Carlos são tios do Pedro* "João and Carlos are Pedro's uncles".

In this sentence, the family relation "uncle" holds not only between *João* and *Pedro*, but also between *Carlos* and *Pedro*. In order to capture cases like this, where there is more than one relation in the same sentence, the following rules were made:

```
if( FAMILY(#1,#2,#3) & #1[pl] & COORD(#4,#2) & COORD(#4,#5))
   FAMILY(#1,#5,#3)

if( FAMILY(#1,#2,#3) & COORD(#4,#3) & COORD(#4,#5))
   FAMILY(#1,#2,#5)
```

Fig. 5. XIP rule: creates an additional relation in cases where a sentence has a relation with three arguments.

These rules verify if there is a coordination dependency between one of the arguments of the previously detected relation and another entity. Whenever this condition is met the same relation is propagated to the second entity.

The last special case is related to anaphora. Anaphora may be defined as the referential relation that holds between two instances in a text: an expression (the *anaphor*) that refers to another expression, which has occurred previously in the same text (the *antecedent*). A module for anaphora resolution is currently being developed for the Portuguese language at L2F/INESC-ID by another researcher. Once this module is in place, its results are likely to improve the relations extraction task.

Among the different types of anaphoric devices, zero anaphora [Mitkov, 2002] constitutes a particularly challenge to anaphora resolution systems. Zero anaphora holds between a void anaphor, i.e. an empty syntactic slot, and its antecedent; it is a form of ellipsis, used to avoid word repetition.

For example, in the sentence *O João é irmão do Pedro mas cunhado do Carlos* "João is Pedro's brother but [he is] Carlo's brother-in-law" the subject of the relation noun *cunhado* is also *João* but it has been zeroed not to be repeated, since it refers to the subject of the first coordinate sentence [Pereira, 2010].

Zeroing also occurs in discourse following turn taking (e.g. answering a question), like in the sentence: *É tio do Pedro* "[He] is Pedro's uncle". In these cases, the zeroed anaphor may be identified but its reference can not be solved at this stage, for its antecedent is not in the current sentence. Therefore, a dummy node is created, inheriting the features that the relation extraction rules can recover from the context (v.g. in the sentence above, the masculine, singular, third person). In the extracted relation, a dummy feature *A0* is added. At a future stage of the anaphora resolution module, this information would be used to correctly calculate the antecedent of this zero anaphor.

5 Evaluation

For the evaluation of this task, we use three metrics which are common to other research papers in this field. These three metrics are:

$$Precision = \frac{Correct\_Relations}{Relations\_Identified}$$

$$Recall = \frac{Correct\_Relations}{Total\_Relations}$$

$$f-measure = \frac{2*precision*recall}{precision+recall}$$

At this stage, we only consider for evaluation purposes the patterns where the relation noun and its argument named entities are explicitly present in the text, such as in the examples discussed above. Other cases will be discussed in the final section of the paper. We have implemented 99 rules to extract these family relations.

Since the *Family* category has been treated differently by the systems presented in Chapter 2, and also because the evaluation corpus is different in almost every investigation (except in joint evaluations), it is not possible to compare rigorously the system here presented with those referred to above.

Two evaluation corpora were used to evaluate the relation extraction task. The first evaluation corpus is a text containing the biography of all Portuguese kings. We decided to use this type of documents because they are very rich in family relations and they have common Portuguese names that should be easily identified as a person named entities by the system. These biographies were gathered from Wikipedia[6] and they were then manually annotated for the family relations they present.

The total number of family relations present in this evaluation corpus is 105. The annotation task was made after the implementation phase, so that we could not adapt our rules to these specific cases.

We performed a second evaluation because we noticed that the Portuguese Kings biographies have many implicit relations, assuming that whoever is looking at the text already knows something about the person whose life is being described. An automatic system does not work so, and this may influence the results.

[6] http://www.wikipedia.pt

The second evaluation corpus is then made up of the first 110 sentences containing at least one relation noun, from a list of about 100 names, and retrieved from the CETEMPúblico corpus[7].

The results of the first evaluation corpus are presented in Table 1. Some factors hindered these results, namely the implicit relations, mentioned above, false positive relations, identification of relations with the incorrect arguments, and other errors due to an incorrect performance of the NER module, which is beyond the scope of this task.

For example, in the following sentence: *O acordo foi firmado em 1174 pelo casamento de Sancho, então príncipe herdeiro, com a infanta Dulce Berenguer, irmã mais nova do rei Afonso II de Aragão.* The relation that should have been extracted is:

`FAMILY_SPOUSE(Sancho, infanta Dulce Berenguer)`

But the relationship that has actually been extracted is:

`FAMILY_SPOUSE(Sancho, infanta)`

The problem in this example is that a composite node between the title *infanta* "princess" and *Dulce Berenguer* should have been created; if it had, the relation would be correctly established. Therefore, results were reassessed counting cases like this as correct, in order to see their impact on the overall results. As seen on the right side of Table 1, these results are indeed better.

Table 1. Results of the first evaluation corpus (the right table considers as correct the relations with errors on the identification of people names).

Precision	Recall	F-measure
0.59	0.19	0.29

Precision	Recall	F-measure
0.71	0.23	0.35

Table 2 presents the results for the second evaluation corpus. This second corpus contained 110 sentences; only 21 of these had an explicit family relation, while the remaining 89 contained at least one relation word, but no explicit family relation to associate pairs of entities. As was mentioned above, only explicit relations with both entities expressed were considered for evaluation in this paper.

Table 2. Results of the second evaluation corpus

Precision	Recall	F-measure
0.71	0.24	0.36

[7] `http://www.publico.pt/`

Some of the cases that lowered the precision are due to the incorrect identification of the arguments, i.e., the relation is explicitly present in the sentence, but the rule is unable to correctly capture one of the arguments. The following example exhibits this problem:

Quanto ao Troféu BMW 320iS, Jorge Petiz ultrapassou o seu irmão Alcides a meio da corrida para obter uma vitória fácil, com 2,382′′ de avanço, deixando o 3o, António Barros, a 4,788′′.

In this case, the name *Alcides* is an apposite to the word *irmão ("brother")*; only this half of the relation (*FAMILY_A0_1M_SIBLING*) is correctly identified but a dummy (anaphoric) first argument (A0) is construed since the system is currently unable to identify *Jorge Petiz*, the sentence subject, as the relation's first argument. In order to do so, the reference of the possessive pronoun would have to be resolved, which is outside the scope of this paper.

The evaluation corpus also contains some foreign names that are not identified as person named entities by the system, thus preventing the relation extraction. This problem derives from the NER module and not from the relation extraction module. Therefore, it was decided to add the missing names to the XIP's lexicon, and to perform a new evaluation. As expected the results are better (see Table 3), particularly the recall.

Table 3. New Results of the second evaluation corpus

Precision	Recall	F-measure
0.70	0.33	0.45

6 Conclusions and future work

The extraction of semantic relations between named entities in text is a very complex and challenging task. This paper reports a first attempt to extract from texts semantic relations between named entities, using the NLP chain built at L2F. Results from two different corpora are still unsatisfactory. The main problems detected came from (i) the insufficient performance of the named entity recognition module, which produced much of the incorrect matching of the relations' arguments; and (ii) the limited coverage of the syntactic-semantic dependency extraction module (deep parser). Several dependency rules were built to capture new patterns in order to improve the recall measure of the task.

One of the dependencies that requires further attention is apposition, as in the sentence: [. . .] *o grupo tinha em seu poder o tenente-coronel Mike Couillard, 37 anos, e o seu filho Matthew, dez anos.* "the group had in his power the lieutenant-coronel Mike Couillard, age 37, and his son Matthew, age 10". Even if the age insertions were sorted out, the system still does not correctly handle titles, when they are adjoined to a proper name. The same happens with relation nouns, like *filho* "son", often appearing in front of the named entity.

Data from the corpora have shown that family relations in texts are most often expressed by way of incomplete mention, using just the relation noun, as in the following example: *No regresso, o meu pai já vinha a dormir.* "On the way home, my father was already sleeping". In this case, a parenting relation is present, but the relation noun also designates a person whose reference still needs to be established in the previous discourse. On the other hand, the possessive refers to the enunciation subject, which calls for a much complex calculation, across direct and reported speech. Correference resolution is therefore an unavoidable track parallel to relation extraction task.

References

[Baptista et. al., 2006] Jorge Baptista, Fernando Batista, Nuno Mamede, Building a Dictionary of Anthroponyms, In PROPOR 2006 - Computational Processing of the Portuguese Language, Springer Verlag, Berlin Heidelberg, vol. 3960, pages 21-30, Itatiaia, Brazil, May 2006.

[Cardoso, 2008] Nuno Cardoso: Rembrandt - reconhecimento de entidades mencionadas baseado em relações e análise detalhada do texto, Chapter 11, pages195-211, 2008.

[Culotta & Sorensen, 2004] Aron Culotta, Jeffrey Sorensen: Dependency tree kernels for relation extraction. In ACL'04: Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, page423, Morristown, NJ, EUA, 2004. Association for Computational Linguistics.

[Mamede, 2007] Nuno Mamede: A Cadeia de Processamento XIP. L2F (Laboratório de Sistemas de Língua Falada, Maio 2007.

[Medeiros, 1995] J. C. Medeiros, Processamento Morfológico e Correcção Ortográfica do Português. Master's thesis, Instituto Superior Técnico - Universidade Técnica de Lisboa, Portugal, 1995.

[Mitkov, 2002] Ruslan Mitkov, Anaphora Resolution, Pearson ESL, 1st Edition, 2002, ISBN: 978-0582325050, Chapter 1.

[Mota & Santos, 2008] Cristina Mota, Diana Santos: Desafios na avaliação conjunta do reconhecimento de entidade smencionadas: O Segundo HAREM, 2008.

[Freitas et. al., 2009] Cláudia Freitas, Diana Santos, Cristina Mota, Hugo Gonçalo Oliveira, Paula Carvalho: Relation detection between named entities: report of a shared task. In DEW'09: Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions, pages 129-137, Morristown, NJ, EUA, 2009. Association for Computational Linguistics.

[Pardal, 2007] Joana Pardal, Manual do Utilizador do RuDriCo. Technical report, Instituto Superior Técnico - Universidade Técnica de Lisboa, Portugal, 2007.

[Ribeiro et. al., 2003] Ricardo Ribeiro, Nuno Mamede, and Isabel Trancoso, *Computational Processing of the Portuguese Language: 6th International Workshop, PROPOR 2003, Faro, Portugal, June 26-27, 2003*, volume 2721 of Lecture Notes in Computer Science. chapter Using Morphossyntactic Information in TTS Systems: Comparing Strategies for European Portuguese. Springer, 2003.

[Pereira, 2010] Simone Pereira, Linguistic Parameters for Anaphora Resolution, MA thesis, Univ. Algarve, Faro, 2010.

[Xerox, 2003] Xerox, Xerox Incremental parser – Reference Guide, 2003.