Knowledge-Lean Coreference Resolution and its Relation to Textual Cohesion and Coherence

Sanda M. Harabagiu
Southern Methodist University
Dallas, TX 75275-0122
sanda@seas.smu.edu

Steven J. Maiorano
AAT
Washington, D.C. 20505
stevejm@ucia.gov

Abstract

In this paper we present a new empirical method for coreference resolution, implemented in the COCKTAIL system. The results of COCKTAIL are used for lightweight abduction of cohesion and coherence structures. We show that referential cohesion can be integrated with lexical cohesion to produce pragmatic knowledge. Upon this knowledge coherence abduction takes place.

1 Motivation

Coreference evaluation was introduced as a new domain-independent task at the 6th Message Understanding Conference (MUC-6) in 1995. The task focused on a *subset* of coreference, namely the *identity* coreference, established between nouns, pronouns and noun phrases (including proper names) that refer to the same entity. In defining the coreference task (cf. (Hirschman and Chinchor, 1997)) special care was taken to use the coreference output not only for supporting Information Extraction(IE), the central task of the MUCs, but also to create means for research on coreference and discourse phenomena independent of IE.

Annotated corpora were made available, using SGML tagging within the text stream. The annotated texts served as training examples for a variety of coreference resolution methods, that had to focus not only on precision and recall, but also on robustness. Two general classes of approaches were distinguished. The first class is characterized by adaptations of previously known reference algorithms (e.g. (Lappin and Leass, 1994), (Brennan et al., 1987)) to the scarce syntactic and semantic knowledge available in an IE system (e.g. (Kameyama, 1997)). The second class is based on statistical and machine learning techniques that rely on the tagged corpora to extract features of the coreferential relations (e.g. (Aone and Bennett, 1994) (Kehler, 1997)).

In the past two MUC competitions, the high scoring systems achieved a recall in the high 50's to low 60's and a precision in the low 70's (cf. (Hirschman et al., 1998)). A study[1] of the contribution of each form of coreference to the overall performance shows that generally, proper name anaphora resolution have the highest precision (69%), followed by pronominal reference (62%). The worse precision is obtained by the resolution of definite nominals anaphors (46%). However, these results need to be contrasted with the distribution of coreferential links on the tagged corpora. The majority of coreference links (38.42%) connect names of people, organizations or locations. In addition, 19.68% of the tagged coreference links are accounted by appositives. Only 16.35% of the tagged coreferences are pronominal. Nominal anaphors account for 25.55% of the coreference links, and their resolution is generally poorly represented in IE systems.

Due to the distribution of coreference links in newswire texts, a coreference module that is merely capable of handling recognition of appositives with high precision and incorporates rules of name alias identification can achieve a baseline coreference precision up to 58.1%, without sophisticated syntactic or discourse information. Precision increase is obtained by extending high-performance pronoun resolution methods (e.g. (Lappin and Leass, 1994)) to nominal coreference as well. Such enhancements rely on semantic and discourse knowledge.

In this paper we describe COCKTAIL, a high-performance coreference resolution system that operates on a mixture of heuristics that combine semantic and discourse information. The resulting

[1]The study, reported in (Kameyama, 1997), was performed on the coreference module of SRI's FASTUS (Appelt et al., 1993), an IE system representative of today's IE technology.

coreference chains are shown to contribute in the derivation of cohesive chains and coherence graphs. Both cohesive and coherence structures are considered, partly because of their incremental complexity and partly because the tradition (started with (Hobbs, 1979)) of studying the interaction of coreference and coherence. Section 2 presents COCKTAIL and the coreference methods it built upon. Sections 3 and 4 describe the derivation the cohesion and coherence structures.

2 Coreference Resolution

Coreference resolution relies on a combination of linguistic and cognitive aspects of language. Linguistic constraints are provided mostly by the syntactic modeling of language, whereas computational models of discourse bring forward the cognitive assumptions of anaphora resolution. Three different methods of combining anaphoric constraints are known to date. The first one integrates anaphora resolution in computational models of discourse interpretation. Dynamic properties of discourse, especially focusing and centering are invoked as the primary basis for identifying antecedents. Such computational methods were presented in (Grosz et al., 1995) and (Webber, 1988).

A second category of approaches combines a variety of syntactic, semantic and discourse factors as a multi-dimensional metric for ranking antecedent candidates. Anaphora resolution is determined by a composite of several distinct scoring procedures, each of which scores the prominence of the candidate with respect to a specific type of information. The systems described in (Asher and Wada, 1988) (Carbonell and Brown, 1988) and (Rich and Luperfoy, 1988) are examples of the mixed evaluation strategy.

Alternatively, other discourse-based methods consider coreference resolution a by-product of the recognition of coherence relations between sentences. Such methods were presented in (Hobbs et al., 1993) and (Wilensky, 1978). Although AI-complete, this approach has the appeal that it resolves the most complicated cases of coreference, uncovered by syntactic or semantic cues. We have revisited these methods by setting the relation between coreference and coherence on empirical grounds.

2.1 Pronominal Coreference

Two tendencies characterize current pronominal coreference algorithms. The first one makes use of the advances in the parsing technology or on the availability of large parsed corpora (e.g. Treebank (Marcus et al.1993)) to produce algorithms inspired by Hobbs' baseline method (Hobbs, 1978). For example, the *Resolution of Anaphora Procedure* (RAP) introduced in (Lappin and Leass, 1994) combines syntactic information with agreement and salience constraints. Recently, a probabilistic approach to pronominal coreference resolution was also devised (Ge et al., 1998), using the parsed data available from Treebank. The knowledge-based method of Lappin and Leass produces better results. Nevertheless, RAPSTAT, a version of RAP obtained by using statistically measured preference patterns for the antecedents, produced a slight enhancement of performance over RAP.

Other pronominal resolution approaches promote knowledge-poor methods (Mitkov, 1998), either by using an ordered set of general heuristics or by combining scores assigned to candidate antecedents. The CogNIAC algorithm (Baldwin, 1997) uses six heuristic rules to resolve coreference, whereas the algorithm presented in (Mitkov, 1998) is based on a limited set of preferences (e.g. definitiveness, lexical reiteration or immediate reference). Both these algorithm rely only on part-of-speech tagging of texts and on patterns for NP identification. Their performance (close to 90% for certain types of pronouns) indicates that full syntactic knowledge is not required by certain forms of pronominal coreference.

The same claim is made in (Kennedy and Boguraev, 1996) and (Kameyama, 1997), where algorithms approximating RAP for poorer syntactic input obtain precision of 75% and 71%, respectively, a surprising small precision decay from RAP's 86%. These results prompted us to devise COCKTAIL, a coreference resolution system, as a mixture of heuristics performing on the various syntactic, semantic and discourse cues. COCKTAIL is a composite of heuristics learned from the tagged corpora, which has the following novel characteristics:

1. COCKTAIL covers both nominal and pronoun coreference, but distinct sets of heuristics operate for different forms of anaphors. We have devised separate heuristics for reflexive, possessive, relative, 3rd person and 1st person pronouns. Similarly, definite nominals are treated differently than bare or indefinite nominals.
2. COCKTAIL performs semantic checks between antecedents and anaphors. These checks combine sortal constraints from WordNet with co-occurance information from (a) Treebank and (b) conceptual glosses of WordNet.
3. In COCKTAIL antecedents are sought not only in the accessible text region, but we also throughout the current coreference chains. In this way cohesive information, represented in coreference chains, is employed in the resolution process.
4. The heuristics of COCKTAIL allow for lexicalizations (e.g. when the anaphor is an adjunct of a communication verbs) and of simplified coherence cues (e.g.

when the anaphor is the subject of verb add, the antecedent may be a preceding subject of a communication verb).

To exemplify some COCKTAIL heuristics that resolve pronominal coreference, we first present heuristics applicable for reflexive pronoun and then we list heuristics for possessive pronouns and 3rd person pronoun resolution. Brevity imposes the omission of heuristics for other forms of pronoun resolution. COCKTAIL operates by successively applying the following heuristics to the pronoun *Pron*:

◇**if** *(Pron is reflexive)* **then** *apply successively:*
∘*Heuristic 1-Reflexive*(H1R)
Search for PN, the closest proper name from Pron in the same sentence, in right to left order.
if *(PN agrees in number and gender with Pron)*
  if *(PN belongs to coreference chain CC)*
   then *Pick the element from CC which is closest to Pron in Text.*
   else *Pick PN.*
∘*Heuristic 2-Reflexive*(H2R)
Search for a sequence Noun-Relative_Pronoun, in the same sentence, in right to left order.
if *(Noun agrees in number and gender with Pron)*
  if *(Noun belongs to coreference chain CC)*
   then *Pick the element from CC which is closest to Pron in Text.*
   else *Pick Noun.*
∘*Heuristic 3-Reflexive*(H3R)
Search for Pron', the closest pronoun from Pron in the same sentence, in right to left order.
if *(Pron' agrees in number and gender with Pron)*
  if *(Pron' belongs to coreference chain CC)*
   then *Pick the element from CC which is closest to Pron in Text.*
   else *Pick Pron'.*
∘*Heuristic 4-Reflexive*(H4R)
Search for Noun_c, the closest noun from Pron in the same sentence, in right to left order.
if *(Noun_c agrees in number and gender with Pron)*
   then *Pick Noun_c.*

Resolution examples for reflexive pronouns are illustrated in Table 1. The antecedents produced by COCKTAIL are boldfaced, whereas the referring expressions are emphasized. Both referring expressions and resolved antecedents and underlined. Precision results are listed in Table 2.

Antecedents of reflexive pronouns are always sought in the same sentence. Antecedents of other types of pronouns are sought in preceding sentences too, starting from the immediately preceding sentence. Inside the sentence, the search for a specific word is performed from the current position towards the beginning of the sentence, whereas in the preceding sentences, the search starts at the beginning of the sentence and proceeds in a left to right fashion. The same search order was used in (Kameyama, 1997). From now on, we indicate this search by $Search_1$. This search is employed by heuristics for possessive pronoun resolution:

Before Pennzoil's court fight with Texaco over the Getty purchase, **Mr. Liedtke** – one of the ploy's foremost practitioners – portrayed *himself* as something of an oil-patch rube, a notable feat considering his diplomas from Amherst College and Harvard Business School.
The woman who is known to me as hard-working and responsible, clearly isn't *herself.*
Unlike many of her peers, most of whom are males in their 30s, **she** never takes *herself* too seriously.

Table 1: Examples of reflexive pronouns

Heuristic	H1R	H2R	H3R	H4R
Precision on a test set of 100 randomly selected pronouns	95%	92%	98%	89%

Table 2: Coreference precision (reflexive pronouns)

◇**if** *(Pron is possessive)* (i.e. we have a sequence [*Pron* $noun_0$], where $noun_0$ is the head of the NP containing Pron) **then** *apply successively:*
∘*Heuristic 1-Possessive*(H1Pos)
$Search_1$ *for a possessive construct of the form* [$noun_1$'s $noun_2$],
  if ([*Pron* $noun_0$] and [$noun_1$'s $noun_2$] agree in gender, number and are semantically consistent)
   then if *($noun_2$ belongs to coreference chain CC)* **and** *there is an element from CC which is closest to Pron in Text, Pick that element.*
   Pick $noun_2$.
∘*Heuristic 2-Possessive*(H2Pos)
$Search_1$ *for PN, the closest proper name from Pron*
if *(PN agrees in number and gender with Pron)*
  if *(PN belongs to coreference chain CC)*
   then *Pick the element from CC which is closest to Pron in Text.*
   else *Pick PN.*
∘*Heuristic 3-Possessive*(H3Pos)
Search for Pron', the closest pronoun from Pron
if *(Pron' agrees in number and gender with Pron)*
  if *(Pron' belongs to coreference chain CC)*
   and *there is an element from CC which is closest to Pron in Text, Pick that element.*
   else *Pick Pron'*
∘*Heuristic 4-Possessive*(H4Pos)
Search for Noun, the closest common noun from Pron
if *(Noun agrees in number and gender with Pron)*

if *(Noun belongs to coreference chain CC)*
and *there is an element from CC which is closest to Pron in Text, Pick that element.*
else *Pick Noun*

Examples and precision results are listed in Table 3 and Table 4, respectively.

The timing of Mr. Shad's departure is likely to depend on how rapidly the Senate Banking Committee moves to confirm *his* successor.
Ronald Reagan sends him a list of *his* film roles.
The 20-minute flight helps him forget *his* troubles.
The president renewed *his* promise to veto "tax-rate increases."

Table 3: Examples of possessive pronouns

Heuristic	H1Pos	H2Pos	H3Pos	H4Pos
Precision on 100 random pronouns	96%	93%	78%	86%

Table 4: Coreference precision (possessive pronouns)

Given a possessive pronoun in a sequence [*Pron* $Noun_0$], the antecedent *Ante* of *Pron* is semantically consistent if the same possessive relationship can be established between *Ante* and $Noun_0$. the problem is that the possessive relation semantically corresponds to an open list of relations. For example, $Noun_0$ may be a feature of *Ante*, *Ante* may own $Noun_0$ or *Ante* may have performed the action lexicalized by the nominalization $Noun_0$.

COCKTAIL's test of semantic consistency blends together information available from WordNet and on statistics gathered from Treebank. Different consistency checks are modeled for each of the heuristics. We detail here the check that applies to heuristic H1Pos, that resolves the possessive from the first example listed in Table 3. For this heuristic, we have to test whether from the possessive [*Ante* $Noun_1$] we can grant the possessive [*Ante* $Noun_0$] as well. There are three cases that allow us to do so:

- *Case 1* $Noun_1$ and $Noun_0$ corefer.
- *Case 2* There is a sense s_1 of $Noun_1$ and a sense s_0 of $Noun_0$ such that a synonym of $Noun_1^{s_1}$ or of its immediate hypernym is found in the gloss of $Noun_0^{s_0}$ or viceversa.
- *Case 3* There is a sense s_1 of $Noun_1$ and a sense s_0 of $Noun_0$ such that a common concept is found in their glosses.

Cases 2 and 3 extend to synsets obtained through derivational morphology as well (e.g. nominalizations). For cases 2 and 3 COCKTAIL reinforces the coreference hypothesis by using a possessive-similarity metric based on Resnik's similarity measures for noun groups (Resnik, 1995). From a subset of Treebank, we collect all possessives, and measure whether the similarity class of $Noun_0$, $Noun_1$ and their eventual common concept is above a threshold produced off-line.

Other pronominal coreference heuristics employ $Search_2$, a search procedure that enhances $Search_1$, since it prefers antecedents that are immediately succeeded by relative pronouns. This search is incorporated in COCKTAIL's heuristics that resolve 3rd person pronominal coreference:

∘*Heuristic 1-Pronoun*(H1Pron)
$Search_2$ in the same sentence for the same *3rd person pronoun Pron'*
if *(Pron' belongs to coreference chain CC)*
and *there is an element from CC which is closest to Pron in Text, Pick that element.*
else *Pick Pron'.*

∘*Heuristic 2-Pronoun*(H2Pron)
$Search_2$ for PN, the closest proper name from Pron
if *(PN agrees in number and gender with Pron)*
if *(PN belongs to coreference chain CC)*
then *Pick the element from CC which is closest to Pron in Text.*
else *Pick PN.*

∘*Heuristic 3-Pronoun*(H3Pron)
if *Pron collocates with a communication verb*
then *$Search_1$ for pronoun Pron'=I*
if *(Pron' belongs to coreference chain CC)*
and *there is an element from CC which is closest to Pron in Text, Pick that element.*
else *Pick Pron'.*

∘*Heuristic 4-Pronoun*(H4Pron)
if *Pron collocates with a communication verb*
then *$Search_1$ communicator Noun*
if *(Noun belongs to coreference chain CC)*
and *there is an element from CC which is closest to Pron in Text, Pick that element.*
else *Pick Noun.*

∘*Heuristic 5-Pronoun*(H5Pron)
$Search_2$ for Pron', the closest pronoun from Pron
if *(Pron' agrees in number and gender with Pron)*
if *(Pron' belongs to coreference chain CC)*
and *there is an element from CC which is closest to Pron in Text, Pick that element.*
else *Pick Pron'*

∘*Heuristic 6-Pronoun*(H6Pron)
$Search_2$ for Noun, the closest noun from Pron
if *(Noun agrees in number and gender with Pron)*
if *(Noun belongs to coreference chain CC)*
and *there is an element from CC which is closest to Pron in Text, Pick that element.*
else *Pick Noun*

COCKTAIL doesn't employ semantic consistency checks for this form of pronominal coreference res-

olution. From our initial experiments, we do not see the need for special semantic consistency checks, since all heuristics performed with precision in excess of 90% Part of this is explained by our usage of pleonastic filters and of recognizers of idiomatic usage. Table 5 illustrates some of the successful coreference resolutions.

He says that in many years as a banker *he* has grown accustomed to "dealing with honest people 99% of the time.
Sen. Byrd takes pains to reassure the voter that *he* will see to it that the trade picture improves.
A nurse who deals with the new patient admits *she* isn't afraid of her temper.

Table 5: Examples of 3rd person pronouns

2.2 Nominal Coreference

Noun phrases can represent referring expressions in a variety of cases. For example, it is known that not all definite NPs are anaphoric. Conditions that define anaphoric NPs are still under research (cf. (Poesio and Vieira, 1998)). In the tagged corpora, we have found only 20.93% of the nominal coreference cases to be definites, the majority (78.85%) being bare nominals[2], and only 1.32% were indefinites. However, more than 50% of the nominal referring expressions were names of people, organizations or locations. Adding to this, 15.22% of nominal coreference links are accounted by appositives. Based on this evidence, COCKTAIL implements special rules for name alias identification and for robust recognition of appositions. Moreover, the heuristics for nominal coreference resolution apply $Search_3$, and enhancement of $Search_1$ that searches starting with the coreference chains, and then with the accessible text. To resolve nominal coreference, COCKTAIL successively applies the following heuristics:

∘*Heuristic 1-Nominal*(H1Nom)
if (*Noun* is the head of an appositive)
  then Pick the preceding NP.
∘*Heuristic 2-Nominal*(H2Nom)
if (*Noun belongs to an NP, $Search_3$ for NP'*
  such that Noun'=same_name(head(NP),head(NP'))
  or Noun'=same_name(adj(NP),adj(NP')))
  then if *(Noun' belongs to coreference chain CC)*
  then *Pick the element from CC which is*
    closest to Noun in Text.
  else *Pick Noun'.*
∘*Heuristic 3-Nominal*(H3Nom)
if *Noun is the head of an NP*
  then *$Search_3$ for proper name PN*
  such that head(PN)=Noun
  if *(PN belongs to coreference chain CC)*
  and *there is an element from CC which is*
    closest to Noun in Text, Pick that element.
  else *Pick PN.*
∘*Heuristic 4-Nominal*(H4Nom)
$Search_3$ for a proper name PN with the same
  category as Noun
  if *(PN belongs to coreference chain CC)*
  and *there is an element from CC which is*
    closest to Noun in Text, Pick that element.
  else *Pick PN.*
∘*Heuristic 5-Nominal*(H5Nom)
$Search_3$ Noun' a synonym or hyponym of Noun
  if *(Noun' belongs to coreference chain CC)*
  and *there is an element from CC which is*
    closest to Noun in Text, Pick that element.
  else *Pick Noun'.*
∘*Heuristic 6-Nominal*(H6Nom)
$Search_3$ for Noun either in definites or
  in NPs having adjuncts in coreference chain CC)
if *Ante semantically consistent with Noun*
  if *(Ante belongs to coreference chain CC)*
  and *there is an element from CC which is*
    closest to Noun in Text, Pick that element.
  else *Pick Ante.*
∘*Heuristic 7-Nominal*(H7Nom)
  if *(Noun or one of his hypernyms*
  or holonyms is a nominalization N)
  then *Search for the verb V deriving N*
  or one of its synonyms)
  then *Pick NP, the closest adjunct of V*
  if *(NP belongs to coreference chain CC)*
  and *there is an element from CC which is*
    closest to Noun in Text, Pick that element.
  else *Pick NP*
∘*Heuristic 8-Nominal*(H8Nom)
  if *(Noun is the head of a prepositional*
  phrase preceded by a nominalization N)
  then *Search for the verb V deriving N*
  or one of its synonyms)
  if *(Noun' is an adjunct of V)* and
  (Noun' and Noun have the same category
  if *(Noun' belongs to coreference chain CC)*
  and *there is an element from CC which is*
    closest to Noun in Text, Pick that element.
  else *Pick Noun'*
∘*Heuristic 9-Nominal*(H9Nom)
$Search_3$ for Noun', a metonymy whose
  coercion *is Noun*
  Pick Noun'

Some non-trivial examples are listed in Table 6. Heuristic H1Nom uses coreference cues indicated by appositions, whereas heuristic H2Nom promotes

[2]We count as bare nominals coreferring adjuncts as well.

IMB and Mr. York would;t discuss his compensation package which could easily reach into seven figures. *The subject* is sensitive at a time when IMB is laying off thousands of employees
Mr Iacocca led Chrysler through one of the largest stock sales ever for a U.S. industrial company, raising $1.78 billion. Chrysler is using most of *the proceeds* to reduce its $4.4. billion unfunded pension liability.
We read where the Clinton White House is seeking a deputy to chief of staff Mack McLarty to impose some disciplined coherence on *the place*'s rambunctious young staff.

Table 6: Examples of nominal coreference

the term repetition indicator, when consistency checks apply. For this heuristic, consistency checks are conservative, imposing that either the adjuncts be identical, coreferring or the adjunct of the referent be less specific than the antecedent. Specificity principles apply also to H5Nom, where *hyponymy* is promoted, similarly to (Poesio and Vieira, 1998). Heuristic H3Nom allows coreference between *"the Securities and Exchange Commission"* and *"the commission"* but it bans links between *"Reardon Steel Co."* and *"tons of steel"*.

Many times coreferring nominals share also semantic relations (e.g. *synonymy*). Heuristic H5Nom identifies such cases, by applying consistency checks. Based on experiments with the coreference module of FASTUS, where this heuristic was initially implemented, we require that most frequent senses of nouns be promoted. The same precedence of frequent senses is implemented in the assignment of categories, defined as the immediate WordNet *hypernym*. The category of proper names is dictated by the proper name recognizer, assigning such categories as *Person*, *Organization* or *Location*.

In this way, coreference between *"IBM"* and *"the wounded computer giant"* can be established, since sense 3 of noun *giant* is *Organization*, the category of *"IBM"*. Similar category-based semantic checks allow the recognition of the antecedent of *proceeds* from the second example listed in Table 6. The *hypernym* of *proceeds* is *gain*, whose gloss genus is *amount*, the category of *$1.78 billion*. Semantic checks are also required in H7Nom and H8Nom, heuristic that rely on derivational morphology. The first example from Table 6 is resolved by H7Nom, since *discussion* the nominalization of *discuss* has the category *communication*, a hypernym of *subject*. The antecedent is the object of the verb *discuss*.

The last heuristic, H9Nom identifies coreferring links with coerced entities of nominals. Coercions are obtained as paths of meronyms or hypernyms. (Harabagiu, 1998) discusses a coercion methodology based on WordNet and Treebank. Since in our test corpus there we very few cases of metonymic anaphors, Table 7 lists the precision of the other heuristics only.

Heuristic	H1Nom	H2Nom	H3Nom	H4Nom
Precision on	98%	95%	82%	88%
100 random	H5Nom	H6Nom	H7Nom	H8Nom
nominals	77%	82%	89%	63%

Table 7: Nominal coreference precision

The empirical methods employed in COCKTAIL are an alternative to the inductive approaches described in (Cardie and Wagstaff, 1999) and (McCarthy and Lehnert, 1995). Our results show that high-precision empirical techniques can be ported from pronominal coreference resolution to the more difficult problem of nominal coreference.

3 Lexical Cohesion

The heuristics encoded in COCKTAIL make light use of textual cohesion, i.e. the property of texts to "stick together"[3] by using related words. Both pronominal and nominal coherence resolution heuristics use cohesion cues indicated by term repetition while nominal coreference relies on semantic relations between anaphors and their antecedents. In addition, coreference chains are a form of textual cohesion, known as *referential cohesion* (cf. (Halliday and Hassan, 1976)).

Until now, *lexical cohesion*, arising from semantic connections between words, was successfully used as the only form of textual cohesive structure, known as *lexical chains*. At present there are three methods of generating lexical chains. The first one, implemented in the TextTiling algorithm (Hearst, 1997), counts the frequencies of term repetitions and is an ideal, lightweight tool for segmenting texts. The second method, adds knowledge from semantic dictionaries (e.g. *Roget's Thesaurus* in the work of (Morris and Hirst, 1991) or *WordNet* in the methods presented in (Barzilay and Elhadad, 1997), (Hirst and St-Onge, 1998)). Besides term repetition, this approach recognizes relations between text words that are connected in the dictionaries with predefined patterns. This method was applied for generation of text summaries, the recognition of the intentional structure of texts and in the detection of malapropism. The third method is based on a path-finding algorithm detailed in (Harabagiu and Moldovan, 1998). This method creates a richer

[3]Definition introduced in (Halliday and Hassan, 1976) and (Morris and Hirst, 1991)

structure, useful for the abduction of coherence relations from the knowledge encoded in WordNet.

Here we describe a new cohesion structure that (a) incorporates both lexical and referential cohesion and (b) produces a unique chain that contains not only single words, but also textual entities encompassing head-adjunct lists. We use the finite-state parses of FASTUS (Appelt et al., 1993) for recognizing these entities, but the method extends to any basic phrasal parser[4].

We produce this novel cohesive structure to exploit the close relation between text cohesion and coherence. It is known (cf. (Harabagiu, 1999)) that cohesion, as a surface indicator of the text coherence, can indicate the lexico-semantic knowledge upon which coherence is inferred. Our aim is to use this cohesive chain for producing axiomatic knowledge for CICERO, a TACITUS-like system that abducts coherence relations. TACITUS (Hobbs et al., 1993) is a successful abductive system when provided with extensive pragmatic and linguistic knowledge. CICERO is designed as a lightweight version of TACITUS, that performs reliable abductions with minimal knowledge and effective searches. Translating all the lexical, morphological, syntactic and semantic ambiguities from texts would make the search intractable. Out solution for CICERO is to use a cohesive chain to create manageable knowledge upon which the abduction can be performed. Section 4 describes this knowledge and the operation of CICERO.

Our cohesive chain is a linked structure consisting of three parts: (1) the connected *text entity*; (2) its incoming and outgoing *pointers* and (3) a *lexico-semantic graph*, containing paths of WordNet concepts and relations. The lexico-semantic structure is later translated in the axiomatic knowledge that supports coherence inference. To exemplify the cohesion chain, we use the following text, spanned by the coreference chains produced with COCKTAIL:

[Toys R Us]$_1$ named Michael Goldstein [chief executive officer]$_2$, ending years of speculations about who will succeed [Charles Lazarus]$_3$, [the [toy retailer]$_1$'s founder and chief architect.]$_3$
[Robert Nakasone]$_4$, [former vice chairman and widely regarded as the other serious contender for [the top executive]$_2$'s job]$_4$, was named president and chief operating officer, both new positions.

The indexes indicate the four coreference chains. This text has only two repeating terms, the verb *name* and the noun *executive*, thus it generates little information with the *TextTiling* algorithm. The cohesion method detailed in (Barzilay and Elhadad, 1997) can detect one lexical chain: [*chief executive officer, chairman, executive, president*]. We would like to obtain richer lexico-semantic information, thus we build a cohesion chain that contains larger textual entities. To recognize the entities, we use the coreference chains and the following parse, produced by FASTUS:

```
#<PHRASE(ORGANIZATION-NAME):"Toys R Us">
#<PHRASE(BASIC):"named">
#<PHRASE(PERSON-NAME):"Michael Goldstein">
#<PHRASE(NG):"chief executive officer">
#<PHRASE(COMMA):",">
#<PHRASE(GERUND):"ending">
#<PHRASE(NG):"years of speculation">
#<PHRASE(PREP):"about">
#<PHRASE(RELPRO):"who">
#<PHRASE(BASIC):"will succeed">
#<PHRASE(PERSON-NAME):"Charles Lazarus , the toy retailer 's
                      founder and chief architect">
-------------------------------------------------------------
#<PHRASE(PERSON-NAME):"Robert Nakasone,
formerly vice chairman">
#<PHRASE(CONJ):"and">
#<PHRASE(BASIC):"widely regarded">
#<PHRASE(PREP):"as">
#<PHRASE(NG):"the other serious contender">
#<PHRASE(PREP):"for">
#<PHRASE(NG):"the top executive 's job">
#<PHRASE(COMMA):",">
#<PHRASE(BASIC):"was named">
#<PHRASE(NG):"president and chief operating officer,
  both new positions">
-------------------------------------------------------------
```

Textual entities are either basic phrases contained in the coreference chains or lists of phrases collected from the parse, by scanning for all NGs or NAME-phrases directly connected to a verb phrase through a *Subject*, *Object* or prepositional relations. For example, as phrase *"Toys R Us"* is the antecedent from a coreference chain, its corresponding textual entity is:

"Toys R Us"–Subject→*"name"*
name←Object1–*"Michael Goldstein"*
name←Object2–*"chief executive officer"*

The cohesion chain for our text is illustrated in Figure 1. The algorithm that generates cohesion chains is:

Algorithm **Cohesion-Chain-Builder**

1. **if** *(current NG belongs to a coreference chain)*
Create its textual entity TE and place it on the chain
2. **if** *(the antecedent is already in the chain)*
Place the coreference pointer between the two TEs
3. **if** *(the coreference is not an appositive)*
Populate the lexico-semantic structure(TE)

The derivation of the lexico-semantic structure (LSS) follows the steps:

*1.***for every** *relation $r(w_1, w_2)$ from a TE*
if *(there is s_1 a sense of w_1 and s_2 a sense of w_2 such that the same relation $r'(w_3, w_4)$ is found in a gloss from the hierarchies of $w_1^{s_1}$ or $w_1^{s_1}$)*
Add relation r' to LSS
*2.***for every** *word w in a TE*

[4]Such a parser operates on part-of-speech tagged text, with several noun and verb grouping rules.

if *(there is a concept C in LSS such that there is a collocation [w c] in a gloss from the hierarchy(w))*
Add w to LSS

3. if *(word w is already in LSS)*
Add new connection to w in LSS

For example, in the first TE illustrated in Figure 1, we have the relation *Object(name,CEO)*. We find an *Object* relation also in the gloss of *appoint*, the hypernym of sense 3 of verb *name*. The new *Object* relation connect verb *assume* with the synset {*duty, responsibility, obligation*}. A hypernym of *CEO* is *manager*, collocating with *position* in the gloss of *managership*. Noun *position* belongs to the hierarchy of *duty*, thus the new *Object* relation can be added to the LSS.

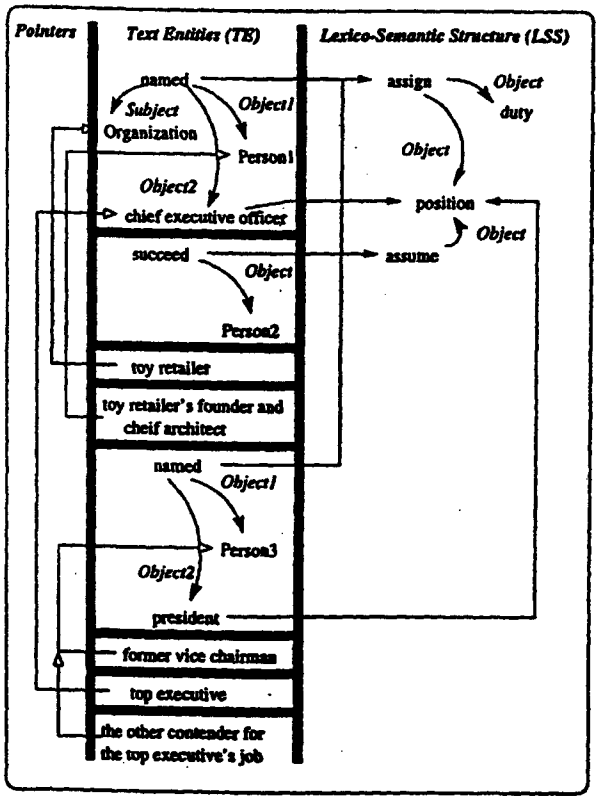

Figure 1: Cohesion chain

4 Text Coherence

We base our consideration of textual coherence on the definitions introduced in (Hobbs, 1985). The formal definition of relations that capture the coherence between textual assertions is based on the relations between the states they infer, their changes and their logical connections. States, changes and logical connections can be retrieved from pragmatic knowledge, accessible in lexical knowledge bases like WordNet. The complex structure of our cohesion chains help guiding these inferences.

For each textual unit, defined from the parse of the text, axiomatic knowledge produced. The acquisition of axiomatic knowledge is cued by the concepts and relations from the LSS portion of the cohesion chain, and is mined from WordNet. CICERO, our system, adds to this knowledge axioms that feature the characteristics of every coherence relation. CICERO's job is to abduct the coherence structure of a text. To do so, it follows the steps:

1. for every *textual unit* TU_i
2. *Derive pragmatic knowledge for* TU_i
3. for every *pair* $(TU_i, TU_j), i \neq j$
4. for every *coherence relation* $\mathcal{R}_k$
5. *hypothesize* $\mathcal{R}_k(TU_i, TU_j)$
6. *Perform abduction* $\mathcal{R}_k(TU_i, TU_j)$
7. *Choose cheapest abduction*

For the text illustrated in Section 3, this procedure generates the coherence graph illustrated in Figure 2.

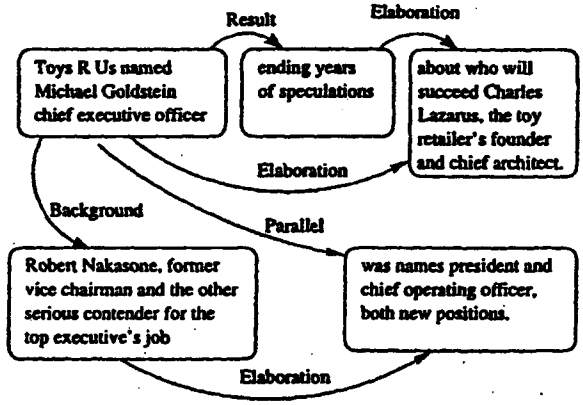

Figure 2: Coherence graph

We exemplify the operation of CICERO on this text by presenting the way it derives the *Elaboration* relation between the textual unit from the first sentence that announces the nomination of Michael Goldstein (TU_a) and the textual unit from the same sentence that deals with the succession of Charles Lazarus (TU_b) . First, CICERO generates the knowledge upon which the abductions can be performed. This knowledge is represented in axiomatic form, using the notation proposed in (Hobbs et al., 1993) and previously implemented in TACITUS. In this formalism each text unit represents an event or a state, thus has a special variable e associated with it. Events are lexicalized by verbs, which are maped into predicates *verb(e,x,y)* , where x represents the subject of the event, and y represents its object (in the case of intransitive verbs, y is not attached to a predicate,

whereas in the case of bitransitive verbs, y is mapped into y_1 and y_2). Moreover, predicates from the text are related to other predicates, derived from a knowledge base. These relations are captured in first order predicate calculus. For example, the pragmatic knowledge used for the derivation of the *Elaboration* relation between TU_a and $TU_b$is:

TU_a: $assign(e_1, x_1)\&positionx_1 \Rightarrow$ $name(e_1, x_2, x_3, x_1)\&org(x_1)\&person(x_2)$
$vacant\_position(e_1) \Rightarrow assign(e_2, x_1)\&positionx_1$
TU_b: $leave(e_1, x_1, x_2)\&person(x_1)\&position(x_2) \Rightarrow$ $vacant\_position(e_2)$
$leave(e_1, x_1, x_2)\&person(x_1)\&position(x_2)\&$ $assume(e_2, x_3, x_2)\&person(x_3) \Rightarrow$ $succeed(e_3, x_1, x_3)$

In the next step, all coherence relations are hypothesized, and the cost of their abduction is obtained. The appendix lists the *LISP* function created on the fly by CICERO that produces the abduction of the *Elaboration* function. Because of the computational expense, an intermediary step simplifies the axiomatic knowledge. The appendix lists also the full abduciton and its cost. CICERO is a system still under development, and at present we did not evaluate the precision of its results.

5 Conclusion

We have introduced a new empirical method for coreference resolution, implemented in the COCKTAIL system. The results of this algorithm are used to guide the abduction of coherence relations, as performed in our CICERO system. In an intermediary step, a rich cohesion structure is produced. This novel relation between coreference and coherence contrasts with the traditional view that coreference is a by-product of coherence resolution. Moreover, we reiterate the belief that coherence builds up from cohesion.

References

Chinatsu Aone and Scott W. Bennett. 1997. Evaluating automated and manual acquisition of anaphora resolution strategies. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (ACL-97)*, pages 122–129, Madrid, Spain.

Douglas E. Appelt, Jerry R. Hobbs, John Bear, David Israel, Megumi Kameyama and Mabry Tyson. 1993. The SRI MUC-5 JV-FASTUS Information Extraction System. In *Proceedings of the Fifth Message Understanding Conference (MUC-5).*

Nicholas Asher and Henri Wada. 1988. A computational account of syntactic, semantic and discourse principles for anaphora resolution. *Journal of Semantics*, 6:309–344.

Brack Baldwin. 1997. CogNIAC: high precision coreference with limited knowledge and linguistic resources. In *Proceedings of the ACL'97/EACL'97 Workshop on Operational factors in practical, robust anaphora resolution*, pages 38–45, Madrid, Spain.

Regina Barzilay and Michael Elhadad. 1997. Using Lexical Chains for Text Summarization. In *Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization*, Madrid, Spain.

Susan E. Brennan, Marilyn Walker Friedman and Carl J. Pollard. 1987. A centering approach to pronouns. In *Proceedings of the 25th Annual Meeting of the ACL (ACL-87)*, pages 155-162.

Jaime Carbonell and Richard Brown. 1988. Anaphora Resolution: A Multi-Strategy Approach. In *Proceedings of the 12th International Conference on Computational Linguistics*, pages 96–101.

Claire Cardie and Kiri Wagstaff. 1999. Noun phrase coreference as clustering. In *Proceedings of the Joint Conference on Empirical Methods in NLP and Very Large Corpora.*

Niyu Ge, John Gale and Eugene Charniak. 1998. Anaphora Resolution: A Multi-Strategy Approach. In *Proceedings of the 6th Workshop on Very Large Corpora, (COLING/ACL'98).*

Barbara J. Grosz, Aravind K. Joshi and Scott Weinstein. 1995. Centering: A Framework for Modeling the Local Coherence of Discourse. *Computational Linguistics*, 21(2).

M.A.K. Halliday and R. Hassan. 1976. Cohesion in English. Longman, London.

Sanda M. Harabagiu. 1998. Deriving metonymic coercions from WordNet. In *Proceedings of the Workshop of the Usage of WordNet in Natural Language Processing Systems, COLING-ACL'98*, pages 142–148.

Sanda M. Harabagiu and Dan I. Moldovan. 1998. A Parallel System for Text Inference Using Marker Propagations. *IEEE Transactions on Parallel and Distributed Systems*, 9(8):729–747.

Sanda M. Harabagiu. 1999. From Lexical Cohesion to Textual Coherence: A Data Driven Persoective. *International Journal of Pattern Recognition and Artificial Intelligence*, 13(2):1–18.

Marti A. Hearst. 1997. TextTiling: Segmenting Text into Multi-paragraph Subtopic Passages. *Computational Linguistics*, 23(1):33–64.

Lynette Hirshman and Nancy Chinchor. 1997. MUC-7 Coreference Task Definition.

Lynette Hirshman, Patricia Robinson, John Burger and Marc Vilain. 1998. The role of Annotated Training Data.

Graeme Hirst and David St-Onge. 1998. Lexical Chains as Representations of Context for the Detection and Correction of Malapropism. In *WordNet - An Electronic Lexical Database*, Edited by Christiane Fellbaum, MIT Press.

Jerry R. Hobbs. Resolving pronoun references. *Lingua*, 44:311–338.

Jerry R. Hobbs. 1979. Coherence and coreference. *Cognitive Science*, 3(1):67–90.

Jerry R. Hobbs. 1985. On the coherence and structure of discourse Technical Report CSLI-85-37, Stanford University.

Jerry R. Hobbs, Mark Stickel, Doug E. Appelt, and Paul Martin. 1993. Interpretation as abduction. *Artificial Intelligence*, 63:69–142.

Shalom Lappin and Herbert Leass. 1994. An algorithm for pronominal anaphora resolution. *Computational Linguistics*, 20(4):535–562.

Megumi Kameyama. 1997. Recognizing Referential Links: An Information Extraction Perspective. In *Proceedings of the Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts, (ACL-97/EACL-97)*, pages 46–53, Madrid, Spain.

Andrew Kehler. 1997. Probabilistic Coreference in Information Extraction. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing (SIGDAT)*, pages 163–173.

Christopher Kennedy and Branimir Bogureav. 1996. Anaphora for everyone: Pronominal anaphora resolution without a parser. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING-96)*.

William C. Mann and Sandra A. Thompson. 1988. Rhetorical Structure Theory: Toward a functional theory of text organization. *Text*, 8:243–281.

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

Joseph F. McCarthy and Wendy Lehnert. 1995. Using decision trees for coreference resolution. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI-95)*, pages 1050–1055.

Kathy McKeown. 1985. Discourse strategies for generating natural-language text. *Artificial Intelligence*, 27:1–41, 1985.

George A. Miller. 1995. WordNet: A Lexical Database. *Communication of the ACM*, 38(11):39–41.

Ruslan Mitkov. 1998. Robust pronoun resolution with limited knowledge. In *Proceedings of COLING-ACL'98*, pages 869–875.

Jane Morris and Graeme Hirst. 1991. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics* , 17:21–48.

Massimo Poesio and Renata Vieira. 1998. A corpus-based investigation of definite description use. *Computational Linguistics* , 24(2):183–216.

Philip Resnik. 1995. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI-95)*, pages 448–453.

Elaine Rich and Susan Luperfoy. 1988. An architecture for anaphora resolution. In *Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics (ACL-88)*, pages 18–24.

Bonnie Webber. Discourse deixis: Reference to discourse segments. In *Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics (ACL-88)*, pages 113–121.

Robert Wilensky. 1987. *Understanding Goal-Based Stories*. PhD thesis, Yale University, New Haven, CT.

Appendix

```
(defun name-succeed()
  (compile-axioms
   '(((((name1 e11 x1)1.2))((assume-position e21 x1)
                            (Elaboration e21 e11)) () )
     ((((assume-position e22 x2).6)
                  ((empty-position e22 x2).6))
     ((no-speculations e22 x2)) () )
      ((((Elaboration e23 e13)1.2))
        ((CoRel e13 e23 e13))())
        (() ((name1 e1 a)) () )) nil '(e1 a))
  (interpret (compile-logical-form
    '(((CoRel e1 e2 e) 10)((no-speculations e2 a)10)) nil nil)))
```

```
> (name-succeed)
0 Cost: 20 initial logical form:
  ((COREL E1 E2 E),10.00,0)((NO-SPECULATIONS E2 A),10.00,0)
1 Cost: 22.0 from expanding COREL in 0 using axiom 3.0:
  ((ELABORATION E2 E1),12.00,1)((NO-SPECULATIONS E2 A),10.00,0)
  (COREL E1 E2 E1)
2 Cost: 22.0 from expanding NO-SPECULATIONS in 0 using axiom 2.0:
  ((COREL E1 E2 E),10.00,0)((ASSUME-POSITION E2 A),6.00,1)
  ((EMPTY-POSITION E2 A), 6.00, 1)
  (NO-SPECULATIONS E2 A)
3 Cost: 24.400002 from expanding ELABORATION in 1 using axiom 1.1:
   (ASSUME-POSITION E2 A)(ELABORATION E2 X44)(NO-SPECULATIONS E2 A)
   (COREL E1 E2 E1)
8 Cost: 10 from expanding NAME1 in 3 using axiom 4.0:
   ((NO-SPECULATIONS E2 A), 10.00, 0)
   (NAME1 E1 A)(ELABORATION E2 E1)(ASSUME-POSITION E2 A)
   (COREL E1 E2 E1)
9 Cost: 6.0 from expanding NO-SPECULATIONS in 8 using axiom 2.0:
   ((EMPTY-POSITION E2 A),6.00,1)
   (NO-SPECULATIONS E2 A)(NAME1 E1 A)(ELABORATION E2 E1)
   (ASSUME-POSITION E2 A)(COREL E1 E2 E1)
10 Cost: 26.400002 from expanding NO-SPECULATIONS in 3
  using axiom 2.0:
   ((NAME1 E1 X42), 14.40, 2) ((ASSUME-POSITION E2 A),6.00, 1)
   ((EMPTY-POSITION E2 A), 6.00, 1)
   (NO-SPECULATIONS E2 A)(ELABORATION E2 E1)
   (ASSUME-POSITION E2 X42)(COREL E1 E2 E1)
```