

University of Sofia "St. Kliment Ohridski"

Faculty of Mathematics and Informatics

Department of Information Technologies

**A Light-weight Approach to Coreference Resolution for
Named Entities in Text**

by Marin Dimitrov

**under the supervision of Dr. Kalina Bontcheva (University of Sheffield, UK) and
Dr. Hamish Cunningham (University of Sheffield, UK)**

MSc programme in Artificial Intelligence

faculty number 42020

**Sofia, Bulgaria
20/02/2002**

Abstract. This thesis presents a lightweight approach to resolving pronominal coreference in the case when the antecedent is named entity. It falls under the category of the so-called "knowledge poor" approaches which do not rely extensively on linguistic and domain knowledge. We provide a practical implementation of this approach as a component of the General Architecture for Text Engineering (GATE). The results of the evaluation show that even such shallow and inexpensive approaches provide acceptable performance for resolving the pronoun anaphors of named entities in texts.

Acknowledgements

I would like to use the opportunity to thank to few of the people who helped and influenced my development in the recent years:

- Kalina Bontcheva and Hamish Cunningham (University of Sheffield) for their constant help, guidance and encouragement
- Diana Maynard (University of Sheffield) for her tireless clarifications, corrections and contributions to this work
- Atanas Kiryakov (OntoText) for stimulating my professional career and research interests, for providing the time frame that made this work possible and for the discussions on this work
- Kiril Simov (OntoText and Bulgarian Academy of Sciences) for stimulating the research interests in me and many other students
- The GATE team - Hamish, Di, Valy, Cristi and Kali - for the nice working atmosphere I was provided with in the last few months and for their excellent work on the GATE project
- Vladimir Alexiev (WorkLogic) for encouraging an attitude of striving for excellence
- most of all to my parents

INTRODUCTION	6
1 INFORMATION EXTRACTION.....	8
1.1 INFORMATION EXTRACTION OVERVIEW	8
1.2 INFORMATION EXTRACTION TASKS	10
1.2.1 <i>Named Entity recognition</i>	11
1.2.2 <i>Coreference Resolution</i>	12
1.2.3 <i>Template Element construction</i>	13
1.2.4 <i>Template Relation construction</i>	14
1.2.5 <i>Scenario Template construction</i>	14
1.3 SCORING.....	15
1.4 INFORMATION EXTRACTION PROGRAMS AND COMPETITIONS	16
1.4.1 <i>Message Understanding Conference (MUC)</i>	16
1.4.2 <i>Automatic Content Extraction (ACE)</i>	17
2 COREFERENCE RESOLUTION	19
2.1 INTRODUCTION	19
2.2 COREFERENCE TYPES.....	20
2.2.1 <i>Pronominal coreference</i>	20
2.2.2 <i>Proper names coreference</i>	21
2.2.3 <i>Apposition</i>	21
2.2.4 <i>Predicate Nominals</i>	21
2.2.5 <i>Identical Sets or Types</i>	22
2.2.6 <i>Function-value coreference</i>	23
2.2.7 <i>Ordinal Anaphora and One-anaphora</i>	23
2.2.8 <i>Part-whole coreference</i>	23
2.3 SPECIAL CASES	24
2.3.1 <i>Pleonastic It</i>	24
2.4 POPULAR ALGORITHMS & SYSTEMS	26
2.4.1 <i>Lappin & Leass</i>	26
2.4.2 <i>Breck Baldwin's CogNIAC</i>	27
2.4.3 <i>Ruslan Mitkov's knowledge-poor approach</i>	27
3 THE GENERAL ARCHITECTURE FOR TEXT ENGINEERING (GATE)	28
3.1 GATE OVERVIEW.....	28
3.1.1 <i>GATE - a General Architecture for Language Engineering</i>	28
3.1.2 <i>Resources in GATE</i>	28
3.1.3 <i>Applications and Databases</i>	29
3.1.4 <i>Annotations</i>	29
3.1.5 <i>Built-in Processing Resources</i>	30

3.1.6	<i>English Tokeniser</i>	31
3.1.7	<i>Gazetteer</i>	32
3.1.8	<i>Sentence Splitter</i>	32
3.1.9	<i>POS Tagger</i>	32
3.1.10	<i>Named Entity Transducer</i>	32
3.1.11	<i>Orthographic Matcher</i>	33
3.2	JAPE OVERVIEW	33
4	IMPLEMENTATION	36
4.1	THE ACE CORPUS	36
4.2	REQUIREMENTS DEFINITION	36
4.3	INITIAL SET OF RULES	37
4.4	ANALYSIS OF THE ACE CORPORA	40
4.4.1	<i>Total pronouns</i>	41
4.4.2	<i>Distribution of pronouns by type</i>	41
4.4.3	<i>Pleonastic It Statistics</i>	42
4.5	DESIGN OF THE COREFERENCE RESOLUTION MODULE.....	43
4.6	ARCHITECTURE	44
4.7	QUOTED SPEECH SUB-MODULE	45
4.8	PLEONASTIC IT SUB-MODULE	46
4.8.1	<i>Extension of Modal Adjectives and Cognitive Verbs</i>	46
4.8.2	<i>Extended Rules</i>	47
4.9	PRONOMINAL RESOLUTION SUB-MODULE.....	49
4.9.1	<i>Resolution of she, her, her\$, he, him, his, herself, himself</i>	53
4.9.2	<i>Resolution of it, its, itself</i>	53
4.9.3	<i>Resolution of I, me, my, myself</i>	55
4.9.4	<i>Unresolved Pronouns</i>	57
4.10	RESULTS AND ERROR ANALYSIS	57
5	FUTURE WORK	61
	BIBLIOGRAPHY	62
	APPENDIXES	65
	APPENDIX A - LIST OF ALL ACRONYMS.....	65
	APPENDIX B - SOURCE CODE	66

Introduction

Anaphora resolution and the more general problem of coreference resolution are very important for several fields of Natural Language Processing such as Information Extraction, Machine Translation, Text Summarization and Question Answering Systems.

Because of its importance, the problems are addressed in various works and many approaches exist. The approaches differ in the technology they use for the implementation (symbolic, neural networks, machine learning, etc.), in the domain of the texts that they are tuned for, in their comprehensiveness (e.g. is only pronominal anaphora considered) and in the results achieved.

This work falls under the class of "knowledge poor" approaches to pronominal resolution, which are intended to provide inexpensive (in terms of the cost of development) and fast implementations that do not rely on complex linguistic knowledge, yet they work with sufficient success rate for practical tasks.

Our approach follows the salience-based approach in existing implementations, which perform resolution following the steps:

- identification of the antecedents in the context of the pronoun
- inspecting the context for candidate antecedents that satisfy a set of consistency restrictions
- assigning salience values to each antecedent based on a set of rules and factors
- choosing the candidate with the best salience value

The approaches that influenced our implementation were focused on anaphora resolution of certain set of pronouns in technical manuals. The goal of our work is resolution of pronoun anaphora in the case where the antecedent is a named entity - a person, organization, location, etc. The implementation relies only on the part-of-speech information, named entity recognition and orthographic coreferences existing between the named entities. No syntax parsing, focus identification or world-knowledge based approaches were employed. The texts that we used for the evaluation were newswire articles part of the ACE (Automatic Content Extraction competition) training corpus. The evaluation showed that acceptable results could be achieved with such inexpensive approaches.

We provide an implementation of the approach, available as a component integrated with the General Architecture for Text Engineering (GATE) - a Language Engineering framework and set of tools developed by the University of Sheffield.

The structure of this thesis is:

- chapter I presents an overview of Information Extraction where coreference resolution is one of the major tasks. The chapter presents details about the goals of IE and the performance achieved by such systems on the Message Understanding Conference (MUC) tasks. An overview of the Automated Content Extraction (ACE) project, which is the successor of MUC, is made.

- chapter II presents an introduction to the problem of anaphora resolution and the more general problem of coreference resolution. Various types of anaphoric occurrences are described. A brief overview of the most popular "knowledge-poor" approaches is made
- chapter III is an overview of GATE (which is used for the implementation). A short description is presented of the ANNIE system on which our implementation depends. The JAPE engine, which is used for the work of certain submodules in our implementation is presented.
- chapter IV presents our work. It contains the requirements and the goals that the implementation tries to achieve. The results of our analysis of the ACE corpus and the heuristic patterns we have identified are presented. A description of the submodules (pleonastic it module, quoted text module and pronominal module) in our implementation is made. Details specific to each set of pronouns are presented. Finally the results and an analysis of the problems is made
- chapter V presents our ideas for future extensions of the functionality that will be made in the context of the GATE system.

1 Information Extraction

1.1 Information Extraction overview

The amount of data available nowadays is enormous. As estimated in [Lyman00], there are about 240 terabytes of information produced each year in books, newspapers, periodicals and office documents¹. Almost 10% (23 terabytes) of this information is textual. Finding the facts of interest present there is very difficult, because this information is unstructured and mostly available in natural language form, so in addition to locating the *documents* of interest (using Information Retrieval techniques or simple keyword search provided by a web search engine such as Google) much time is usually spent on further reading and analyzing of the texts in order to extract the *facts* of interest.

Information Extraction (IE) is the process of analyzing unstructured texts and extracting the information relevant to some problem into a structured representation, or as described in [Grishman97] - the process of *selective information structuring*.

The information relevant to the problem is usually divided into:

- *entities* - persons, organizations, locations, etc. that are located in the text
- *attributes* that are related to the entities (e.g. the title of the person or the type of the organization)
- *facts* - the relations that exist between the entities (e.g. the company a person works for)
- *events* in which entities participate

An IE system is usually designed to find only some of the available types described above and to varying degree of accuracy: for example identifying entities present in the text is a relatively easy and very precise process, while extracting facts and events from the text is quite challenging task performed with relatively low quality.

The main characteristics the IE process are:

- *It works with unstructured sources* such news articles, technical manuals, legal documents, speech transcripts or radio broadcasts. All these texts have no predefined structure - they are mostly natural language form. Analyzing some of the sources is more difficult than other because of their low quality (for example speech transcripts) which additionally affects the final results.

¹ The total amount of information produced each year is estimated in [Lyman00] to be 2,000,000 terabytes

- *It locates the relevant information and ignores the non-relevant.* As mentioned earlier finding documents that are relevant to some predefined problem is not sufficient because the texts should additionally be read and analyzed. IE tries to extract facts from the text that are relevant to the problem. For example one may be interested in acquiring the list of companies and their managers mentioned in articles from the Financial Times. With the proper IE system the user would not have to read all the articles and locate the names of companies and their managers - the system should be able to locate this information and structure it in the proper output format (for example an Excel sheet).
- *It extracts the information in a predefined format and the result is a structured representation of the facts of interest available in the texts.* Consider the following news fragment:

Airlines have grabbed much of the spotlight, with the high profile failures of Swissair and Belgium's Sabena, Canada3000, and Ansett in Australia.

The IE processing could transform the free form text into the following structured presentation (conforming to some imaginary predefined template, which could be XML/SGML or some spreadsheet):

```
<TEMPLATE name='template01' source='Financial Times'>
  <ORGANIZATION id='100'>
    <NAME>Swissair</NAME>
    <TYPE>company</TYPE>
  </ORGANIZATION>
  <ORGANIZATION id='101'>
    <NAME>Sabena</NAME>
    <TYPE>company</TYPE>
    <LOCATION id='201'>
  </ORGANIZATION>
  <LOCATION id='201'>
    <NAME>Belgium</NAME>
  </LOCATION>
  <ORGANIZATION id='101'>
    <NAME>Sabena</NAME>
    <TYPE>company</TYPE>
    <LOCATION id='201'>
  </ORGANIZATION>
  <ORGANIZATION id='102'>
    <NAME>Canada3000</NAME>
    <TYPE>company</TYPE>
```

```
</ORGANIZATION>  
  
<ORGANIZATION id='103'>  
  <NAME>Ansett</NAME>  
  <TYPE>company</TYPE>  
  <LOCATION id='202'>  
</ORGANIZATION>  
  
<LOCATION id='202'>  
  <NAME>Australia</NAME>  
</LOCATION>  
  
</TEMPLATE>
```

- *It is usually domain dependent* - an IE system is usually designed and trained to deal with texts that are specific to some domain. Some of the patterns and rules the system will employ are not always applicable to a different problem. For example a system that is designed to extract facts about bankrupting companies from news articles will perform with much lower precision on technical manuals. Building IE systems that are relatively domain independent or could be ported to different domain more easily is an open issue.

1.2 Information Extraction tasks

The tasks performed by IE systems usually differ, but the following classification from the seventh (and final) Message Understanding Conference (MUC) characterizes the most common tasks:

- Named Entity recognition (NE) - finds the entities in the text
- Coreference Resolution (CO) - finds identities between entities
- Template Element construction (TE) - finds the attributes of the entities
- Template Relation construction (TR) - finds the relations between entities
- Scenario Template construction (ST) - finds the events in which entities participate

The quality of the results produced from each task strongly depends on the quality of the results produced from the tasks performed before it (the 'snowball effect'), so while the precision of the Named Entity task is very high, the quality of the results from the Scenario Template task is usually low.

A more detailed description of the tasks follows. It is mostly based on [Chinchor98] and [Cunningham99].

1.2.1 Named Entity recognition

During the Named Entity recognition phase an IE system tries to identify all mentions of proper names and quantities in the text such as:

- names of persons, locations and organizations
- dates and times
- mentions of monetary amounts and percentages

The quality of the results produced from the NE task is usually very high - the best F-Measure² achieved from an IE system in the MUC-7 competition was 93% while humans achieved 98%.

As noted in [Cunningham99], this task is weakly domain dependent - changing the type of the texts being analyzed may or may not induce degradation of the performance levels.

The relative percentage of the three types of named entities depends on the domain of the texts analyzed. Usually proper names account for 70-80% of the named entities, dates and times account for 10-20% and the number of quantities mentioned in the text is less than 10% (analysis of the distribution of named entities in the MUC6 and MUC7 corpora is available in [Marsh98] and [Sundheim95]).

Note that the definition of an entity is domain dependent. It may be a person/organization /date/percentage in the case of analyzing financial news articles, but it may for example be a product name (if texts being analyzed are product catalogs) or a protein/molecule name (in the case of scientific texts).

² The F-Measure metric will be explained in details in the next section

entities. The following example shows correctly identified pronominal coreference chains (*Lisette₁, her₁, her₁, she₁, her₁*) and (*Stephen₂, his₂*).

Lisette₁ began to tell a story about her₁ day at school. Stephen₂ watched her₁ as she₁ spoke, his₂ eyes scrutinizing her₁ face.

The coreference task is usually domain specific but some of the rules and heuristics used are applicable independently of the domain of the texts analyzed. The F-Measure for this task achieved by the IE systems in MUC-7 is relatively low, about 62%, but properly resolving some kinds of coreference is usually difficult even for humans - human annotators achieved about 80%.

1.2.3 Template Element construction

The goal of this task is to find the attributes describing the entities identified from the NE task. As a result, with each entity is associated some descriptive information, in addition to its name (identified during NE recognition). The definition of an attribute is domain dependent. For example persons could be characterized by title/alias, organizations could be characterized by type/location. The attributes for an entity, extracted during this phase, are grouped into a template element for this entity. The following text³ is an extract from the MUC evaluations. The descriptions of some of the entities are shown next:

Fletcher Maddox, former Dean of the UCSD Business School, announced the formation of La Jolla Genomatics together with his two sons. La Jolla Genomatics will release its product Geninfo in June 1999. L.J.G. is headquartered in the Maddox family's hometown of La Jolla, CA.

```
entity {
  ID = 1,
  NAME = "Fletcher Maddox "
  DESCRIPTOR = "Former Dean of USCSD Business School"
  CATEGORY = person
}

entity {
  ID = 2
  NAME = "La Jolla Genomatics"
  ALIAS = "LJG"
  DECSRIPTOR = ""
  CATEGORY = organization
}
```

³ Available at http://www.itl.nist.gov/iaui/894.02/related_projects/muc/

```
entity {  
    ID = 3  
    NAME = "La Jolla"  
    DESCRIPTOR = "the Maddox family hometown"  
    CATEGORY = location  
}
```

The performance achieved by the MUC-7 systems is about 87% (F-Measure), while human annotators achieved 93%.

1.2.4 Template Relation construction

The goal of the TR task is finding the relationships that exist between the template elements extracted from the text (during the TE task). Just like the definition of an entity and entity attributes depend on the problem and the nature of the texts being analyzed, the relationships that may exist between template elements is domain dependent too. For example persons and companies may be related by `employee_of` relation, companies and locations may be related by `located_of` relations, companies may be interrelated by `subdivision_of` relations.

For the sample text from the MUC evaluations the following relations may be identified:

```
employee_of (Fletcher Maddox, UCSD Business School)  
employee_of (Fletcher Maddox, La Jolla Genomatics)  
product_of(Geninfo, La Jolla Genomatics)  
location_of(La Jolla, La Jolla Genomatics)  
location_of(CA, La Jolla Genomatics)
```

The quality of the results of the TR task is higher than the results of the TE task - the best MUC-7 system achieved about 76% F-Measure.

1.2.5 Scenario Template construction

The ST task tries to extract information about the events in which the template elements identified in the text participated. Again, the definition of what an event is and what information is relevant to an event is domain specific. The MUC-7 systems performed quite poorly on this task - the best F-Measure achieved was 51%, but human annotators achieved poor performance too - about 85%.

The events identified for the MUC sample may look like:

```

company-formation-event {
    PRINCIPAL = "Fletcher Maddox"
    DATE = ""
    CAPITAL = ""
}

product-release-event {
    COMPANY = "La Jolla Genomatics"
    PRODUCES = "Geninfo"
    DATE = "June 1999"
    COST = ""
}

```

1.3 Scoring

In order to evaluate the performance of IE systems and compare it to the performance of similar systems or humans, several evaluation metrics were defined, the most popular being precision, recall and F-measure.

Precision is the number of correctly identified items (entities or template slots) as percentage of the total number of identified items. Note that some items may not be identified at all by the system, but the precision metric will not degrade. The degradation in precision is induced only by incorrectly identified items. In the following formula for the precision metric, *correct* is the number of correctly identified items (correct answers), and *produced* is the total number of items identified (answers produced):

$$precision = \frac{correct}{produced}$$

Recall is the number of correctly identified items as percentage of the total number of items available in the text. Here the degradation is induced by items not being identified. In the following formula, *correct* is the number of correctly identified items (correct answers), and *key* is the total number of items available in the text (possible answers):

$$recall = \frac{correct}{key}$$

High precision may often be achieved at the expense of low recall and vice versa. A combined metric exists, *F-measure*, defined as:

$$F - measure = \frac{P * R}{\beta * P + (1 - \beta) * R}$$

In the above formula P is the precision, R is the recall, and the parameter β is the weight of the relative importance of precision/recall. Values close to 0 favour precision, values close to 1 favour recall

A value of 0.5 gives equal weight to precision and recall, which is the most commonly used form or F-measure:

$$F - measure = \frac{2 * P * R}{P + R}$$

1.4 Information Extraction Programs and Competitions

In order to stimulate the development of new IE systems and to create a common basis for the evaluation of their performance, several projects were established.

The first and most significant is the *Message Understanding Conference (MUC)* - a sequence of evaluations held every two years between 1987 and 1998 with the sponsorship of DARPA⁴.

The other most significant program is the *Automatic Content Extraction (ACE)* - a project sponsored by NIST⁵.

1.4.1 Message Understanding Conference (MUC)

The MUC organizers were responsible for defining the IE tasks (what information should be extracted) and the templates (how the extracted information should be structured) for each competition, preparing the test corpora according to the domain chosen for the IE tasks, developing the evaluation methodology and scorers. The domains chosen for evaluation were different (ranging from terrorist attacks to news articles about joint ventures and management changes in corporations) and the IE tasks included in each evaluation were also different.

The following table from [Chinchor98] presents the tasks included in the evaluations from MUC-3 to MUC-7 and the best performance achieved for each task (P, R and F are the best precision, recall and F-measure respectively):

⁴ US Defense Advanced Research Projects Agency

⁵ National Institute of Standards and Time

	NE	CO	TE	TR	ST
MUC-3	-	-	-	-	R < 50% P < 70%
MUC-4	-	-	-	-	F < 56%
MUC-5	-	-	-	-	F < 53%
MUC-6	F < 97%	R < 63% P < 72%	F < 80%	-	F < 57%
MUC-7	F < 94%	F < 62%	F < 87%	F < 76%	F < 51%

Table 1. The performance achieved for different tasks for MUC-3 through MUC-7. Missing value means that the task was not performed for this competition.

The MUC competitions stimulated the research and development of IE systems at that time because they made the comparison of their performance possible (by defining a common task, a common data set and common scorers for all participants). They also eased the exchange and implementation of successful ideas and approaches between the participants. As mentioned in [Appelt99], one of the main benefits is that *"people tried to do things that without MUC they probably wouldn't try—or maybe even think of"*.

1.4.2 Automatic Content Extraction (ACE)

ACE is a program similar to MUC and its objective is to stimulate developments in automatic content extraction from natural texts. The sources for the texts are newswire, broadcast news (that were processed with automatic speech recognition tools) and newspaper (processed with optical character recognition tools).

One important difference between MUC and ACE is that the latter uses various input sources such as speech recordings and images (scanned newspaper articles) and the sources are processed with the help of the ASR and OCR technologies, which are not mature yet. This induces substantial degradation in the quality of the input texts and leads to a decreased performance of ACE systems.

The general ACE tasks are detection of *Entities*, *Relations* and *Events* in texts. The first task may be considered as a combination of the Named Entity task, the Coreference resolution task and the Template Element construction task in MUC. The relation detection may be considered as equivalent to the Template Relation construction tasks in MUC. The event detection task corresponds to the Scenario Template one in MUC.

There is a slight difference in the types of entities that are detected in MUC and ACE. The former classifies entities as named entities (persons, locations, organizations), time expressions (time, date) and numeric expressions (percents, money). The latter limits entities to the following five types:

- Persons
- Organizations
- Locations - mountains, continents, rivers, etc.
- Geographical-Political Entities (GPE) - politically defined geographical regions such as countries, cities, etc. GPEs can also refer to governments and people
- Facilities - human made structures and buildings and elements of the transportation infrastructure

More information about ACE and its task definitions are available in [ACE00].

2 Coreference resolution

2.1 Introduction

The term *anaphora* denotes the phenomenon of referring to an entity, already mentioned in text, by different ways - most often with the help of a pronoun or a different name. As noted in [Mitkov99], *anaphora* comes from the Greek ἀναφορά, meaning “carrying back”.

Some examples of anaphoric reference are:

*Lisette*₁ began to tell a story about *her*₁ day at school. *Stephen*₂ watched *her*₁ as *she*₁ spoke, *his*₂ eyes scrutinizing *her*₁ face.

In this sentence, there are several references (like *her*, *she*, *he*) that point to the two entities, “Lisette” and “Stephen”, mentioned earlier in text. The reference that points back to some entity is called *anaphor* while the entity it refers to is called *antecedent*. So in the above example “Lisette” and “Stephen” are antecedents, while the pronouns are anaphora (the anaphora and their respective antecedent have the same subscript).

The process of finding the proper antecedent for each anaphora in text is called *anaphora resolution*. Such resolution is very important because without it the text would not be fully and correctly understood – without finding the proper antecedent, the meaning and the role of the anaphor cannot be realized.

As pointed out in [Lappin94] we could define an antecedent-anaphor relation *antecedes*(X, Y) which links together each anaphor Y and its respective antecedent X . This relation is transitive (if X is antecedent of Y and Y is antecedent of Z then X is also antecedent of Z) and reflexive (each entity X could be considered as referring to itself). The antecedent-anaphor relation is not symmetric, but a new relation – *coref*(X, Y) could be defined with the help of the *antecedes*(X, Y) one.

The relation of *coref*(X, Y) holds if at least one of the following holds:

- *antecedes*(X, Y)
- *antecedes*(Y, X)
- *antecedes*(Z, X) and *coref*(Z, Y)
- *antecedes*(Z, Y) and *coref*(Z, X)

The *coref* relation is reflexive, transitive and symmetric, thus it is an equivalence relation and it defines equivalence classes of the entities and their referents in the text. The equivalence classes (also called coreference chains) are defined as:

$$\text{equiv}(X) = \{Y \mid \text{coref}(X, Y)\}$$

In the above example, the entities and their referents form two coreference chains {Lisette, her, her, she, her} and {Stephen, his}.

The process that identifies the coreference chains in text is called *coreference resolution* - a task more general than anaphora resolution.

2.2 Coreference types

There are different kinds of coreference but not all of them are equally important for the coreference resolution task. For example the evaluation of the relative importance of the different kinds of coreference in [Bagga98] shows that while pronominal coreference constitutes about 20% of the coreferences observed in text, the demonstrative phrases coreference constitutes only about 2%, so for a coreference resolution system it is much more important to handle precisely pronoun referents.

The following classification of the basic coreference types is based on [Bagga98], [Hirschman97] and [Denber98].

2.2.1 Pronominal coreference

This is the most common type of coreference. It includes finding the proper antecedent for the following types of pronouns:

- personal: I, you,...me, him,...mine, yours...
- possessive⁶: my, your,...
- reflexive: myself, yourself,...

An example for pronominal coreference is ([Hirschman97]):

Every TV network_i reported its_i profits yesterday. They_i plan to release full quarterly statements tomorrow.

The relative importance of the three types of pronouns depends on the domain of the texts being analysed but the following values observed in technical texts and reported in [Mitkov01] are more or less representative:

	personal	Possessive	reflexive
% of pronouns	85.3%	13.5%	1.4%

Table 2. Pronoun distribution by type, reported in [Mitkov01]

⁶ Strictly speaking "mine", "yours", etc. are possessive pronouns, while "my", "your", etc. are possessive adjectives but the part-of-speech tagger used in this work classifies the former as personal pronouns and the latter as possessive pronouns.

The results reported show that proper resolution of the different kinds of pronouns will have different impact on the overall performance of a coreference resolution system (for example precise resolution of personal pronouns is much more important than resolution of reflexive pronouns)

2.2.2 Proper names coreference

This type of coreference links together all the variations of a name (of a person or a company for example) that are observed in text. For example:

President Clinton₁ will meet with Israeli and Palestinian leaders separately in Washington later this month. Clinton₁ will host Israeli Prime Minister at the White House on Jan. 20 and Palestinian leader two days later.

Additional variants and aliases of the same name may include: Mr. Clinton, Bill Clinton, etc.

2.2.3 Apposition

Appositives are usually used to provide some additional information for a named entity. The additional information is separated from the name of the entity by a comma and is usually placed immediately after or before the entity name. For example ([Hirschman97]):

Julius Caesar₁, the well-known emperor₁, ...

...the well-known emperor₁, Julius Caesar₁, ...

Appositional phrases are considered corefering with a named entity only when they occur in a noun phrase different from the one containing the named entity. If the apposition occurs as a modifier of the named entity in the same noun phrase no coreference should be identified:

...former U.S. senator George J. Mitchell...

2.2.4 Predicate Nominals

A predicate nominal (also known as subject complement or predicate complement) completes a reference to the subject of a clause. It always occurs after copular verb (is, seems, looks, appears, etc). For example:

George Bush₁ is the President of the United States₁

In this sentence, the phrase "*the President of the United States*" is a predicate nominal and thus it is considered as corefering with the subject "*George Bush*".

Coreference is *not* considered if only the possibility that an assertion is true is stated or if the assertion is negative as in the following examples:

Al Gore may be the President of the United States

Al Gore is not the President of the United States

2.2.5 Identical Sets or Types

In this type of coreference the anaphor and the antecedent both refer to sets which are identical or to identical types. In the context of the following example, "Protestant guerillas" and "the prisoners" refer to the same set of individuals, while "the Northern Ireland peace talks" and "the discussions" refer to the same activity:

*Threats to the resumption of **the Northern Ireland peace talks**₁ receded today after a **British cabinet minister**₂ entered **the Maze prison**₃ and pressed **Protestant guerrillas**₄ held **there**₃ to support continuing **the discussions**₁. After **she**₂ left, **the prisoners**₄ did what **she**₁ asked.*

The proper identification of such cases of coreference is very difficult because it requires some world knowledge (so that we could infer that Protestant guerillas held in the Maze prison are prisoners there). Two special cases could be considered, that are easier to be resolved.

In the first case there exists a synonymy or hyponymy⁷/hypernymy⁸ lexical relation between the anaphor and the antecedent, so finding the proper antecedent of an anaphor is easier if a lexical resource such as WordNet is used. An example is:

*This morning when I got on the subway a mother and **daughter**₁ boarded with me. At every stop **the girl**₁ raised her fists above her head and shouted, "Yaaaaayyyyy!"*

In the example above *girl* and *daughter* are synonyms, so the latter could be considered as candidate antecedent for the former.

In the second case the anaphor matches exactly or is a substring of the noun phrase of the antecedent or all the words of the noun phrase of the anaphor appear in the noun phrase of the antecedent too. An example is:

*The majority of registered shareholders choose **the Bank's manager**₁. **The manager**₁ is responsible for the Communication and Capital Exchange Accounts.*

⁷ As defined informally in [Miller90a], a concept *x* is said to be hyponym of concept *y* if *x* is more specific than *y* and the statement *An x is a kind of y* is true.

⁸ Hypernymy is the inverse relation of hyponymy

2.2.6 Function-value coreference

If a function-value relation exists between two phrases then they could be considered coreferential.

AMSC reported loss₁ for the 1st quarter is \$80.2 million₁, while the revenue₂ is \$30.3 million₂.

In this example "AMSC reported loss" and "\$80.2 million" refer to the same amount of money (the financial loss could be referred to as to a function where "80.2 million" is the value) so the two phrases are considered coreferential. The same holds for "the revenue" and "\$30.3 million".

2.2.7 Ordinal Anaphora and One-anaphora

Ordinal anaphora is observed when the anaphor is a cardinal number like *first*, *second*, etc or an adjective such as *former* or *latter*.

An example for ordinal anaphora could be:

John₁ and Gloria₂ talk a great deal, the latter₂ more than the former₁.

One-anaphora is the case where the anaphor is "one"-phrase.

We have two lectures₁ today. The afternoon one₁ is more interesting.

2.2.8 Part-whole coreference

As defined in [Bean99], part-whole coreference is the case when the anaphor specifies part(s) of a larger antecedent. An example from the same source is:

The meal₁ was a disaster because the main dish₁ was overdone

If a lexical resource such WordNet is employed, the resolution of this type of anaphora could be much easier, because of the meronymy⁹/holonymy¹⁰ relations available there.

⁹ Meronymy is the part-whole relation, defined informally in [Miller90a] as: a concept *x* is said to be meronym of a concept *y* if the statement *An x is a part of y* is true.

¹⁰ Holonymy is the inverse relation of meronymy

2.3 Special cases

2.3.1 Pleonastic *It*

Pleonastic pronoun (also known as *expletive pronoun*) is the case when a pronoun (usually "it") does not refer to any particular antecedent. The term *pleonasm* comes from Greek and means "redundant". Phrases that use more words than semantically necessary are considered pleonastic. Examples for "pleonastic it" are:

It's 2 a.m.

It is raining.

It is Monday.

"It's important that Clinton goes to China knowing that Tibet is an issue important to America"

Although pleonastic pronouns are not considered anaphoric (since they don't have an antecedent), identifying such occurrences is important so that the coreference resolution system will not try to look for their antecedents.

In [Denber98] "pleonastic it" occurrences are classified as:

- state references
- passive constructs

State references are usually used for assertions about the weather or the time, so this type of "pleonastic it" is further divided into *meteorological* references and *temporal* references.

Meteorological references usually follow one of the three patterns:

- <It *be* PCP> where PCP is the present progressive form of "meteorological verbs" like "snowing", "raining", "storming", etc., like in:

It is snowing

- <It *be* ADJ> where ADJ is a "meteorological adverb" such as "sunny", "windy", "cold", etc., like in:

It is cold outside

- <It *be* N> where N could be a season/month/weekday name.

Temporal anaphora is observed in expressions like:

It's too late.

It's 2 a.m.
It's time to go.

[Denber98] suggests three patterns that cover temporal anaphora:

- <It be ADV* TIME> where TIME is a time expression. This pattern will match the second example
- <It be PREP* time> - this pattern will match cases similar to the third example
- <It be ADV* (early | late)> which will match cases like the first example.

Passive constructs for "pleonastic it" are studied in detail in [Lappin94]. The authors identify several patterns that cover most of the pleonasm of this type. First, a class of *modal adjectives* that are commonly used in "pleonastic it" constructs is specified:

advisable, convenient, desirable, difficult, easy, economical, certain, good, important, legal, likely, necessary, possible, sufficient, useful

The set of modal adjectives is extended with the comparative and superlative forms of the above adjectives.

A class of *cognitive verbs* is identified including:

anticipate, assume, believe, expect, know, recommend, think

With the help of the classes defined above, a set of patterns is identified, covering most common occurrences of "pleonastic it" (pattern No:6 could be regarded as temporal anaphora that was already discussed):

1. It is **Modaladj**¹¹ that **S**
2. It is **Modaladj** (for **NP**) to **VP**
3. It is **Cogv-ed**¹² that **S**
4. It seems/appears/means/follows (that) **S**
5. **NP** makes/finds it **Modaladj** (for **NP**) to **VP**
6. It is time to **VP**
7. It is thanks to **NP** that **S**

To the best of our knowledge, the relative importance of the pleonastic pronouns (especially "pleonastic it") has not been studied in detail and there are just a few reports about the percentage of such occurrences in various corpora. Three of the studies of anaphora resolution - [Lappin94], [Denber98], [Mitkov01] - contain statistics about pleonastic pronouns, but there is some variation in the percentages reported.

The study of Lappin & Leass ([Lappin94]) reports 8% of pronouns being pleonastic in the training corpus. M. Denber reports in [Denber98] that "pleonastic it"

¹¹ one of the modal adjectives identified above or its comparative/superlative form

¹² the passive participle of a cognitive verb

constitutes 12.8% of the pronouns¹³ observed in the sample corpus. Finally, Mitkov & Barbu report¹⁴ in [Mitkov01] that in a corpus of 28,272 words the non-anaphoric pronouns (not just pleonastic *it*) are at average 14.2% (with the lowest value being 7.8% and the highest being 24%). The last report is perhaps the most representative because the corpus used is much bigger compared to the other two studies.

We have performed a similar study on parts of the ACE test corpus (almost 190,000 words) and the results, which will be reported in detail in the next chapter, are quite different from those shown above.

2.4 Popular algorithms & systems

Much research has been performed in the field of anaphora and coreference resolution and especially in the field of pronominal resolution. Works with significant importance include [Lappin94], [Boguraev96], [Baldwin96], [Kameyama97], [Mitkov98] and [Mitkov01]. The approaches differ in the set of anaphora that are processed, in the employment of syntax structure for the analysis and in employment of focusing and centering theory techniques.

Most often the algorithm for pronominal anaphora resolution consists of the steps:

- Identify the part of the text surrounding the pronoun that will be inspected for candidate antecedents
- Reject the candidates that fail to satisfy certain gender/number/sort/etc. consistency checks
- According to a predefined set of rules assign salience values to each candidate
- Choose the candidate ranked highest in the previous step

A brief overview of the most popular approaches to pronominal anaphora resolution follows.

2.4.1 Lappin & Leass

S. Lappin and H. Leass present in [Lappin94] a syntax-based approach for identifying the antecedent of 3rd person pronouns and lexical anaphors. The algorithm used syntax information for the text being processed and contains a set of factors assigning the salience value of each candidate antecedent. The implementation also contains a *pleonastic it* identification component.

The authors report 86% successfully identified antecedents in a corpus containing technical manuals.

A modification of the algorithm that does not employ deep syntactic parsing is proposed in [Boguraev96]. The authors report 75.5% success in resolution on a corpus containing texts of different genres. A comparative evaluation of the algorithm

¹³ the author reports 41 pronouns being anaphoric and 6 cases of "pleonastic *it*"

¹⁴ the authors report 60 out of 362 pronouns being non-anaphoric

of Kennedy and Boguraev performed in [Mitkov01] reports a much more balanced and reliable success rate of 61.6% for this algorithm.

2.4.2 Breck Baldwin's CogNIAC

Breck Baldwin presents in [Baldwin96] a simple yet effective algorithm for pronominal resolution. The algorithm does not rely on heavy syntax parsing but instead employs a set of 6 rules that assign proper salience values to candidate antecedents.

The authors report 92% precision and 64% recall on a corpus containing The Wall Street Journal articles. Again, the comparative study in [Mitkov01] reports much more realistic success rate of 50% and precision of 43%.

2.4.3 Ruslan Mitkov's knowledge-poor approach

Ruslan Mitkov reports in [Mitkov98] a knowledge poor approach to pronominal resolution. The algorithm does not employ syntactic information but relies on a set of indicators (rules) such as definiteness, heading, collocation, referential distance, term preference, etc. The indicators assign salience values to the antecedents. The author reports success rate of 89.7% on a corpus of technical manuals. The comparative study in [Mitkov01] reports success rate of 57% and precision of 49%

3 The General Architecture for Text Engineering (GATE)

3.1 GATE overview

GATE is the General Architecture for Text Engineering - a language-engineering environment developed by the University of Sheffield. Since its first release in 1996, GATE was used in a number of IE and other applications (see [Maynard00]).

In this work we use GATE for corpus analysis and as a development environment for the implementation of the coreference algorithm. The module developed is now distributed as part of the ANNIE IE system in GATE.

3.1.1 GATE - a General Architecture for Language Engineering

GATE provides a software infrastructure for NLP researchers, which is made up of three main elements:

- an *architecture* for describing the components composing a language processing system
- a *framework* (Java class library and API) that could be used as a basis for building such systems
- a graphical *development environment* (built on top the framework) comprising of a set of tools and components for language engineers

GATE aims to provide uniform access to various linguistic and ontological resources. Such resources are modeled in an object-oriented way (both in the framework and in the graphical environment) by a common abstraction and in this way the specific complexities related to the representation of each different resource are hidden from the language engineer. When using GATE everyday tasks such as storing and visualizing data as well as loading different processing resources that operate on the data become transparent to the language researcher.

3.1.2 Resources in GATE

GATE distinguishes the following three types of resources:

- *Language Resources* (LR) representing documents, lexicons, corpora, ontologies, etc.
- *Processing Resources* (PR) representing components that operate on language resources such as tokenisers, POS taggers, parsers, etc.
- *Visual Resources* (VR) representing GUI components that visualize and edit the language and processing resources

These three types of resources are used to model the components comprising a language processing system based on GATE. The set of processing resources that are integrated with the system is called CREOLE, the Collection of Reusable Objects for Language Engineering. Language engineers can develop their own resources and integrate them into the GATE system in a plug-and-play manner (details are available in the GATE User Manual [Cunningham01])

3.1.3 Applications and Datastores

Two other abstractions provided by GATE for the language researchers are *applications* and *datastores*. Applications are sets of processing resources grouped together and executed in sequence over some language resource. The two types of applications available at present are *Pipeline*, which is general for any sequence of processes, and *Corpus Pipeline*, which operates over a corpus. Datastores in GATE provide persistent representation for language resources. At present there are two types of datastore: *Serial Datastore*, storing language resources directly on the filesystem, and *Database Datastore* which uses an RDBMS as persistent storage.

3.1.4 Annotations

When processing resources (such as parsers, tokenisers, taggers, etc.) that are part of some NLP system operate on the original texts, they produce information about this text. Such information, for example, is the type of a specific token (word, number, punctuation, etc.) that is generated from the tokeniser, or the part of speech for the word (proper noun, pronoun, verb, etc.) that is generated from the POS tagger. The information about the text is usually represented as sets of *annotations*. As described in [Cunningham00c] there are two approaches to storing annotations.

The first approach is the *embedded markup* approach and it is based on embedding the annotation data in the original text, usually with the help of SGML, XML or other markup language. An example for embedded markup is the British National Corpus (BNC).

The other approach, accepted by GATE, is the *reference annotation* model. Examples of this type include the TIPSTER format and the LDC¹⁵'s ATLAS format. In recent years the two approaches have converged, with the SGML and XML markup communities starting to adopt "standoff" markup, where annotations are stored in a separate file with offsets referring back to the text. The reference model stores annotations in annotation graphs and this information is not embedded in the original text but instead the generated annotations refer to fragments of the text by reference.

A GATE annotation consists of:

¹⁵ Linguistic Data Consortium

- ID, which is unique in the document the annotation refers to
- type, which denotes the type of the annotation (different processing resources usually generate annotations of different types)
- start and end node, which denote the span of the annotation, i.e. the fragment of the original text the annotation refers to
- a set of features (attribute/value pairs) that provide additional information about the annotation

For example, for the following sentence:

Mr. Bush is the President of the United States.

...an annotation graph like the one on Figure 2 will be created, containing a Title annotation for the sequence of tokens "Mr" and ".", a Person annotation for the sequence "Mr", "." and "Bush", etc.

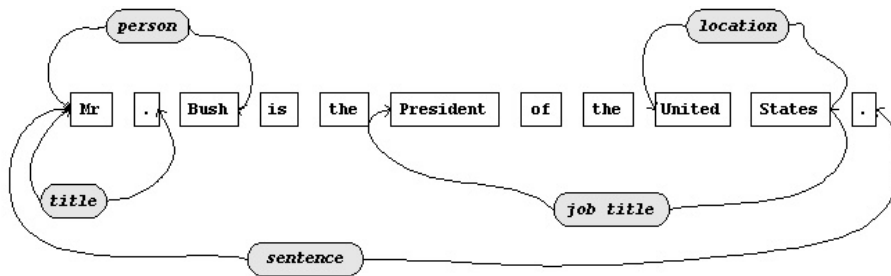


Figure 2. Simplified annotation graph

Note that this is not the full annotation graph, most of the annotations are actually skipped for simplicity. The complete annotation graph for the sentence contains more than 30 annotations, together with their features.

3.1.5 Built-in Processing Resources

GATE contains a set of built-in components called ANNIE (A Nearly-New IE system). The components that comprise ANNIE are:

- tokeniser
- gazetteer
- sentence splitter
- POS tagger
- named entity transducer
- orthographic name-matcher

The ANNIE components form an IE pipeline which looks like (figure 3):

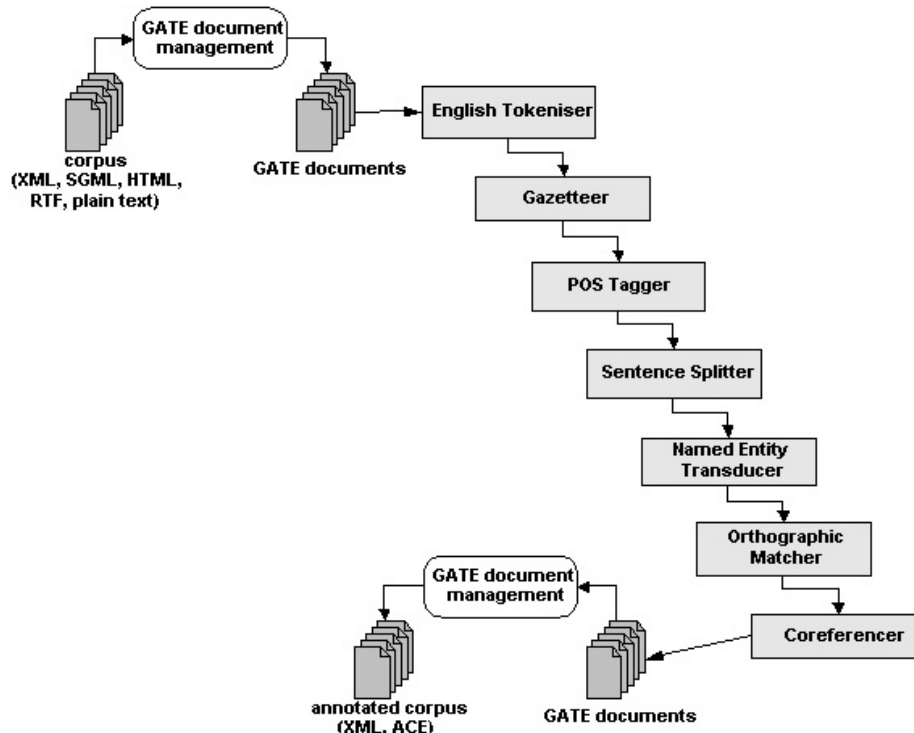


Figure 3. The ANNIE pipeline in GATE

The coreference resolution component that was developed has now been integrated with ANNIE.

The following sections present brief overviews for each ANNIE component.

3.1.6 English Tokeniser

The tokeniser analyses the input text and splits it into tokens for words, numbers, punctuation, symbols and white spaces. A word is a sequence of contiguous upper or lowercase letters, including a hyphen, a number is a sequence of digits, symbol tokens represent symbols such as '@', '#', '\$', etc., white spaces are space characters or new line marks. For the words, numbers, punctuation and symbols recognized in the text, the tokeniser creates an annotation of type "Token", while for the white spaces "SpaceToken" annotations are created.

3.1.7 Gazetteer

The task of the gazetteer is to create annotations that provide information about the entities such as persons, organizations, locations, money amounts and time expressions mentioned in the text. For its work the gazetteer uses lookup lists containing countries, cities, currencies, first person names, family names, etc. The current implementation of the gazetteer uses flat lists, with one entry per line for each known entity, but work is in progress to transform the gazetteer into a hierarchical one using taxonomies and ontologies to represent inheritance relations between the entities.

The gazetteer reads the lookup lists and produces a finite state machine (FSM) that generates annotations of type "Lookup" for the entries in the text that are matched by the FSMs. Each Lookup annotation contains information about the type of the entity (person, company, etc.) that was recognized by the finite state machine. Of course it is virtually impossible to create and maintain lists containing all possible mentions of persons, cities, companies, etc. and the process of entity name recognition is far from completed only with the gazetteer results. These results are used as a basis for the actual name recognition performed later by the Named Entity Transducer and the OrthoMatcher.

3.1.8 Sentence Splitter

The Sentence Splitter segments the input text into sentences. It produces Sentence annotations for each recognized sentence, and Split annotations for sentence breaks such as full-stop (or other punctuation) and line break.

3.1.9 POS Tagger

GATE uses the Brill-style POS tagger¹⁶ developed by Mark Hepple (University of Sheffield). The tagger produces the proper part-of-speech tag for each word or symbol. The part-of-speech tags are compliant with the Penn TreeBank¹⁷ tags.

3.1.10 Named Entity Transducer

The NE Transducer component in ANNIE is a JAPE¹⁸ transducer, which applies special rules to the annotations produced during the previous phases of the processing in order to generate new annotations. This is where the actual name recognition is performed.

An example for such rule could be *"If the sequence of tokens 'University', 'of' is followed by a singular proper noun token, then create a new Organization annotation*

¹⁶ More information about Hepple's POS tagger is available at

<http://www.dcs.shef.ac.uk/~hepple/papers/>

¹⁷ Penn TreeBank project homepage is <http://www.cis.upenn.edu/~treebank/>

¹⁸ The Java Annotation Patterns Engine (JAPE) is presented in more details in the next section

for the three tokens" (of course rules are not presented in natural language but according to the JAPE grammar). This sample rule will match the phrase "University of Sofia" as a name of an organization entity (and will create a corresponding Organization annotation) although the name of the Sofia University is not mentioned in the gazetteer lookup lists at all.

3.1.11 Orthographic Matcher

The final component of ANNIE is the OrthoMatcher. It is responsible for performing the orthographic coreference based by default on the Person, Location and Organization annotations generated by the NE Transducer.

As a result, different mentions of the same proper name are linked together in a coreference chain. For example if both "Bill Clinton" and "Clinton" are found in the text, then the two annotations will be marked as coreferring.

Another important task for the OrthoMatcher is setting appropriate type for the Unknown annotations if such information was deduced (Unknown annotations are these for which the NE transducer could not propose any proper type). For example if "Merill Lynch" is found in the text and the company name does not appear in the lookup lists of the Gazetteer and is not matched by the rules of the Named Entity transducer, then it may not be possible to properly choose the type between "Organization", "Person" and "Location". But if "Merill Lynch HSBC" appears in the text and is recognized as Organization, then the OrthoMatcher will first match the two occurrences and then will change the type of the first one from "Unknown" to the type of the second, which will be "Organization".

3.2 JAPE overview

JAPE is the Java Annotation Patterns Engine in GATE. It provides finite state transduction based on regular expressions over GATE annotations.

Each JAPE transducer is provided with a JAPE grammar and if the rules defined in the grammar are satisfied by the annotations generated so far in the system it will perform the action which is specified in the grammar for the rule.

A JAPE grammar consists of a set of *phases*, which are executed sequentially. A phase consists of:

- unique (for the grammar) name
- input specifier, which defines the annotation types that are considered as valid input for the phase
- options specifier, which modifies the behaviour of the JAPE engine when executing this phase
- one or more *macros* (optional)
- one or more *rules*

A rule consist of:

- unique name
- optional priority
- left-hand-side (LHS) specifying a regular expression to be matched against annotations
- right-hand-side (RHS) specifying the action to be performed if the rule is satisfied

Since the LHS of a rule is a regular expression, it can contain regular expression operators such as "*", "?", "|" or "+". The RHS may contain any valid block of Java statements, which makes it quite powerful and flexible.

A detailed overview of the JAPE engine and the BNF for JAPE grammars is available in [Cunningham00b] and [Cunningham01]. We will present only a simplified example.

A sample JAPE grammar is:

```
Phase: Name
Input: Token
Options: control = appelt

Rule:OrgUni
Priority: 25
// University of Sheffield
// University of New Mexico
(
    {Token.string == "University"}
    {Token.string == "of"}
    ( {Token.category == "NNP"} )+
) :orgName
-->
:orgName.Organization = {rule = "OrgUni"}
```

The sample grammar begins with definition of a single phase. The name of the phase is specified together with the annotations that the phase will accept as a valid input. In this case only Token annotations will be matched against the rules and all other annotations already generated in the system will be ignored.

Additionally, the options of the phase instruct the JAPE engine to run in "appelt" mode, which means that if several rules are satisfied by the input at some moment (i.e. their LHS are matched by the input), then only the longest matching rule will be applied (the one that matches the longest sequence of annotations from the input). If there are several rules that match a region of the input with the same length, then rule priority is considered and the rule with the highest priority will be applied. Another style of processing is "brill" and when the engine runs in this mode then all the rules that are satisfied are applied (and respectively the priorities assigned to the individual rules are ignored). The final style is "first" which instructs the JAPE engine to apply the first rule that matches the input region.

The "Rule01" rule has a priority set to 25 (priorities are useful only for "appelt" style of the phase).

The LHS of the rule (the block in brackets preceding the "→" symbol) contains a regular expression that will be matched against the input (Token) annotations. The rule says that the sequence of tokens "University", "of" and one or more proper noun tokens should be matched (this is because the "string" feature of Token annotations contain the actual word or symbol of the token and the "category" feature contains the part-of-speech of the token assigned by the POS tagger). The matched sequence of input annotations can later be referred to as "orgName".

Finally the statements in the RHS (the block in brackets following the "→" symbol) will be executed whenever the LHS is satisfied by the input. The RHS will create a new annotation of type Organization for the span of the matched annotations. The Organization annotation will have a single feature "rule" that will be set to "OrgDept".

If we apply the rule to the input sequence of Token annotations "University", "of" and "Sheffield" a new Organization annotation will be created for the span "University of Sheffield".

4 Implementation

4.1 The ACE corpus

The ACE test corpus was used for the development and the evaluation of the coreference resolution module.

As mentioned earlier, ACE corpora are divided in three different types, according to the source:

- *broadcast news programs* (BNEWS), generated with the help of automated speech recognition (ASR) systems. The news is from news programs of ABC News, CNN, VOA and PRI¹⁹. Contains 60,000+ words.
- *newspaper* (NPAPER), generated by optical-character recognition (OCR) processing of newspaper sources. The corpus contains articles mostly from "The Washington Post". Contains 61,000+ words
- *newswire* (NWIRE). Contains 66,000+ words

The fact that the corpora are generated from processing, which is not very precise itself, induces additional difficulties for ACE systems.

4.2 Requirements definition

The requirements for the coreference module are made after considering the existing functionality and the need for coreference functionality in addition to that provided already by OrthoMatcher. The functional requirements are heavily influenced by ACE, where the University of Sheffield participates with an improved version of the ANNIE system.

So the main requirements were defined as:

1. The module performs pronominal coreference resolution for named entities in the text
2. The module is developed as a CREOLE component, so that it can be easily integrated with GATE
3. The module has an open architecture, so that it is easily extensible with additional functionality later (e.g. nominal coreference or appositional coreference resolution)
4. The module is configurable, so that the user can specify parameters that influence the behaviour of the module. The configuration is specified in the standard way for CREOLE component in GATE (in the creole.xml file)

¹⁹ Public Radio International

5. The results (annotations or features) generated by the module are in a predefined format that other GATE modules will use
6. [optional] *Pleonastic it* identification is performed, in order to improve the precision

4.3 Initial set of rules

As already discussed, there are many approaches for coreference resolution. Existing research in the field has given us valuable ideas and directions that are used as a basis for our implementation.

The basic observations could be summarized as:

1. Intrasentential anaphora is observed more often than intersentential anaphora
2. Recency factor is very important for choosing the antecedent
3. Personal pronouns are the most often observed type of pronouns, possessive pronouns take the second place, while reflexive pronouns are quite rare.
4. The number of non-anaphoric pronouns is not very high, but still represents a sufficient percentage of all the pronouns
5. The statistics generated from different corpora may vary substantially
6. The scope where the antecedent is located is not very large

Comments for each observation follow:

1. *Intrasentential vs. intersentential anaphora.* Several publications report statistics about the relative importance of the two types of anaphora. While there is substantial difference in the statistics reported, a common trend is that intrasentential anaphora is observed more often, which means that the chances the antecedent is found in the same sentence as the anaphor are higher. An implication of this observation (which also relates to the sixth one) is that usually the proper antecedent could be resolved even without inspecting of the context preceding the sentence too much backwards.

The following table is based²⁰ on statistics reported in [Lappin94] and [Mitkov01]. The entry for [Lappin94] corresponds to the training corpus, while the entry for [Mitkov01] corresponds to the whole corpus:

Source	Pronouns	Intersentential cases	Intersentential cases (%)	Intrasentential cases	Intrasentential cases (%)
[Lappin94]	560	89	15.9%	471	84.1%
[Mitkov01]	362	121	33.4%	241	66.6%

Table 3. Relative importance of intrasentential and intersentential anaphora

²⁰ Only the statistics for [Lappin94] are copied directly from the paper. The ones for [Mitkov01] are calculated from the published number of anaphoric pronouns and the number of intrasentential anaphors.

While there is some variation in the reported percentages, the overall conclusion is that intrasentential anaphora is observed more often than intersentential ones.

2. *Recency factor.* Most of the implementations for anaphora resolution employ a recency factor. In [Lappin94] the recency factor is the one with highest weight among a set of factors that influence the choice of antecedent. The recency factor states that if there are two (or more) candidate antecedents for an anaphor and all of these candidates satisfy the consistency restrictions for the anaphor (i.e. they are qualified candidates) then the most recent one (the one closest to the anaphor) is chosen.

Statistics related to the recency factor are available in [Mitkov01]. The published results show that the average distance (in sentences) between the anaphor and the antecedent is 1.3 sentences, and the average distance in noun phrases is 4.3 NPs.

The recency factor is very important for the process of choosing the best antecedent from a set of candidates. Most often there is not a single candidate, so a decision should be made and the recency factor makes such choice easier.

3. *Pronoun types.* The distribution of different types of pronouns depends on the corpus (and the domain of the texts in it) but some patterns are observed that hold with slight variations for most kinds of texts. The pattern observed most often is that the relative importance of personal pronouns is very high, because they constitute the largest share of pronouns in texts. The next most often observed type of pronouns is the set of possessive pronouns and finally the reflexive pronouns. The latter usually constitute a very small portion of the pronouns.

One implication of this is that personal pronouns are more important for the anaphora resolution implementation than possessive and reflexive pronouns. An algorithm that performs well on personal pronouns is expected to have good overall performance.

The following table from [Mitkov01] presents the relative distribution of pronouns by type, observed in the evaluation corpus²¹:

²¹ The corpus consists of technical manuals: "Linux Access HOWTO" (ACC), Windows hHelp file (WIN), "Beowulf HOWTO" (BEO), "Linux CD-ROM HOWTO" (CDR)

Source	pronouns	pers.	pers.(%)	poss.	poss. (%)	refl.	refl.(%)
ACC	182	161	88.5%	18	9.9%	3	1.9%
WIN	51	40	78.4%	11	21.6%	0	0.0%
BEO	92	74	80.4%	18	19.6%	0	0.0%
CDR	97	85	87.6%	10	10.3%	2	2.4%
TOTAL	422	360	85.3%	57	13.5%	5	1.4%

Table 4. Relative distribution of personal, possessive and reflexive pronouns reported in [Mitkov01]

Note that there is some variation in the reported percentages, for example possessive pronouns range from 9.9% to 21.6% of the total pronouns, but still the base pattern holds.

4. *Non-anaphoric pronouns.* Non-anaphoric pronouns are important because if they are not properly identified the performance of the resolution algorithm will deteriorate. Most often exact measures of non-anaphoric pronouns are unavailable. For example [Lappin94] and [Denber98] contain statistics about *pleonastic It* cases (which is only subset of all non-anaphoric cases that may be observed in text), while [Mitkov01] reports percentage of all non-anaphoric pronouns.

The next table summarizes reports²² non-anaphoric/pleonastic occurrences in different reports:

Source	Non-anaphoric* / <i>pleonastic it</i> **
[Lappin94]	7.7% **
[Denber98]	12.8% **
[Mitkov01]	14.2% *

Table 5. Pleonastic It / Non-anaphoric pronouns statistics from different reports.

Note that the percentages from [Mitkov01] are not really comparable with the other two reported values, since the former contains statistics about all non-anaphoric pronouns, while the latter report only *pleonastic It* occurrences. Also the statistics from [Denber98] are unlikely to be representative, because the evaluation corpus was too small (41 pronouns / 6 pleonastic occurrences).

5. *Variation in results according to the corpus used.* Various corpora differ in the number of words, the domain of the texts, the types and the distribution of pronouns, types and distribution of named entities, the complexity of resolving

²² Only the statistics from [Mitkov01] are copied directly, the ones for [Lappin94] and [Denber98] are calculated from the reported number of pronouns and pleonastic-It occurrences.

certain constructs. An algorithm performing very well on a corpus of technical manuals may fail for a corpus of news articles or dialogs containing quoted speech. It is therefore important that the implementation considers specifics of the target texts upon which it is intended to operate.

6. *Context*. Although most of the anaphoric links are intrasentential, looking for antecedent candidates only in the anaphor sentence is unlikely to produce good results. The algorithm should also inspect the context preceding the anaphor sentence. Various implementations use different limits for the size of the context being inspected. An interesting experiment reported in [Gaizauskas96] shows that if the size of the context is increased then the precision decreases while the recall increases. This is easy to explain, since with the increase of the size of context chance that a candidate antecedent will be found increases (thus increasing the recall). But at the same time the possibility that several candidates will be found increases too, which induces a certain risk that the wrong antecedent will be chosen (thus decreasing the precision). From the report we can conclude that context size of three to four sentences is usually optimal.

4.4 Analysis of the ACE corpora

We performed certain analyses over the ACE test corpora, in order to have better understanding of the specifics related to each type of corpus.

First we made an analysis of the pronoun distribution in the texts, and later an analysis of *pleonastic it* occurrences was performed. No analysis of the relative percentage of intrasentential and intersentential anaphora was performed, as well as analysis for the average referential distance between the anaphor and its antecedent.

Not all pronouns were included in the analysis, only the following categories:

- *personal* - I, me, you, he, she, it, we, they, me, him, her, us, them
- *possessive adjectives* - my, your, her, his, its, our, their
- *possessive pronouns* - mine, yours, hers, his, its, ours, theirs
- *reflexive* - myself, yourself, herself, himself, itself, ourselves, yourselves, themselves, oneself

There were cases in which a pronoun can be classified into more than one category. For example "his" and "its" may be possessive pronoun or possessive adjective. This is not a problem, since the POS tagger will identify this and will assign the proper category for the pronoun ("PRP" for possessive pronouns, and "PRP\$" for possessive adjectives).

4.4.1 Total pronouns

The percent of words that are pronouns reported in [Mitkov01] is 1.5% (422 pronouns out of 28,272 words). The average ratio we observed was almost three times higher. This is probably due to the specific differences in the domain of the analyzed texts. The corpus in [Mitkov01] consists of technical manuals where specific grammatical constructs and language is being used. The ACE corpus we analyzed consists of news articles and interviews where the number of named entities and the pronouns used to refer to them is unsurprisingly much higher.

The percentage of pronouns is shown in the following table:

source	Words	Pronouns	Pronouns (% of words)
npaper	61319	2264	3.7%
bnews	60316	3392	5.6%
nwire	66331	2253	3.4%
TOTAL	187966	7909	4.2%

Table 6. Number of pronouns and number of words in the ACE corpus

It's worth pointing out that the NWIRE and NPAPER part of the ACE corpus contain similar percentage of pronouns, while the percentage of pronouns in BNEWS is much higher. This is due to the fact that BNEWS contains mostly quotes speech dialogs, where pronouns are used more often than the names of the entities.

4.4.2 Distribution of pronouns by type

The relative distribution of pronouns by type is similar to the one reported in [Mitkov01]. Again the most significant share is the one of the personal pronouns, followed by the possessive pronouns while the share of reflexive pronouns is insignificant.

source	pronouns	pers.	pers. %	poss.	poss. %	refl.	refl. %
npaper	2264	1593	70.4%	627	27.7%	42	1.9%
bnews	3392	2862	84.4%	491	14.5%	39	1.1%
nwire	2253	1629	72.3%	586	26.0%	38	1.7%
TOTAL	7909	6084	76.9%	1704	21.5%	119	1.5%

Table 7. Distribution of personal, possessive and reflexive pronouns in the ACE corpus

The similarity between NPAPER and NWIRE corpora is observed again. The percentages for BNEWS are quite different from the rest and are closer to the ones reported in [Mitkov01].

The following table shows the relative importance of the 10 most often observed pronouns in each corpus:

npaper		bnews		nwire	
pronoun	%	pronoun	%	pronoun	%
HE	18.3%	IT	18.9%	IT	18.9%
IT	16.8%	I	11.6%	HE	16.5%
HIS	12.0%	YOU	11.6%	HIS	11.0%
ITS	8.6%	HE	10.5%	I	8.2%
THEY	8.0%	THEY	10.1%	THEY	8.1%
I	6.5%	WE	9.4%	ITS	6.7%
WE	6.4%	HIS	6.1%	WE	6.7%
SHE	4.8%	ITS	3.1%	YOU	5.0%
HER\$	3.3%	SHE	2.6%	SHE	2.6%
THEM	2.7%	HER\$	2.0%	HER\$	2.2%

Table 8. Relative importance of the 10 most often observed pronouns in different parts of the ACE corpus. HER\$ is the possessive adjective for SHE, not the object personal pronoun for SHE.

There exists significant difference in the distribution of certain pronouns in different corpora. For example "I" and "you" and "we" which are expected to indicate quoted speech presence constitute around 13% and 19% of the pronouns in NPAPER and NWIRE respectively, while the percentage for BNEWS is almost twice as high - 32.6%.

Another fact of interest that is not shown in the table is the relative importance of possessive pronouns (*mine*, *yours*, etc.) in the text. There were only two such pronouns observed in the NPAPER corpus, constituting 0.1% of the pronouns, and there were no such pronouns in the BNEWS and NWIRE corpora. This implies that the pronominal resolution algorithm may effectively ignore such pronouns because their (un)successful handling will not influence the final performance to any significant degree.

The same may hold for reflexive pronouns. They constitute about 1.5% of the pronouns in the three corpora, so their effective resolution is unlikely to contribute sufficiently to good performance.

4.4.3 *Pleonastic It Statistics*

We analyzed the three corpora for *pleonastic it* constructs. A full analysis for all non-anaphoric pronouns was out of the scope of this work. The statistics results for pleonastic it are expected to contain some imprecision, since the texts were analyzed

only by one person. The percentage of *pleonastic it* occurrences we observed was very low compared to the percentages reported in [Lappin94] (7.7%) and [Denber98] (12.8%). Although the latter is unlikely to be representative, because of the very small corpus used for the evaluation, there exist a huge difference in our results and the ones in [Lappin94]. This is most likely a consequence of the different domain of the analyzed texts - technical manuals vs. news articles and interviews.

Table 9 contains the results from the *pleonastic it* analysis:

	pronouns	IT	pleon-It	pleon-It (% of pronouns)	pleon-It (% of IT)
npaper	2264	381	79	3.5%	20.7%
bnews	3392	642	105	3.1%	16.4%
nwire	2253	425	70	3.1%	16.5%
TOTAL	7909	1448	254	3.2%	17.5%

Table 9. *Pleonastic it* occurrences as nominal value, percentage of all pronouns, percentage of "it"

Note that the statistics for BNEWS and NWIRE are quite similar but they differ a lot from the ones for NPAPER. It is also worth pointing that *pleonastic it* constitutes a large percent of the total number of occurrences of "it" so if pleonastic pronouns are ignored in the implementation of the resolution algorithm, the final results for "it" are likely to be unsatisfactory. This is even more important if we consider that "it" constitutes about 19% of the pronouns in the three corpora.

4.5 Design of the coreference resolution module

The analysis of the ACE corpora helped us clarify and prioritize the requirements for the implementation of the module.

The coreference resolution module is required to be a GATE Processing Resource (PR) so that it can be easily plugged in the CREOLE framework and used by other modules. The coreference module has modular structure - it consists of a main module and a set of sub-modules. The main module takes care to initialize the sub-modules, to execute them in the specified order and finally to combine the results generated from the sub-modules and eventually to perform some post processing over the result.

This modular structure provides sufficient flexibility, so that the behaviour of the coreference module may be modified or tuned for specific tasks. Such specific tasks may require that the order in which sub-modules are executed may be changed (unless there are interdependencies between them). For certain tasks it may not be feasible to load and execute some modules at all if they are unlikely to contribute much for the final result. This is the case with technical manuals, which do not usually contain

quoted speech fragments, so the sub-module identifying such fragments in the text will not be useful.

The modular structure also makes it possible that new sub-modules be plugged in the main coreference modules when they become available. This is especially important for GATE because our intent is to extend the basic pronominal coreference functionality once certain lexical and ontological resources are integrated with the system (such integration is in progress at present).

4.6 Architecture

The main module consists of three sub-modules so far:

- pronominal resolution module
- quoted text module
- *pleonastic it* module

The last two modules are helper sub-modules for the pronominal one, because they do not perform anything related to coreference resolution except locate quoted fragments and *pleonastic it* occurrences in text and generate temporary annotations that are used by the pronominal sub-module (such temporary annotations are removed later).

The main coreference module can operate successfully only if all ANNIE modules were already executed. The module depends on the following annotations created from the respective ANNIE modules:

- Token (English Tokenizer)
- Sentence (Sentence Splitter)
- Split (Sentence Splitter)
- Location (NE Transducer, OrthoMatcher)
- Person (NE Transducer, OrthoMatcher)
- Organization (NE Transducer, OrthoMatcher)

For each pronoun (anaphor) the coreference module generates annotation of type "Coreference" containing two features:

- *antecedent offset* - this is the offset of the starting node for the annotation (entity) which was proposed as antecedent or *null* if no antecedent can be proposed
- *matches* - this is a list of annotation IDs that comprise the coreference chain this pair of anaphor/antecedent is part of.

The following diagram clarifies the structure of the coreference module and its interaction with other ANNIE modules:

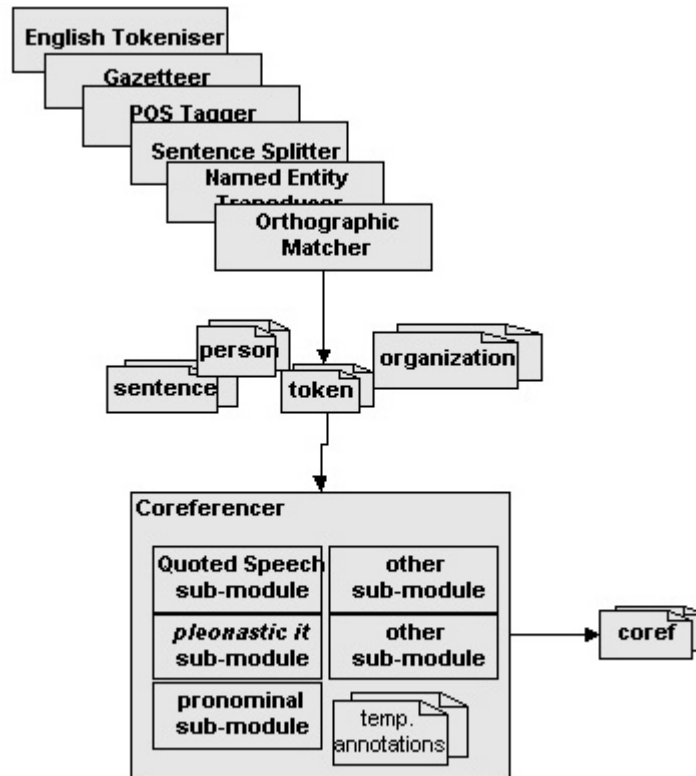


Figure 4. The coreferencer uses Sentence, Split, Token, Person, Organization and Location annotations generated from ANNIE modules and produces Coreference annotations

The next sections present overviews of each implemented module.

4.7 Quoted Speech Sub-module

The quoted speech sub-module identifies quoted fragments in the text being analyzed. The identified fragments are used by the pronominal coreference sub-module for the proper resolution of pronouns such as *I*, *me*, *my*, etc. that appear in quoted speech fragments. The module produces "Quoted Text" annotations.

The sub-module itself is a JAPE transducer that loads a JAPE grammar and builds an FSM over it. The FSM is intended to match the quoted fragments and generate appropriate annotations that will be used later by the pronominal module.

The JAPE grammar consists of only four rules, which create temporary annotations for all punctuation marks that may enclose quoted speech, such as " ' ; ` ", etc. and then try to identify fragments enclosed by such punctuation. Finally all temporary

annotations generated during the processing, except the ones of type "Quoted Text", are removed (because no other module will need them later).

The sub-module does not handle perfectly all the possible constructs of quoted fragments, which degrades the performance of the pronominal sub-module later. The main reason for that is the lack of correctly balanced quotation marks in the ACE corpora, especially the texts which were produced by OCR.

4.8 *Pleonastic It* sub-module

The *pleonastic it* sub-module matches pleonastic occurrences of "it". Similar to the quoted speech one, this module is a JAPE transducer operating with a grammar containing patterns that match the most commonly observed *pleonastic it* constructs.

As already explained, the number of *pleonastic it* occurrences observed was significantly less than the numbers reported by other researchers. Yet the relative share of *pleonastic it*, as a percentage of all the occurrences of *it* makes identification of the former useful.

Previous work, such as [Denber98] and [Lappin94], contains patterns about *pleonastic it*. The latter contains a set of rules for recognizing the pleonastic constructs most often observed in texts. These rules were already explained in detail in chapter 2.

Unfortunately we realized that these patterns would not be sufficient for all the cases observed. The problems we have identified for [Lappin94] are:

- often a synonym or antonym of a modal adjective or a synonym of a cognitive verb appears in the construct
- the patterns are not flexible enough and miss even small variations of the defined constructs
- it is unclear to which extent the patterns will deal with various syntactic variants of *be* that could be used
- there are certain constructs in the ACE corpus which will not be matched by the predefined patterns

The first problem could be resolved with the addition of the proper synonyms and antonyms from WordNet or another lexical resource. The other problems have to be resolved by extending the base patterns.

4.8.1 Extension of Modal Adjectives and Cognitive Verbs

With the help of WordNet and according to our observations of the ACE corpus, we extended the set of modal adjectives from [Lappin94] to the following set (original adjectives appear in bold):

acceptable, adequate, **advisable**, appropriate, bad, better, **certain**, clear, common, **convenient**, decent, **desirable**, **difficult**, doubtful, **easy**, **economical**,

efficient, enough, essential, expected, fair, feasible, **good**, great, hard, important, illegal, imperative, implausible, **important**, impossible, impractical, improbable, inadequate, inadvisable, inappropriate, inconvenient, inefficient, inessential, insufficient, invalid, **legal**, **likely**, mandatory, **necessary**, obligatory, painless, plausible, **possible**, practical, probable, rare, reasonable, recommended, safe, sensible, **sufficient**, suggested, suitable, sure, tough, typical, unacceptable, unadvisable, unclear, undesirable, unexpected, unfair, unimportant, unlikely, unnecessary, unreasonable, unsafe, unsuitable, unsure, unusual, unwise, unworthy, **useful**, useless, usual, valid, wise, wonderful, worthy, wrong

Of course such extension makes the *pleonastic it* module more computationally demanding and slows down the overall performance, so a balance should be made between extending the rules and processing with acceptable performance.

The cognitive verbs were extended to the set:

accept, advise, **anticipate**, **assume**, **believe**, consider, demand, deny, estimate, **expect**, foresee, intend, **know**, predict, promise, propose, realize, **recommend**, recognize, report, require, suggest, **think**

4.8.2 Extended Rules

We have extended the base patterns from [Lappin94] in order to be more appropriate for the pleonastic constructs observed in the ACE corpus. The extended patterns are:

1. It **be** (adverb01) modaladj (conj01) S
2. It **be** (adverb01) modaladj (for NP) to VP
3. It is (adverb01) cogv-ed that S
4. It (adverb01) verb01 (conj02 | to) S
5. NP verb02 it (adverb01) modaladj (conj01 NP) to VP

We have dropped patterns 6 and 7 from the original paper, because they constituted less than one percent of the observed *pleonastic it* occurrences.

In the patterns above we have:

be = {be, become, remain}

adverb01 = {highly, very, still, increasingly, certainly, absolutely, especially, entirely, simply, particularly, quite, also, yet, even, more, most, often, rarely}

modaladj is the set of modal adjectives already discussed

conj01 = {for, that, is, whether, when}

conj02 = {that, if, as, like}

cogv-ed is the passive participle of the cognitive verbs defined above

verb01 = {seem, appear, look, mean, happen, sound}

verb02 = {find, make, consider}

The implementation of the pattern extends the rules so that:

1. Different forms of the sets of verbs **be**, **verb01** and **verb02** are recognized (base, present 3rd person, present non-3rd person, past participle)
2. Question forms are matched
3. Modal verbs used with the above sets are matched
4. Negation is matched

We identified one more pattern that was observed often in the ACE corpus, but we did not implement it in the JAPE grammar, because the pattern was not generic enough and depends on too many specific expressions. The pattern looks like

6. It **be/take time-expr before/since S**

...where **time-expr** represents time expressions such as *two weeks*, *today*, *one month*, *a while*, *longer*, etc.

The following table lists the distribution of the pleonasms from each type observed in the ACE corpus together with the percentage of the occurrences correctly identified.

Pattern	occurrences	% of <i>pleonastic it</i>	identified
1	35	13.9%	72.0%
2	65	25.8%	72.0%
3	3	1.2%	33.3%
4	18	7.1%	77.8%
5	11	4.4%	72.7%
6	16	6.3%	-
unclassified	104	41.3%	-
total	252		37.7%

Table 10. Pleonastic-it statistics - number of pleonastic occurrences per pattern, percentage of the occurrences per pattern, percentage of occurrences identified

Note that patterns 1 and 2 are observed most often and the percentage of pleonastic it constructs that were not matched by any pattern is very high -more than 40%. The precision (number of occurrences matched / all occurrences of this type) of the specific rules is relatively good and with the exception of one rule it is more than 70% but the high number of unclassified occurrences degrade the overall performance.

4.9 Pronominal Resolution Sub-module

The main functionality of the coreference resolution module is in the pronominal resolution submodule. This submodule uses the result from the execution of the quoted speech and *pleonastic it* submodules.

The module works according to the following algorithm:

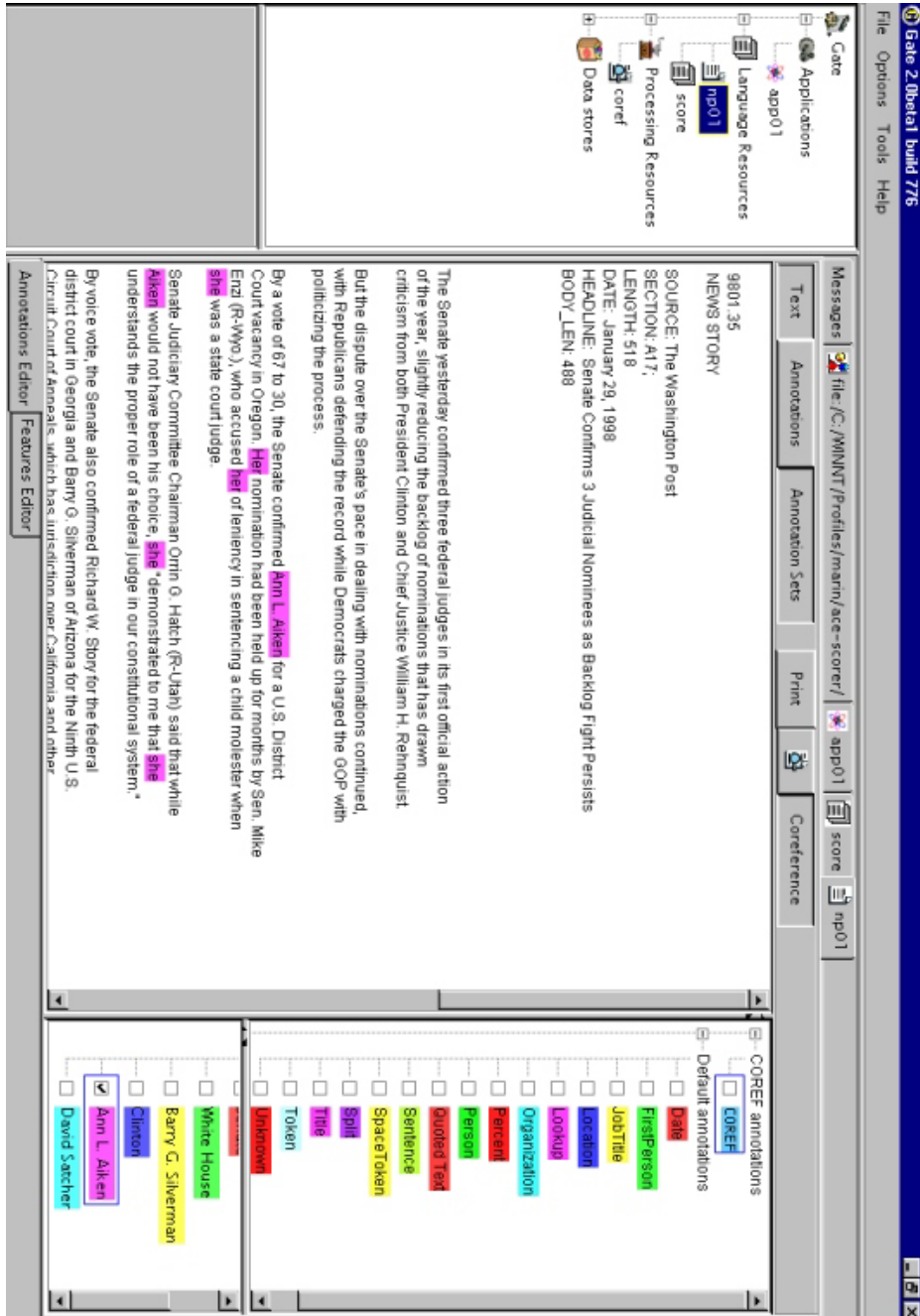
1. *Preprocess* the current document. This step locates the annotations that the submodule need (such as Sentence, Token, Person, etc.) and prepares the appropriate data structures for them.
2. *For each pronoun do* the following:
 - inspect the proper appropriate context for all candidate antecedents for this kind of pronoun
 - choose the best antecedent (if any)
3. Create the coreference chains from the individual anaphor/antecedent pairs and the coreference information supplied by the OrthoMatcher (this step is performed from the main coreference module)

The steps in more details:

1. Preprocessing. The preprocessing task includes the following subtasks:
 - Identifying the sentences in the document being processed. The sentences are identified with the help of the Sentence annotations generated from the Sentence Splitter.
 - For each sentence a data structure is prepared that contains three lists. The lists contain the annotations for the person/organization/location named entities appearing in the sentence. The named entities in the sentence are identified with the help of the Person, Location and Organization annotations that are already generated from the Named Entity Transducer and the OrthoMatcher.
 - The gender of each person in the sentence is identified and stored in a global data structure. It is possible that the gender information is missing for some entities - for example if only the person family name is observed then the Named Entity transducer will be unable to deduce the gender. In such cases the list with the matching entities generated by the OrthoMatcher is inspected and if some of the orthographic matches contains gender information it is assigned to the entity being processed
 - The identified *pleonastic it* occurrences are stored in a separate list. The "Pleonastic It" annotations generated from the pleonastic submodule are used for the task.
 - For each quoted text fragment, identified by the quoted text submodule, a special structure is created that contains the persons and the 3rd person singular pronouns such as "he" and "she" that appear in the sentence containing the quoted text, but *not* in the quoted text span (i.e. the ones preceding and succeeding the quote)
2. Pronoun resolution. This task includes the following subtasks:
 - retrieving all the pronouns in the document. Pronouns are represented as annotations of type "Token" with feature "category" having value "PRP" or "PRP\$". The former classifies possessive adjectives such as *my*, *your*, etc. and the latter classifies personal, reflexive etc. pronouns. The two types of pronouns are combined in one list and sorted according to their offset in the text.
 - For each pronoun in the list the following actions are performed:
 - if the pronoun is *it* then a check is performed if this is a pleonastic occurrence and if so then no further attempt for resolution is made.
 - the proper context is determined. The context size is expressed in the number of sentences it will contain. The context always includes the

- current sentence (the one containing the pronoun), the preceding sentence and zero or more preceding sentences.
- depending on the type of pronoun a set of candidate antecedents is proposed. The candidate set includes the named entities that are compatible with this pronoun. For example if the current pronoun is *she* then only the Person annotations with "gender" feature equal to "female" or "unknown" will be considered as candidates.
 - From all candidates one is chosen according to evaluation criteria specific for the pronoun.
3. Coreference chain generation. This step is actually performed by the main module. After executing each of the submodules on the current document, the coreference module follows the steps:
- retrieves the anaphor/antecedent pairs generated from them
 - for each pair, the orthographic matches (if any) of the antecedent entity is retrieved and then extended with the anaphor of the pair (i.e. the pronoun). The result is the coreference chain for the entity. The coreference chain contains the IDs of the annotations (entities) that co-refer.
 - a new Coreference annotation is created for each chain. The annotation contains a single feature "matches" which value is the coreference chain (the list with IDs). The annotations are exported in a pre-specified annotation set.

The result from processing a sample file with all ANNIE modules and with the coreference module is shown on the next picture. The coreference chain for the entity "Ann L. Alkien" is highlighted. Note that the coreference module only extends the coreference chain created by the OrthoMatcher with the pronouns for the entity. All the visualization components are already present as part of the GATE architecture and were not developed as part of this work.



Picture 2. Coreference chain for the named entity "Ann L. Alkien"

4.9.1 Resolution of *she, her, her\$²³, he, him, his, herself, himself*

The resolution for *she, her, her\$²³, he, him, his, herself* and *himself* is similar because the analysis of the corpus showed that these pronouns are related to their antecedents in similar manner. The characteristics of the resolution process are:

- context inspected is not very big - cases where the antecedent is found more than 3 sentences further back than the anaphor are rare
- recency factor is heavily used - the candidate antecedents that appear closer to the anaphor in the text are scored better
- anaphora has higher priority than cataphora. If there is an anaphoric candidate and a cataphoric one then the anaphoric one is preferred, even if the recency factor scores the cataphoric candidate better

The resolution process performs the following steps:

1. Inspect the context of the anaphor for candidate antecedents. A candidate is considered every Person annotation. Cases where *she/her* refers to inanimate entity (ship for example) are not handled.
2. For each candidate perform a gender compatibility check - only candidates having "gender" feature equal to "unknown" or compatible with the pronoun are considered for further evaluation.
3. Evaluate each candidate with the best candidate so far:
 - If the two candidates are anaphoric for the pronoun then choose the one that appears closer
 - The same holds for the case where the two candidates are cataphoric relative to the pronoun
 - If one is anaphoric and the other is cataphoric then choose the former, even if the latter appears closer to the pronoun

The statistics for resolving this set of pronouns and analysis of the problems will be presented in section 4.10.

4.9.2 Resolution of *it, its, itself*

This set of pronouns also shares many common characteristics. The resolution process contains certain differences with the one for the previous set of pronouns.

Successful resolution for *it, its, itself* is more difficult because of the following factors:

²³ This is the possessive adjective for *she*, not the object for of the personal pronoun

- there is no gender compatibility restriction. In the case when there are several candidates in the context, the gender compatibility restriction is very useful for rejecting some of the candidates. When no such restriction exists, and with the lack of any syntactic or ontological information about the entities in the context, the recency factor plays the major role for choosing the best antecedent.
- the number of nominal antecedents (i.e. entities that are referred not by name) is much higher compared to the number of such antecedents for *she*, *he*, etc. In this case trying to find antecedent only amongst named entities degrades the precision a lot.

We have performed analysis of the occurrences of *it*, *its* and *itself* in the ACE corpus in order to determine the usefulness of the recency factor as the only factor applied for choosing the best antecedent. Our analysis showed that:

- in 52% of the cases the most recent named entity of type Organization and Location was the correct antecedent
- in 15% of the cases the antecedent was a named entity which was not the most recent related to the anaphor
- in 33% of the cases the antecedent was nominal where the resolution will ultimately fail

The analysis shows that the recency factor all by itself offers some means of correct pronominal resolution. Further, we identified that half of the cases (7.3%) where the antecedent was not the most recent named entity were appositional. For example:

Yamaichi Securities Co₁, once *Japan₂*'s largest securities house, officially closed *its₁* last offices today after authorities revealed the severity of its losses.

In this example if the best antecedent is chosen on the basis of recency then *its* will be incorrectly matched to *Japan*. If apposition were partly identified, then the most proper choice would have been the named entity to which the apposition itself refers (in this case *Yamaichi Securities Co₁*).

The resolution steps are similar to the ones for *she*, *he*, etc. with the following differences:

- annotations of type Location and Organizations are considered candidate antecedents
- only recency is considered for choosing the best antecedent
- cataphora is penalized (i.e. named entities that are cataphoric to the pronoun are not considered as candidate antecedents)

The statistics for resolving this set of pronouns and analysis of the problems will be presented in section 4.10.

4.9.3 Resolution of *I, me, my, myself*

Resolution of these pronouns is dependent on the work of the quoted speech submodule.

One important difference from the resolution process of other pronouns is that the context is not measured in sentences but depends solely on the quote span. Another difference is that the context is not contiguous - the quoted fragment itself is excluded from the context, because it is unlikely that an antecedent for *I, me*, etc. appears there. The context itself consists of:

- the part of the sentence where the quoted fragment originates, that is not contained in the quote - i.e. the text prior to the quote
- the part of the sentence where the quoted fragment ends, that is not contained in the quote - i.e. the text following the quote
- the part of the sentence preceding the sentence where the quote originates, which is not included in other quote

For example the context for the following example is underlined:

Others believe things will move more slowly. "I don't expect to see a significant change as of April 1," said Mitsuru Saito, market economist for Sanwa Bank.

The underlined text in the next example is context for the *second* quoted fragment. The context for the first one will include the first part of the underlined text and parts (or the whole) preceding sentence (which is not shown).

Sen. John McCain said on CBS's Face the Nation, "I'm optimistic that we can get this done by this summer." Noting that the White House has complained, McCain said, "I think we may be well-positioned."

Another difference with resolution of pronouns of the first set (*he, she, his, him*, etc.) is that candidate antecedents are considered to be not only annotations of type Person but also the pronouns *he* and *she*. The latter are identified by annotations of type Token with features {category = "PRP", string = "he"} or {category = "PRP", string = "she"} respectively.

We identified several patterns that classify the relation between the pronouns *I, me, my, myself* and their proper antecedents. The subset of the corpus that was analyzed consisted of almost 40 documents containing 95 quoted fragments with 72 occurrences of the pronouns of interest. The patterns we identified for these 72 occurrences are:

- the antecedent is the closest named entity in the text following the quoted fragment. This pattern is observed in 52% of the cases. An example is:

"I₁ did not urge anyone to say anything that was untrue," Clinton₁ told Lehrer.

- the antecedent is found in the sentence preceding the sentence where the quoted fragment originates. If the preceding sentence also contains a quote then the antecedent is usually the named entity (or pronoun) which is the one most close to the end of the quote. This pattern was observed in 29% of the cases. An example is:

*"I₁ did not urge anyone to say anything that was untrue," Clinton₁ told Lehrer.
"That's my₁ statement to you"*

- the antecedent is the closest named entity preceding the quote in the sentence where the quote originates. This pattern counts for less than 3% of the cases. An example is:

U.S. officials said there was confusion about whether China would fulfill the contracts, but Cohen₁ declared: "I₁ believe we have assurances that such sales will not continue."

- the antecedent is either nominal (13%) or a named entity in position where the patterns above will not identify it correctly (3%). These cases will not be handled properly from the algorithm, which will induce degradation in recall and possibly degradation in precision (if the wrong antecedent is proposed).

It is worth noting that contrary to other pronouns, the antecedent for *I*, *me*, *my* and *myself* is most often cataphoric or if anaphoric it is not in the same sentence with the quoted fragment.

The resolution algorithm consists of the following steps:

1. Locate the quoted fragment description that contains the pronoun. If the pronoun is not contained in any fragment then return without proposing an antecedent.
2. Inspect the context for the quoted fragment (as defined above) for candidate antecedents. Candidates are considered annotations of type Pronoun or annotations of type Token with features {category = "PRP", string = "she"} or {category = "PRP", string = "he"}.
3. Try to locate a candidate in the text succeeding the quoted fragment (first pattern). If more than one candidate is present, choose the closest to the end of the quote. If a candidate is found then propose it as antecedent and exit.
4. Try to locate candidate in the text preceding the quoted fragment (third pattern). Choose the closes one to the beginning of the quote. If found then set as antecedent and exit.
5. Try to locate antecedents in the unquoted part of the sentence preceding the sentence where the quote starts (second pattern). Give preference to the one closest to the end of the quote (if any) in the preceding sentence or closest to the sentence beginning.

The statistics for resolving this set of pronouns and analysis of the problems will be presented in section 4.10.

4.9.4 Unresolved Pronouns

The pronouns our algorithm attempts to resolve count for only 68% of the pronouns in the ACE corpus. We tried to consider for resolution only the pronouns for which this is feasible i.e. the kinds that have relatively high share of the total pronouns in the corpus, and at the same time relatively simple heuristics for their resolution exist.

As only three out of the top-ten most often observed pronouns are not handled:

- *they* - counting for 9.0% of the pronoun occurrences in the corpus
- *we* - 7.8% of the pronouns
- *you* - 7.0% of the pronouns

Summary of the problems we encountered with each of the unresolved pronouns follows:

- *they, them* - most of the antecedents for these pronouns are nominal. The current implementation does not attempt to deal with nominal antecedents. When the WordNet integration and the hierarchical gazetteer are implemented, such an attempt for extending the pronominal submodule will be made. Report of good performance in the resolution of *they* and *them* achieved with simple heuristics is available in [Wooley88].
Often there is not a single antecedent for these pronouns, but instead a *split antecedents* case is observed, which requires more complex resolution.
- *you, your, we, us, our* - the antecedents are quite often nominals. In many cases these pronouns are *non-anaphoric* i.e. they do not have an antecedent in the discourse. Identifying such non-anaphoric cases is a complex task. *Split antecedents* cases are not observed that often.
- *other pronouns* - pronouns such as *their, yourself, ourselves, themselves, oneself, mine, yours, hers, ours* and *theirs* contribute for less than 9% of the pronouns observed in the ACE corpus. Their resolution also faces the difficulties related to nominal resolution, split antecedents handling and non-anaphoric pronoun identification and the comparatively small share they constitute makes their resolution relatively unfeasible.

4.10 Results and Error Analysis

We have manually annotated a small subset of the ACE copora in order to evaluate precision, recall and F-measure for the implementation. The subset consists of 21

randomly selected documents (7 from each corpus) containing 352 pronouns. The sample corpus represents 5% of the documents in the ACE corpus and contains 4.5% of the pronouns.

No pronouns were excluded from the evaluation. Occurrences of the pronouns that the implementation does not handle yet will degrade the recall. Nominal antecedents will degrade the precision.

The recall, precision and F-measure achieved on the evaluation corpus are:

precision = 63.1%
recall = 44.1%
F-measure = 51.9%

The following table contains the results for each individual group of pronouns:

Pronoun group	precision	recall	f-measure
1	79.3%	77.2%	78.2%
2	43.5%	51.7%	47.2%
3	47.6%	37.0%	41.7%

Table 11. Precision, recall and f-measure for the three groups of pronouns (1st group includes *he, she, etc.*, the 2nd group includes *it, its* and *itself*, the 3rd one includes *I, me, myself* and *my*)

The results show that the resolution of pronouns such as *he, she, her, etc.* is relatively successful even with such simple heuristic patterns used and without incorporating any syntax or centering information. The precision is degraded by the ratio of nominal antecedents. The algorithm will also benefit from some syntax information indicating the subject of the sentence, because the results show that the recency factor and the gender agreement are not sufficient.

The resolution of pronouns such as *it, itself* and *its* is less successful. Apart from the nominal antecedent which have even greater impact for this group, additional degradation is induced from the low performance of the *pleonastic it* module, which although using rules that cover much more cases than the ones in [Lappin94] will still identify only 38% of the pleonastic occurrences. It worth noting that the *pleonastic it* module has very high precision and very low recall, so a further extension of its patterns will improve the recall and will have positive impact on the resolution of *it*.

The very low recall for resolution of *I, me, etc.* is mostly caused by flaws of the quoted speech submodule, which fails to recognize certain quoted constructs. Additionally the performance is negatively impacted by the specifics of the BNEWS corpus, where the quoted fragments are not marked in the text, and as a result no attempt for resolution of the pronouns of the 3rd group will be made in this part of the ACE corpora. We were surprised by the low precision of the resolution, because nominal antecedents for pronouns of this group counted for only 13% of the cases. Additional analysis showed that the reason for the low precision is not in the resolution implementation but in the fact that named entity recognition modules of

ANNIE failed to recognize a certain percent of the entities that were antecedents for pronouns of this type.

As already mentioned the performance is partly degraded by mistakes made in ANNIE modules that were executed before the coreference module. This is unavoidable in any pipeline system where components rely on the quality of the results of the preceding components. It is important to mention that IE systems need substantial task specific tuning to achieve good results, and the ANNIE modules were used without such tuning for the ACE corpora²⁴.

The external mistakes that have negative impact on the coreference module are:

- Named entities not being recognized
- Sentences not being properly split

The first problem has impact on the resolution of all three groups of pronouns. The latter affects only the resolution of *I*, *me*, *my* and *myself* because the sentence where the quote originates and the one where it ends are crucial for choosing the context that will be expected. When the context is improperly defined then some named entities are improperly considered as candidates, which often leads to proposing the wrong antecedent.

We have identified the cases where the coreference module chooses the wrong antecedent only because a named entity is not recognized or because the sentences are not properly split. In an ideal case where such mistakes were not made, the improvements in precision, recall and F-measure would have been:

Precision: +10.8%
Recall: +8.8%
F-measure: +9.8%

The following table contains the improvements for each group of pronouns:

Pronoun group	precision	recall	f-measure
1	+9.0%	+9.5%	+9.3%
2	+3.7%	+5.9%	+4.7%
3	+39.4%	+39.9%	+39.9%

Table 12. Improvements in precision, recall and F-measure for each group of pronouns, in the imaginary case where all preceding modules perform perfectly.

The table shows that errors in named entity recognition have some negative impact on the performance. The impact is greater for pronouns in the first group (*she*, *he*, etc.)

²⁴ Modified versions of grammars of the NE recognition modules tuned for ACE are available, but they were not used.

because most often the antecedent is named entity. For pronouns in the second group the impact is less, because nominal antecedents are observed much more often.

In the case of pronouns like *I*, *me*, etc. the mistakes in the named entity recognition and sentence splitting severely degrade the performance.

5 Future Work

The lightweight approach we presented achieves acceptable performance without using any syntax structure information or centering theory methods, which shows that even simple heuristic rules identified from analysis of the text can be sufficient for simple coreference functionality.

Unfortunately any improvement in the precision and recall just by incorporating lightweight techniques is unlikely to be achieved. That is why we intend to incrementally extend the basic functionality presented with new features.

Two in-progress activities make it feasible to research and experiment with additional functionality. The first activity is representing WordNet as a Language Resource and providing the means to represent ontologies as Ontological Resources within GATE. The other activity is the Hierarchical Gazetteer being developed which will make it possible that certain ontological information be attached to the tokens in the text.

The main directions in which we intend to perform subsequent research and implementation are:

- *Apposition identification.* Appositional coreference has high importance in coreference resolution, because this kind of coreference is observed very often. If syntax information was available for the texts being processed then apposition could be identified relatively easy. Since ANNIE does not have syntax components we intend to analyze the corpora in order to find simple heuristic patterns that can identify an acceptable percentage of the appositional occurrences.
- *Extending the set of pronouns* being processed. We intend to perform further analysis in order to identify patterns that may help in resolving antecedents for pronouns that are still not handled and that are often observed in text.
- *Nominal coreference* based on synonymy/hyperonymy. A Language Resource such as WordNet makes it possible that certain candidate antecedents be identified on the basis of synonymy and hyperonym/hyponymy relations between the words.
- *Nominal coreference* based on world knowledge information. The ability to make ontologies available to the GATE system makes it possible that certain coreference relations that require world knowledge be identified.

Bibliography

- [ACE00] "Entity Detection and Tracking - Phase I", ACE Pilot study definition available from the ACE site (<http://www.itl.nist.gov/iaui/894.01/tests/ace/index.htm>)
- [Appelt99] D. Appelt and D. Israel: "Introduction to Information Extraction Technology". Tutorial, IJCAI-99, 1999.
- [Bagga98] Amit Bagga: "Evaluation of Coreferences and Coreference Resolution Systems". Proceedings of the First Language Resource and Evaluation Conference, May 1998
- [Baldwin96] Breck Baldwin: "CogNIAC : A High Precision Pronoun Resolution Engine". Working paper, University of Pennsylvania, 1996
- [Baldwin97] Breck Baldwin: "CogNIAC: high precision coreference with limited knowledge and linguistic resources". ACL'97/EACL'97 workshop on Operational factors in practical, robust anaphora resolution, Madrid, 1997.
- [Bean99] David Bean: "A Model for Automated Anaphora Resolution". PhD proposal, University of Utah, May 1999.
- [Beckwith93] R. Beckwith, G. A. Miller, and R. Teng, "Design and Implementation of the WordNet Lexical Database and Searching Software", Working Paper, Princeton University, 1993.
- [Boguraev96] C. Kennedy and B. Boguraev: "Anaphora for everyone: Pronominal anaphora resolution without a parser", Proceedings of the 16th International Conference on Computational Linguistics, 1996.
- [Byron01] Donna Byron: "The Uncommon Denominator: A Proposal for Consistent Reporting of Pronoun Resolution Results", Computational Linguistics - Special Issue on Computational Anaphora Resolution, Volume 27/Number 4, 2001.
- [Chinchor98] Nancy Chinchor: "Overview of MUC-7", Proceedings of the Seventh Message Understanding Conference (MUC-7), April 1998.
- [Cowie96] J. Cowie and W. Lehnert: "Information Extraction", Communications of the ACM, January 1996.
- [Cunningham99] Hamish Cunningham: "Information Extraction - a User Guide". Research memo CS-99-07, University of Sheffield, April 1999.
- [Cunningham00a] H. Cunningham, K. Bontcheva, W. Peters, Y. Wilks: "Uniform language resource access and distribution in context of a General Architecture for Text Engineering

- (*GATE*)". Proceedings of the Workshop of Ontologies and Lexical Resources (OntoLex), Bulgaria, 2000.
- [Cunningham00b] H. Cunningham, D. Maynard, V. Tablan: "*JAPE - a Java Annotation Patterns Engine*". Research memo CS-00-10, University of Sheffield, November 2000.
- [Cunningham00c] Hamish Cunningham: "*Software Architecture for Language Engineering*". PhD thesis, University of Sheffield, June 2000.
- [Cunningham01] H. Cunningham, D. Maynard, V. Tablan, C. Ursu and K. Bontcheva: "*Developing Language Processing Components with GATE*". GATE v2.0 User Guide, University of Sheffield, 2001.
- [Denber98] Michel Denber, "*Automatic resolution of anaphora in english*", Technical report, Eastman Kodak Co, Imaging Science Division.
- [Doddington01] George Doddington, "*Value-based Evaluation of EDT*", Technical report on the ACE 6-month meeting, May 2001.
- [Gaizauskas96] R. Gaizauskas and K. Humphreys: "*Quantitative Evaluation of Coreference Algorithms in an Information Extraction System*". Corpus-based and Computational Approaches to Discourse Anaphora, 1996.
- [Gaizauskas97] R. Gaizauskas and K. Humphreys: "*Using a semantic network for information extraction*". Journal of Natural Language Engineering, 3(2/3):147-169, 1997.
- [Grishman97] Ralph Grishman: "*Information Extraction: Techniques and Challenges*". Springer-Verlag, Lecture Notes in Artificial Intelligence, Rome, 1997.
- [Hepple00] Mark Hepple: "*Independence and commitment: Assumptions for rapid training and execution of rule-based POS taggers*". Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL-2000), Hong Kong, October 2000.
- [Hirschman97] Lynette Hirschman: "*MUC-7 Coreference Task Definition (v3.0)*". Message Understanding Conference Proceedings, July 1997.
- [Kameyama97] Megumi Kamyama: "*Recognizing referential links: An information extraction perspective*". Proceedings of ACL/EACL-97 Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts, 1997.
- [Lappin94] S. Lappin and H. Leass, "*An algorithm for pronominal anaphora resolution*", Computational Linguistics, 20(4), 1994.
- [Lyman00] P. Lyman, H. Varian, J. Dunn, A. Strygin, K. Swearigen, *How much information?* project. University of California at Berkeley, 2000.
- [Marsh98] E. Marsh, D. Perzanowski, "*MUC-7 Evaluation of IE Technology: Overview of Results*", Proceedings of the Seventh Message Understanding Conference (MUC-7), April 1998.
- [Maynard00] D. Maynard, H. Cunningham, K. Bontcheva, R. Catizone, G. Demetriou, R. Gaizauskas, O. Hamza, M. Hepple, P. Herring, B. Mitchell, M. Oakes, W. Peters, A. Setzer,

- M. Stevenson, V. Tablan, C. Ursu, and Y. Wilks. *"A Survey of Uses of GATE"*, Technical Report CS-00-06, Department of Computer Science, University of Sheffield, 2000.
- [Miller90a] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller, *"Introduction to WordNet: An on-line lexical database,"* International Journal of Lexicography, vol. 3(4), pp. 235--244, 1990.
- [Miller90b] George A. Miller: *"Nouns in WordNet: A Lexical Inheritance System"*, International Journal of Lexicography 3(4), 1990
- [Mitkov98] Ruslan Mitkov: *"Robust Anaphora Resolution with Limited Knowledge "*. In Proceedings of COLING'98/ACL'98, 1998.
- [Mitkov99] Ruslan Mitkov: *"Anaphora Resolution: The State of the Art"*. Working paper, University of Wolverhampton, 1999.
- [Mitkov01] C. Barbu and R. Mitkov: *"Evaluation tool for rule-based anaphora resolution methods"*. Proceedings of ACL'01, Toulouse, 2001.
- [Sundheim95] Beth Sundheim, *"Overview of Results of the MUC-6 Evaluation"*. Proceedings of the Sixth Message Understanding Conference (MUC-6), Columbia, 1995.
- [Wooley88] Bruce Wooley, *"Pronoun resolution of 'they' and 'them'"*. Proceedings of FLAIRS'98, Florida, May 1988.

Appendixes

Appendix A - list of all acronyms

ACE - Automatic Content Extraction
ASR - Automatic Speech Recognition
BNF - Backus Naur Form
ANNIE - A Nearly-New Information Extraction system
API - Application Programming Interface
BNC - British National Corpus
CO - Coreference resolution task (MUC)
CREOLE - Collection of Reusable Objects for Language Engineering
GATE - General Architecture for Text Engineering
IE - Information Extraction
JAPE - Java Annotation Patterns Engine
LR - Language Resource
MUC - Message Understanding Conference
NE - Named Entity
NP - Noun Phrase
OCR - Optical Character Recognition
POS - Part-Of-Speech
PR - Processing Resource
SGML - Simple Generalized Markup Language
ST - Scenario Template task (MUC)
TE - Template Element construction task (MUC)
TR - Template Relation task (MUC)
VR - Visual Resource
XML - eXtesible Markup Language

Appendix B - source code

The source code is part of the GATE 2.0 distribution and could be obtained from <http://www.gate.ac.uk>