



Information Extraction Systems and Nominal Anaphora Analysis Needs

Ireneusz Matysiak

Politechnika Wroclawska
Instytut Informatyki Stosowanej
Wybrzeże Wyspiańskiego 27,
50-370 Wrocław, Poland
ireneusz.matysiak@pwr.wroc.pl

Abstract. This paper presents the needs for anaphora analysis in Information Extraction from practical point of view. It depicts the problem of anaphora phenomenon and its role in texts together with corpus examples. It encloses a short survey of the computational treatment of nominal anaphora. In the end, a uniform approach for nominal anaphora analysis is proposed. Moreover, the paper describes requirements for corpus preparation and framework implementation.

1 Introduction

1.1 Background

An increasing demand for *Information Extraction (IE)* systems and the tendency for improving their performance (precision and recall) call for context and coherence analysis. The coherence can be obtained by different linguistic phenomena among which *anaphora*¹ plays a crucial role. Anaphora analysis for information extraction purposes was taken into consideration during the Message Understanding Conferences (MUC)², the conferences were dedicated to information extraction systems, and finally formulated as *coreference task* during information extraction at the 7th MUC ([14]). The task was restricted only to *coreferences*³ and consists in collecting of all mentions of a given entity. The

¹ This paper concerns *anaphora* as a language phenomenon, which introduces the connection between pointing back expression *anaphor* (called *referent* as well) and *antecedent*; the present article discusses only *anaphors* that can take shape of a pronoun – *pronominal anaphora*, e.g. „If I have *the dog*, I’ll walk *him*” (the pronoun *him* refers to the noun *the dog*; both point out the same object from a context) , or a non-pronominal noun phrase – *lexical noun phrase*, e.g. „I got *a small dog*. *The puppy* is great.” (the anaphor is the noun *the puppy* and points back to the noun phrase *a small dog*). Both types are contained in wider term *nominal anaphora*.

² Sometimes called the Message Understanding Competition after its competition character.

³ Coreference is regarded as type of anaphora where an identity relation, between the anaphor and the antecedent, is preserved.

present linguistic research, corpus evidences and practice combined with new sources of information like *WordNet* lead to the need for the redefinition of the coreference task.

1.2 Previous work on anaphora analysis

The first effort to define anaphora and create methods for analysing them were made in the late 70s by Sidner ([24]) and Hobbs ([15])⁴. Automatic methods from that time based on knowledge engineering – rules which mainly used *syntactic information*. Some of them were supported by late *binding theory* ([6]). Following modifications applied *cognitive techniques* i.e. *focus* (Sidner [24]), *short memory model* ([11]). Certain possibilities boundary of those algorithms can be observed in a frequently applied (and modified) Lappin & Leass' algorithm for pronominal anaphora resolution (Lappin and Leass [17]) (often used as baseline for comparison).

The 90s contributed to a better discourse and coherence understanding, which triggered the emergence of algorithms based on *centering theory* (Grosz [10]). In those times, one made an effort to use corpora for anaphora analysis and machine learning (ML) approach. Almost all kinds of ML methods were, more or less, investigated. However, for the sake of conciseness, only one species of a selected kind is quoted: *case base reasoning* in relative pronouns disambiguation ([4]), *decision trees* for coreference resolution ([18]), *unsupervised learning* (clustering) for noun phrases ([5]) (coreference resolution between noun phrases was treated as the grouping task, where algorithm found suitable similarity metrics determining the resolution). Even a *genetic algorithm* for finding salience weights in pronoun resolution system was implemented ([1]).

In an alternative corpus-based approach a large number of documents is processed for *statistical analysis* (those algorithms required a priori pre-selection of suitable features and a huge corpus size), e.g. personal pronoun resolution ([9]), non-pronominal anaphora ([12]) and coreferences (Hartrumpf [13]).

Recent scientific research tend to apply *knowledge-poor analysis* like Mitkov's approach to personal pronouns ([20]) (shallow parsing and rules which operate on empirical *antecedent indicators* independent of language; it achieved almost 90% *success rate*) or proper names coreferences ([2]) (hand-crafted orthographic rules for coreferences specific to text genre) and the machine learning methods: a decision tree for coreferences ([25]) (definite descriptions and some bridging anaphoras). There are studies on integration semantics and lexical information from various knowledge sources like *WordNet* with machine learning ([23]) or searching the Web for a particular type of lexico-syntactic pattern's statistics for resolving definite descriptions ([3]).

⁴ Some heuristics for personal pronouns were implemented in micro worlds, e.g. Winograd's SHRDLU in late 60s.

2 Anaphora in IE

There are many varieties of anaphoras, which should be resolved if one wants to understand the meaning of the text. An interesting review of the issue is done by Krahmer & Piwek ([16]). Apart from the problem of definition and denotation undertaken in their studies, characteristic properties of anaphora are worth emphasising. Especially the two of them are interesting – a type of non-coreferential relation (*bridges*)⁵ between an anaphor and the antecedent, and constraints on the relation which may have an impact on the scope. The authors enumerate a set of relations among nominal anaphora participants (though the article mentions relation for other types of anaphoras): *set membership*, *part-of* (necessary part, probable part, inducible part), which with relations from ontology (like *specialization/generalisation*), need to be investigated during analysis.

Since extraction information is driven by recognised entities (during *named entity task*), the coreference task become important part of processing, extending scope of extraction. The wider scope system has, the better results it gets, so capturing information about all mentions of a given entity (including non-coreferential relations) should be regarded as nominal anaphora resolution. Consequently, according to previous suggestions, the MUC task formulation should be expanded by adding described relations (with an effect on annotation scheme) as it is suggested by Deemter & Kibble ([8]).

In a modern approach to IE system creation machine learning is used to automate customisation to the text genre with the smallest effort. The system possibilities are limited by delivered data. To sum up, IE sets the *anaphora resolution (AR)* in the following context:

1. should cover a wide range of anaphora (at least the nominal anaphora),
2. has to have the best possible efficiency,
3. should be open for system delivered knowledge and domain (easy adjustable),
4. should be data driven (*data-enriched* approach),
5. does not need to be a general solution.

3 An uniform approach for nominal anaphora analysis

Many of the mentioned algorithms were restricted to a certain kind of anaphoras (with a few exceptions using machine learning techniques), i.e. personal pronouns, lexical noun phrases, and coreferences. However, the ways of their analysis through language levels are similar. Somehow, different kinds of anaphora require various types of knowledge, which determines the accuracy of resolving. The Figure 1 presents anaphora analysis methods at levels of the language analysis (the preliminary levels are omitted). The first and the simplest method (especially important for pronouns) usually described as *co-indexing* (after assignment of the same index to a referent and the anaphor) is based on syntactic

⁵ Sometimes are referred to as *associative* and *associative anaphora*.

knowledge like a parsing tree and morphological attributes to validate assignment. A more complicated *binding*⁶ can reflect more compound dependences coming from formalism, i.e. the binding theory, the centering theory, and sometimes looks up to the dictionary for lexical details needed for verification (the lexical knowledge could be built in formalism like in HPSG grammar).

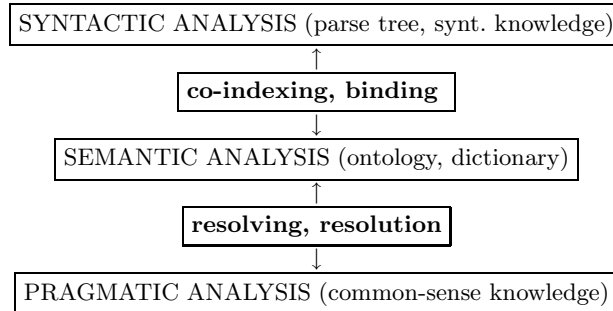


Fig. 1. Anaphora processing in the flow of natural language analysis

The terms *resolving* or *resolution* describe methods which perform a compounded analysis. They can use additional knowledge sources like ontologies, semantic dictionaries for examining potential antecedents. The method could access rules/heuristics involving a common-sense knowledge, and can be genre-specific (i.e. associative anaphora cannot do without the information about a relation between investigated entities). In some systems based on a *sense representation* (e.g. DRT) a reasoning can be carried out.

Approaches presented in Sect. 1.2 use, more or less, the knowledge sources in two ways. In the first one as *restrictions* (constraints) preventing from wrong binding. They can have various forms, starting from syntactic rules like *agreement of morphological attributes*, i.e. gender, number (a simple pronoun and his antecedent have to match number and gender, otherwise the antecedent is discounted), *semantic lexical features* like animacy, etc. (a semantic restriction from a context placed on an antecedent, must be met by an anaphor), ending with *syntactic rule* constraint taken from a selected theory like *c-command* (centering theory). The other strategy is *preference*, which is mandatory and its importance is weighted. Some examples based on syntactic knowledge are *lexical reiteration* (if a noun phrase was mentioned before, it gives more salience), *similar syntactic role* in the sentence, and *collocation* matching. Semantic preferences can be *similar semantic role*, *metonymy/holonymy/hyponymy/hypernymy* relations between an anaphor and the antecedent. Tracing the utterance object can cause preference of a *focused* object.

⁶ The distinction between co-indexing and binding is not obvious in the literature, but helps to organise terminology.

A uniform approach requires treating nominal anaphora as a whole, and should allow to integrate any type of knowledge (which determines the quality of the solution and causes that the solution is driven by data). Also, it need to be easily adapted for the target use (to the delivered knowledge and application domain).

These requirements inspired searching a predictive model that could perform mapping from observations about anaphor features to conclusions about its antecedent. The machine learning technique like a decision tree seems to be a good solution. Moreover, it could support integration of rule based algorithms as special kind of test nodes – *computational nodes*. The nodes are not ordinary attribute test, but the result from the rule/algorithm (e.g. focus tracing algorithm or even complete anaphora algorithm like co-indexation). Moreover, it might give a chance to recover from wrong rule/algorithm result (in case of the involved node is not final). The salience carried by the test (attribute or computational node) can be automatically weighed during the tree creation. In addition, the automatic tree creation can be a useful way of adjusting an algorithm to a specific genre. In that approach, there is no distinction between preferences and restrictions⁷, but the negative and final test result might be interpreted as the restriction.

4 Corpus

The corpus consists of documents published by stock companies quoted at the Warsaw Stock Exchange. The documents contain reports concerning current companies activities and events, which (according to the law) must be announced, i.e. significant agreement contraction, regulations established during a meeting of shareholders, changes of members of the board, etc⁸. The initial corpus has 4175 parsed documents (average file size is 2,5 KB). Some statistics about the corpus are presented in Table 1⁹. The collection is a starting point for creating the information extraction system. The preliminary investigation reveals that for those type of documents a formal language is used, which increases the role of nominal phrases (especially definite descriptions and proper names, see comparison in Table 1). Such situation requires more accurate valuation of a relation between an anaphor and the antecedent.

In the corpus document one can notice a rich set of relations among entities, which could be used anaphorically like part-of relation: członek zarządu (member of the board) – Zarząd (the board); part/whole relation: Zarząd (company board) – Spółka Akcyjna (registered company); set/subset: Grupa Kapitałowa (capital group) – spółka zależna (controlled company), synonymy: firma (firm) –

⁷ With exception of the case where decision tree is only preference selector.

⁸ Detailed list of the events can be obtained from *Rozporządzenie Rady Ministrów w sprawie informacji bieżących i okresowych przekazywanych przez emitentów papierów wartościowych*.

⁹ Since corpus is still in preparation the statistics were collected in a full automatic way and should be considered as rough estimation.

Table 1. Corpus in numbers

	Total	Mean per file	Mean to tokens	%
Tokens	846000	203	1	100
Noun phrases	401745	96	0.47	47
Proper names (found by parser)	18931	4.5	0.02	2
Personal pronouns	3064	0.7	0.003	0.3
Possessive pronouns	1689	0.4	0.002	0.2
Relative pronouns	1188	0.3	0.001	0.1
Reciprocal pronouns	383	0.09	0.0006	0.06

przedsiębiorstwo (company); identity of proper names: Zarządca Warszawa S.A. – Zarządca S.A.; generalization/specialization: Spółka Akcyjna (registered company) – Emitent (share drawer) – Swarżec S.A.; Akcjonariat (shareholders) – Akcjonariusz (shareholder).

4.1 Annotation scheme

For anaphora it is important to mark both a relation type, where it is possible, and additional information about the other antecedents – *anaphoric chain*. Remarks made by Deemter & Kibble ([8]) reject the MUC annotation scheme, because of the corpus evidence of relation (which are not always the identity type) between entities (anaphor – referent) and the issue of marking the same entity in different manner (e.g. separating/joining entities from compound relations). The adapted *MATE* schema ([22]), then, seems to be the most appropriate. The adaptation consists in using *MATE* scheme for marking discourse entities (enables compound relations), anaphors (enable an anaphoric chain) and types of anaphoric relations: *ident* (identity), *subset* (set/subset relation), *poss* (part/whole relation), *genrel* (generalisation/specialization (from the previous version of *MATE*))¹⁰. In original scheme annotations are added to a text in SGML manner, but it is not the most practical solution from the processing point of view. To make it independent of processing resources, like tokenizers and proper name extractors, and easy to transform into other format, annotation should be saved in XML format, with a connection to the original clean text (like file offset) instead of tokens. That assumption puts GATE¹¹ annotation scheme (changed TIPSTER scheme) into the game. In the end, these annotations can be read and processed by GATE application (General Architecture for Text Engineering ([7])) for further purposes.

¹⁰ An addition attribute *subtype* are used, but unlike *MATE* it contains information about referent form – noun/noun phrase/pronoun; its value is deduced from parser output and used for statistical purposes.

¹¹ GATE proposes a coreference annotation scheme, but it does not meet the proposed requirements.

4.2 Web annotation tool

The web annotation tool enables an easy access to the corpus and work without installation problems in any operating system. Moreover, it is simple to maintain and extend. The application consists of two parts (implemented as J2EE application).

The first part organises an access to the corpus files, saves annotation results and files status, whilst the second one transforms (using XSLT) the corpus document content into a web page with javascripts to anaphora annotation (on the web page some proper names, pronouns are marked out to speed up a selection process). In a typical session, an annotator: 1) selects an anaphor (marking in the text the beginning and the end); 2) chooses an appropriate relation (the default 'antecedent' – in case of uncertainty of the relation, 'equality' in case of the identity-of-relation anaphora, 'part' in case of the 'part/whole' relation, 'set' – 'set/subset' relation', 'similarity' – generalization/specialization etc., or 'none' if there is no antecedent); 3) links it to the antecedent (selecting it in the same way as an anaphor). A typical user session is presented in Fig. 2.

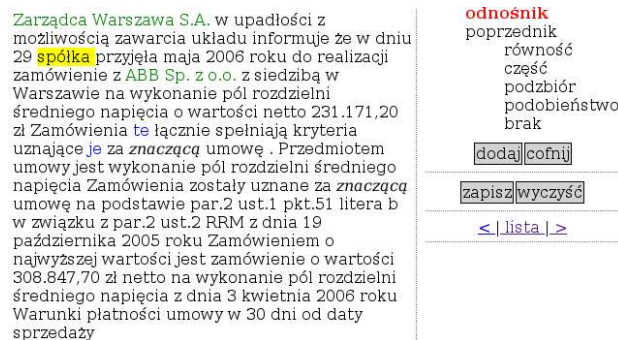


Fig. 2. Screen shot from the web application

The annotation results are kept in separate files for each user (post processed into XML as described in Sect. 4.1) and merged with the original text into one file with annotations.

5 The proposal for framework

The framework is practical realisation of the uniform approach for IE needs. It also utilises the annotation scheme (described in Sect. 4.1). It does not force implementation details, rather than helps to manage processed files and anaphora cases, organises document flow, gives suggestions about an interface for the IE *anaphora resolution module*¹² and a way the module should operate.

¹² The IE system module performing anaphora resolution task.

A typical framework document flow through different forms is depicted in Fig. 3.

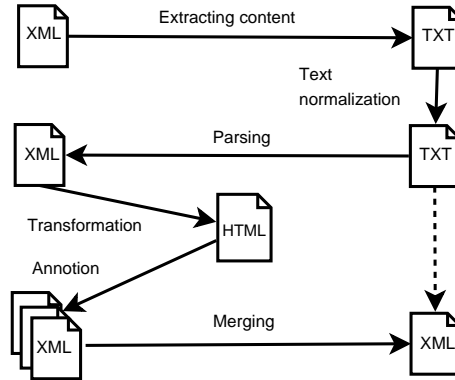


Fig. 3. Document flow

The distinction between document content and its annotated form allows using different document versions (through assignment of XSLT file to the version), so one document can be processed by various tools. Separation clean text files from processed and annotated ones requires document form on demand service from framework.

The anaphora resolution module should operate in four steps:

1. get information about anaphor and all candidate antecedents; according to the approach the system delivers anaphora context (feature vector) – attribute values and the computational nodes execution results (the feature vector should be configurable and can have a simple *CSV* format, e.g. one row contains an anaphor and a candidate features),
2. perform resolution via a decision tree (DT) (the result is a type of anaphora or none, if failed); DT could be treated as an algorithm for anaphora resolution ([18]) or antecedent filter ([21]), in the last case a selection strategy should be used,
3. choose the best antecedent according to the selection strategy (skip this step if DT is an algorithm),
4. return the antecedent – the row number (from *CSV* input) of selected antecedent (or none if it is a discourse new entity).

A module to compute feature values is dependent on the IE system, so it should be a part of IE system. Moreover, it can use different knowledge sources and algorithms, thus data which it operates rely on implementation and could be in an incompatible form (i.e. different parser output format, ontologies, and a serialization format).

Implementation of the anaphora resolution module consists of: 1) creation of associated XSLT file, which is depended on IE system processing tools like tagger, parser, tokenizer; 2) annotation of the corpus with anaphora annotations; 3) implementation of the feature value computation module (and computational nodes); 4) DT training (implementation of DT and strategy selection, if appropriate); 5) anaphora resolution module implementation (implementing access to DT and selection strategy).

The framework supporting varieties of salience features can be a useful tool for investigating various types of nominal anaphora. Furthermore, it can be a handy test bed for checking impact of different knowledge sources, processing tools on the resolution process.

6 Future work

The research is still at the conception stage and a lot of work needs to be done. First, preparation of the corpus with anaphora annotations. Second, implementation of well-known algorithms for Polish language (algorithm for personal pronouns ([19]) is promising; it was tested on few Polish hand prepared documents and obtained a success rate 93.3%; but for comparison a most-recent strategy is desirable as well), which could be used by the approach as computational nodes. Next, implementation of the IE anaphora resolution module (preparation learning cases set for automatic creation of a decision tree, using the following knowledge sources: the Polish WordNet, stock exchange ontology, parser output, selected focus and preferences techniques from the previous works). Having working framework, it would be useful to investigate some aspects, i.e. core anaphora features, which can be adjusted to a specific text genre; computational nodes; recovering from algorithm errors; an impact of different knowledge sources and errors from the previous stages (working in a fully automatic environment).

Acknowledgement. Work financed by the Polish Ministry of Education and Science, project No. 3 T11C 018 29.

References

1. Byron D., Allen J.: Applying genetic algorithms to pronoun resolution. In *AAAI/IAAI*, page 957, 1999.
2. Bontcheva K., Dimitrov M., Maynard D., Tablan V., Cunningham H.: Shallow methods for named entity coreference resolution. In *Proceedings of TALN*. TALN, 2002.
3. Bunescu R.: Associative anaphora resolution: A web-based approach, 2002.
4. Cardie C.: Corpus-based acquisition of relative pronoun disambiguation heuristics. In *Meeting of the Association for Computational Linguistics*, pages 216–223, 1992.
5. Cardie C., Wagstaff K.: Noun phrase coreference as clustering. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in NLP and Very Large Corpora*, pages 82 – 89, University of Maryland, USA, 1999.
6. Chomsky N.: *Lectures on Government and Binding: The Pisa Lectures*. Mouton de Gruyter, 1981.

7. Cunningham H.: GATE, a General Architecture for Text Engineering. *Computers and the Humanities*, 36:223–254, 2002.
8. Deemter van K., Kibble R.: On coreferring: Coreference in MUC and related annotation schemes. *Computational Linguistics*, 26(4):629–637, 2000.
9. Ge N., Hale J., Charniak E.: A statistical approach to anaphora resolution. In *Proceedings of the Workshop on Very Large Corpora*, pages 161–170, 1998.
10. Grosz B., Joshi A., Weinstein S.: Centering: a framework for modelling the local coherence of discourse. *Computational Linguistics*, 21(2):203–226, 1995.
11. Guindon R.: Anaphora resolution: Short-term memory and focusing, 1985.
12. Hall K.: A statistical model of nominal anaphora. Technical report, Brown University, 2001.
13. Hartrumpf S.: Coreference resolution with syntactico-semantic rules and corpus statistics. In *Proceedings of the Fifth Computational Natural Language Learning Workshop (CoNLL-2001)*, pages 137–144, 2001.
14. Hirschman L., Chinchor N.: MUC-7 Coreference Task Definition (version 3.0). In *Proceedings of the 7th Message Understanding Conference (MUC-7)*, Fairfax, Virginia, 1998.
15. Hobbs J.: Resolving pronoun references. *Lingua*, 44(8):311–33, 1978.
16. Krahmer E., Piwek P.: *Varieties of Anaphora: Introduction*, chapter Varieties of Anaphora, pages 1–15. Reader ESSLLI, Birmingham, 2000.
17. Lappin S., Leass H.: An algorithm for pronominal anaphora resolution. *Computational Linguistics*, 20(4):535–561, 1994.
18. McCarthy F., Lehnert W.: Using decision trees for coreference resolution. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI '95)*, 1995.
19. Mitkov R., Evans R., Orasan C., Barbu C., Jones L., Sotirova V.: Coreference and anaphora: developing annotating tools, annotated resources and annotation strategies, 2000. DAARC.
20. Mitkov R.: *Anaphora resolution*. Pearson Education, 2002.
21. Paul M., Yamamoto, K. and Sumita, E.: Corpus-based anaphora resolution towards antecedent preference, 1999.
22. Poesio M.: The mate/gnome proposals for anaphoric annotation, revisited. In *Proceedings of the 5th SIGdial Workshop on Discourse and Dialogue*, pages 154–162, Cambridge, Massachusetts, USA, April 30 - May 1 2004. Association for Computational Linguistics.
23. Poesio M., Ishikawa T., Schulte im Walde S., and Viera R.: Acquiring Lexical Knowledge for Anaphora Resolution. In *Proceedings of the 3rd Conference on Language Resources and Evaluation*, volume IV, pages 1220–1224, Las Palmas, Spain, 2002.
24. Sidner C.: Towards a computational theory of definite anaphora comprehension in english discourse. Technical report, MIT, 1979.
25. Vieira R.: Applying inductive decision trees in co-reference resolution of definite NPs, 1999. ASAI-99.