A SIMPLE BUT USEFUL APPROACH TO CONJUNCT IDENTIFICATION[1]

Rajeev Agarwal Lois Boggess

Department of Computer Science
Mississippi State University
Mississippi State, MS 39762

e-mail: kudzu@cs.msstate.edu

ABSTRACT

This paper presents an approach to identifying conjuncts of coordinate conjunctions appearing in text which has been labelled with syntactic and semantic tags. The overall project of which this research is a part is also briefly discussed. The program was tested on a 10,000 word chapter of the Merck Veterinary Manual. The algorithm is deterministic and domain independent and it performs relatively well on a large real-life domain. Constructs not handled by the simple algorithm are also described in some detail.

INTRODUCTION

Identification of the appropriate conjuncts of the coordinate conjunctions in a sentence is fundamental to the understanding of the sentence. We use the phrase 'conjunct identification' to refer to the process of identifying the components (words, phrases, clauses) in a sentence that are conjoined by the coordinate conjunctions in it. Consider the following sentence:

> "The president sent a memo to the managers to inform them of the tragic incident **and** to request their co-operation."

In this sentence, the coordinate conjunction 'and' conjoins the infinitive phrases "to inform them of the tragic incident" and "to request their co-operation". If a natural language understanding system fails to recognize the correct conjuncts, it is likely to misinterpret the sentence or to lose its meaning entirely. The above is an example of a simple sentence where such conjunct identification is easy. In a realistic domain, one encounters sentences which are longer and far more complex.

This paper presents an approach to conjunct identification which, while not perfect, gives reasonably good results with a relatively simple algorithm. It is deterministic and domain independent in nature, and is being tested on a large domain - the Merck Veterinary Manual, consisting of over 700,000 words of uncontrolled technical text. Consider this sentence from the manual:

> "The mites live on the surface of the skin of the ear **and** canal, **and** feed by piercing the skin **and** sucking lymph, with resultant irritation, inflammation, exudation, **and** crust formation".

This sentence has four coordinate conjunctions; identification of their conjuncts is moderately difficult. It is not uncommon to encounter sentences in the manual which are more than twice as long and even more complex.

The following section briefly describes the larger project of which this research is a part. Then the algorithm used by the authors and its drawbacks are discussed. The last section gives the results obtained when an implementation was run on a 10,000-word excerpt from the manual and discusses some areas for future research.

THE RESEARCH PROJECT

This research on conjunct identification is a part of a larger research project which is exploring the automation of extraction of information from structured reference manuals. The largest manual available to the project in machine-readable form is the Merck Veterinary Manual, which serves as the primary testbed. The system semi-automatically builds and updates its knowledge base. There are two components to the system - an NLP (natural language processing) component and a knowledge analysis component. (See Figure 4 at the end.)

[1] This work is supported in part by the National Science Foundation under grant number IRI-9002135.

The NLP component consists of a tagger, a semi-parser, a prepositional phrase attachment specialist, a conjunct identifier for coordinate conjunctions, and a restructurer. The tagger is a probabilistic program that tags the words in the manual. These tags consist of two parts - a mandatory syntactic portion, and an optional semantic portion. For example: the word 'cancer' would be tagged as *noun//disorder*, the word 'characterized' would be *verb/past_p*, etc. The semantic portion of the tags provides domain-specific information. The semi-parser, which is not a full-blown parser, is responsible for identifying noun, verb, prepositional, gerund, adjective, and infinitive phrases in the sentences. Any word not captured as one of these is left as a solitary 'word' at the top level of the sentence structure. The output produced by the semi-parser has very little embedding and consists of very simple structures, as will be seen below. The prepositional phrase attachment disambiguator and the conjunct identifier for coordinate conjunctions are considered to be "specialist" programs that work on these simple structures and manipulate them into more deeply embedded structures. More such specialist programs are envisioned for the future. The restructurer is responsible for taking the results of these specialist programs and generating a deeper structure of the sentence. These deeper structures are passed on to the knowledge analysis component.

The knowledge analysis component is responsible for extracting from these structures several kinds of objects and relationships to build and update an object-oriented knowledge base. The system can then be queried about the information contained in the text of the manual.

This paper primarily discusses the conjunct identifier for coordinate conjunctions. Detailed information about the other components of the system can be found in [Hodges et al., 1991], [Boggess et al., 1991], [Agarwal, 1990], and [Davis, 1990].

CONJUNCT IDENTIFICATION

The program assigns a **case label** to every noun phrase in the sentence, depending on the role that it fulfills in the sentence. A large proportion of the nouns of the text have semantic labels; for the most part, the case label of a noun phrase is the label associated with the head noun of the noun phrase. In some instances, a preceding adjective influences the case label of the noun phrase, as, for example, when an adjective with a semantic label precedes a generic noun. A number of the resulting case labels for noun phrases (e.g. *time*, *location*, etc.) are similar those suggested by Fillmore [1972], but domain dependent case labels (e.g. *disorder*, *patient*, etc.) have also been introduced. For example: the noun phrase "a generalized dermatitis" is assigned a case label of *disorder*, while "the ear canal" is given a case label of *body_part*. It should be noted that, while the coordination algorithm assumes the presence of semantic case labels for noun phrases, based on semantic tags for the text, it does not depend on the specific values of these labels, which change from domain to domain.

THE ALGORITHM

The algorithm makes the simplifying assumption that each coordinate conjunction conjoins only two conjuncts. One of these appears shortly after the conjunction and is called the **post-conjunct**, while the other appears earlier in the sentence and is referred to as the **pre-conjunct**.

The identification of the post-conjunct is fairly straightforward: the first complete phrase that follows the coordinate conjunction is presumed to be the post-conjunct. This has been found to work in all of the sentences on which this algorithm has been tested. The identification of the pre-conjunct is somewhat more complicated. There are three different levels of rules that are tried in order to find the matching pre-conjunct. These are referred to as level-1, level-2, and level-3 rules in decreasing order of importance. The steps involved in the identification of the pre- and the post-conjunct are described below.

(a) The sentential components (phrases or single words not grouped into a phrase by the parser) are pushed onto a stack until a coordinate conjunction is encountered.

(b) When a coordinate conjunction is encountered, the post-conjunct is taken to be the immediately following phrase, and its type (noun phrase, prepositional phrase, etc.) and case label are noted.

(c) Components are popped off the stack, one at a time, and their types and case labels are compared with those of the post-conjunct. For each component that is popped, the rules at level-1 and level-2 are tried first. If both the type and case label of a popped component match those of the post-conjunct (level-1 rule), then this component is taken to be the pre-conjunct. Otherwise, if the type of the popped component is the same as that of the post-conjunct and the case label is **compatible** (case labels like *medication* and *treatment*, which are semantically

```
sentence([
    noun_phrase(case_label(body_part), [(the, det), (ear, noun | | body_part)])
    verb_phrase([(should, aux), (be, aux), (cleaned, verb | past_p)])
    prep_phrase([(by, prep),
            gerund_phrase([(flushing, verb | gerund)])])
    word([(away, adv | | location)])
    noun_phrase(case_label(unknown), [(the, det), (debris, noun)])
    word([(and, conj | co_ord)])
    noun_phrase(case_label(body_fluid), [(exudate, noun | | body_fluid)])
    gerund_phrase([(using, verb | gerund),
            noun_phrase(case_label(medication), [(warm, adj), (saline,
                    adj | | medication), (solution, noun | | medication)])])
    word([(or, conj | co_ord)])
    noun_phrase(case_label(unknown), [(water, noun)])
    prep_phrase([(with, prep),
        noun_phrase(case_label(medication), [(a, det), (very, adv | | degree),
                (dilute, adj | | degree), (germicidal, adj | | medical),
                (detergent, noun | | medication)])])
    word([(comma, punc)])
    word([(and, conj | co_ord)])
    noun_phrase(case_label(body_part), [(the, det), (canal, noun | | body_part)])
    verb_phrase([(dried, verb | past_p)])
    word([(as, conj | correlative)])
    word([(gently, adv)])
    word([(as, conj | correlative)])
    adj_phrase([(possible, adj)])
]).
```

Figure 1

similar, are considered to be compatible) to that of the post-conjunct (level-2 rule), then this component is identified as the pre-conjunct. If the popped component satisfies neither of these rules, then another component is popped from the stack and the level-1 and level-2 rules are tried for that component.

(d) If no component is found that satisfies the level-1 or level-2 rules and the beginning of the sentence is reached (popping components off the stack moves backwards through the sentence), then the requirement that the case label be either the same or compatible is relaxed. The component with the same type as that of the post-conjunct (irrespective of the case label) that is closest to the coordinate conjunction, is identified as the pre-conjunct (level-3 rule).

(e) If a pre-conjunct is still not found, then the post-conjunct is conjoined to the first word in the sentence.

Although there is very little embedding of phrases in the structures provided by the semi-parser, noun phrases may be embedded in prepositional phrases, infinitive phrases, and gerund phrases on the stack. The algorithm does permit noun phrases that are post-conjuncts to be conjoined with noun phrases embedded as objects of, say, a previous prepositional phrase (e.g., in the sentence fragment "in dogs and cats", the noun phrase 'cats' is conjoined with the noun phrase 'dogs' which is embedded as the object of the prepositional phrase 'in dogs'), or other similar phrases.

We have observed empirically that, at least for this fairly carefully written and edited manual, long distance conjuncts have a strong tendency to exhibit high degrees of parallelism. Hence, conjuncts that are physically adjacent may merely be of the same syntactic type (or may even be syntactically dissimilar); as the distance between conjuncts increases, the degree of parallelism tends to increase, so that conjuncts are highly likely to be of the same semantic category, and syntactic and even lexical repetitions are to be found (e.g., on those occasions when a post-conjunct is to be associated with a prepositional phrase that occurs 30 words previous, the preposition may well be repeated). The gist of the algorithm, then, is as follows: to look for sentential components with the same syntactic and semantic categories as the post-conjunct, first nearby and then with increasing distance toward the beginning of the sentence; failing to find such, to look for the same syntactic category,

```
sentence([
    prep_phrase([(with, prep),
    noun_phrase([(persistent, adj|1time), (or, conj|co_ord), (untreated, adj),
                    (otitis_externa, noun|1disorder)])])
    word([(comma, punc)])
    noun_phrase([(the, det), (epithelium, noun)])
    prep_phrase([(of, prep),
                    noun_phrase([(the, det), (ear, noun|1body_part),
                                    (canal, noun|1body_part)])])
    verb_phrase([(undergoes, verb|3sg)])
    noun_phrase([(hypertrophy, noun|1disorder)])
    word([(and, conj|co_ord)])
    verb_phrase([(becomes, verb|beverb|3sg)])
    adj_phrase([(fibroplastic, adj|1disorder)])
]).
```

Figure 2

first close at hand and then with increasing distance, and if all else fails to default to the beginning of the sentence as the pre-conjunct (the semi-parser does not recognize clauses as such, and there may be no parallelism of any kind between the beginnings of coordinated clauses). Provisions must be made for certain kinds of parallelism which on the surface appear to be syntactically dissimilar - for example, the near-equivalence of noun and gerund phrases. In the text used as a testbed, gerund phrases are freely coordinated with noun phrases in virtually all contexts. Our probabilistic labelling system is currently being revised to allow the semantic categories for nouns to be associated with gerunds, but at the time this experiment was conducted, gerund phrases were recognized as conjuncts with nouns only on syntactic grounds - a relatively weak criterion for the algorithm. Further, there are instances in the text where prepositional phrases are conjoined with adjectives or adverbs - the results reported here do not incorporate provisions for such. Consider the sentence "The ear should be cleaned by flushing away the debris and exudate using warm saline solution or water with a very dilute germicidal detergent, and the canal dried as gently as possible." The semi-parser produces the structure shown in Figure 1. The second 'and' conjoins the entire clause preceding it with the clause that follows it in the sentence. Although the algorithm does not identify clause conjuncts, it does identify the beginnings of the two clauses, "the ear" and "the canal", as the pre- and post-conjuncts, in spite of several intervening noun phrases. This is possible because the case labels of both these noun phrases agree (they are both *body_part*).

THE DRAWBACKS

Before reporting the results of an implementation of the algorithm on a 10,000 word chapter of the Merck Veterinary Manual we describe some of the drawbacks of the current implementation.

(i) The algorithm assumes that a coordinate conjunction conjoins only two conjuncts in a sentence. This assumption is often incorrect. If a construct like [A, B, C, and D] appears in a sentence, the coordinate conjunction 'and' frequently, but not always, conjoins all four components. (B, for example, could be parenthetical.) The implemented algorithm looks for only two conjuncts and produces a structure like [A, B, [and [C, D]]], which is counted as correct for purposes of reporting error rates below. Our "coordinate conjunction specialist" needs to work very closely with a "comma specialist" - an as-yet undeveloped program responsible for, among other things, identifying parallelism in components separated by commas.

(ii) The current semi-parser recognizes certain simple phrases only and is unable to recognize clause boundaries. For the conjunct identifier, this means that it becomes impossible to identify two clauses with appropriate extents as conjuncts. The conjunct identifier has, however, been written in such a way that whenever a "clause specialist" is developed, the final structure produced should be correct. Therefore, the conjunct identifier was held responsible for correctly recognizing only the beginnings of the clauses that are being conjoined.

Similarly, for phrases not explicitly recognized by the semi-parser, the current conjunct specialist is expected only to conjoin the beginnings of the phrases - not to somehow bound the extents of the phrases. Consider the

```
sentence([
    noun_phrase([(antibacterial, adj|medication),
                 (drugs,noun|plural||medication)])
    verb_phrase([(administered, verb|past_p)])
    prep_phrase([(in, prep),
                 noun_phrase([(the, det),(feed, noun)])])
    verb_phrase([(appeared, verb|beverb)])
    inf_phrase([(to, infinitive), verb_phrase([(be, verb|beverb)]),
                 adj_phrase([(effective, adj)])])
    prep_phrase([(in, prep),
    noun_phrase([(some, adj||quantity),
                 (herds, noun|plural||patient)])])
    word([w(and, conj|co_ord)])
    prep_phrase([(without, prep),
                 noun_phrase([(benefit, noun)])])
    prep_phrase([(in, prep),
                 noun_phrase([(others, pro|plural)])])
]).
```

Figure 3

sentence "With persistent or untreated otitis externa, the epithelium of the ear canal undergoes hypertrophy and becomes fibroplastic." The structure received by the coordination specialist from the semi-parser is shown in Figure 2. In this sentence, the components "undergoes hypertrophy" and "becomes fibroplastic" are conjoined by the coordinate conjunction 'and'. The conjunct identifier only recognizes the verb phrases "undergoes" and "becomes" as the pre- and post-conjuncts respectively and is not expected to realize that the noun phrases following the verb phrases are objects of these verb phrases.

(iii) Although it is generally true that the components to be conjoined should be of the same type (noun phrase, infinitive phrase, etc.), some cases of mixed coordination exist. The current algorithm allows for the mixing of only gerund and noun phrases. Consider the sentence "Antibacterial drugs administered in the feed appeared to be effective in some herds and without benefit in others." The structure that the coordination specialist receives from the semi-parser is shown in Figure 3. Note that the prepositional phrases are eventually attached to their appropriate components, so that the phrase "in some herds" ultimately is attached to the adjective "effective". The system does not include any rule for the conjoining of prepositional phrases with adjectival or adverbial phrases. Hence the phrases "effective in some herds" and "without benefit in others" were not conjoined.

RESULTS AND FUTURE WORK

The algorithm was tested on a 10,000 word chapter of the Merck Veterinary Manual. The results of the tests are shown in Table 1. We are satisfied with these results for the following reasons:

(a) The system is being tested on a large body of uncontrolled text from a real domain.

(b) The conjunct identification algorithm is domain independent. While the semantic labels produced by the probabilistic labelling system are domain dependent, and the rules for generalizing them to case labels for the noun phrases contain some domain dependencies (there is some evidence, for example, that a noun phrase

Table 1: Results of the algorithm on the 'Eye and Ear' chapter

Conjunction	Total Cases	Correct Cases	Percentage
and	366	305	83.3%
or	137	109	79.6%
but	41	30	73.2%
TOTAL	544	444	81.6%

consisting of a generic noun preceded by a semantically labelled modifier should not always receive the semantic label of the modifier) the conjunct specialist pays attention only to whether the case labels match – not to the actual values of the case labels.

(c) The true error rate for the simple conjunct identification algorithm alone is lower than the 18.4% suggested by the table, and making some fairly obvious modifications will make it lower still. The entire system is composed of several components and the errors committed by some portions of the system affect the error rate of the others. A significant proportion of the errors committed by the conjunct identifier are due to incorrect tagging, absence of semantic tags for gerunds, improper parsing, and other matters beyond its control. For example, the fact that gerunds were not marked with the semantic labels attached to nouns has resulted in a situation where any gerund occurring as post-conjunct is preferentially conjoined with any preceding generic noun. More often than not, the gerund should have received a semantic tag and would properly be conjoined to a preceding non-generic noun phrase that would have been of the same semantic type. (The conjunction specialist is not the only portion of the system which would benefit from semantic tags on the gerunds; the system is currently under revision to include them.)

From an overall perspective, the conjunct identification algorithm presented above seems to be a very promising one. It does depend a lot upon help received from other components of the system, but that is almost inevitable in a large system. The identification of conjuncts is vital to every NLP system. However, the authors were unable to find references to any current system where success rates were reported for conjunct identification. We believe that the reason behind this could be that most systems handle this problem by breaking it up into smaller parts. They start with a more sophisticated parser that takes care of some of the conjuncts, and then employ some semantic tools to overcome the ambiguities that may still exist due to co-ordinate conjunctions. Since these systems do not have a "specialist" working solely for the purpose of conjunct identification, they do not have any statistic about the success rate for it. Therefore, we are unable to compare our success rates with those of other systems. However, due to the reasons given above, we feel that an 81.6% success rate is satisfactory.

We have noted several other modifications that would improve performance of the conjunct specialist. For example, it has been noticed that the coordinate conjunction 'but' behaves sufficiently differently from 'and' and 'or' to warrant a separate set of rules. The current algorithm also ignores lexical parallelism (direct repetition of words already employed in the sentence), which the writers of our text frequently use to override plausible alternate readings. The current algorithm errs in most such contexts. As mentioned above, the algorithm also needs to allow prepositional phrases to be conjoined with adjectives and adverbs in some contexts. Some attempt was made to implement such mixed coordination as a last level of rules, level-4, but it did not meet with a lot of success.

FUTURE RESEARCH

In addition to the above, the most important step to be taken at this point is to build the comma specialist and clause recognition specialist. Another problem that needs to be addressed involves deciding priorities when one or more prepositional phrases are attached to one of the conjuncts of a coordinate conjunction. For example, we need to decide between the structures [[A and B] in dogs] and [A and [B in dogs]], where A and B are typically large structures themselves, A and B should be conjoined, and 'in dogs' may appropriately be attached to B. It is not clear whether the production of the appropriate structure in such cases rightfully belongs to the knowledge analysis portion of our system, or whether most such questions can be answered by the NLP portion of our system with the means at its disposal. Further, the basic organization of the NLP component, with the tagger and the semi-parser generating the flat structure and then the various specialist programs working on the sentence structure to improve it, looks a lot like a blackboard system architecture. Therefore, one of the future ventures could be to try to look into some blackboard architecture and assess its applicability in this system.

Finally, there are ambiguities inherently associated with coordinate conjunctions, including the problem of differentiating between "segregatory" and "combinatory" use of conjunctions [Quirk et al., 1982] (e.g. "fly and mosquito repellants" could refer to 'fly' and 'mosquito repellants' or to 'fly repellants' and 'mosquito repellants'), and the determination of whether the 'or' in a sentence is really used as an 'and' (e.g. "dogs with glaucoma or keratoconjunctivitis will recover" implies that dogs with glaucoma and dogs with keratoconjunctivitis will recover). The current algorithm does not address these issues.

REFERENCES

Agarwal, Rajeev. (1990). "Disambiguation of prepositional phrase attachments in English sentences using case grammar analysis." MS Thesis, Mississippi State University.

Boggess, Lois; Agarwal, Rajeev; and Davis, Ron. (1991). "Disambiguation of prepositional phrases in automatically labeled technical text." In *Proceedings of the Ninth National Conference on Artificial Intelligence*:1: 155-9.

Davis, Ron. (1990). "Automatic text labelling system." MCS project report, Mississippi State University.

Fillmore, Charles J. (1972). "The case for case." Universals in Linguistic Theory, Chicago Holt, Rinehart & Winston, Inc. 1-88.

Hodges, Julia; Boggess, Lois; Cordova, Jose; Agarwal, Rajeev; and Davis, Ron. (1991). "The automated building and updating of a knowledge base through the analysis of natural language text." Technical Report MSU-910918, Mississippi State University.

Quirk, Randolph; Greenbaum, Sidney; Leech, Geoffrey; and Svartvik, Jan. (1982). A comprehensive grammar of the English language. Longman Publishers.

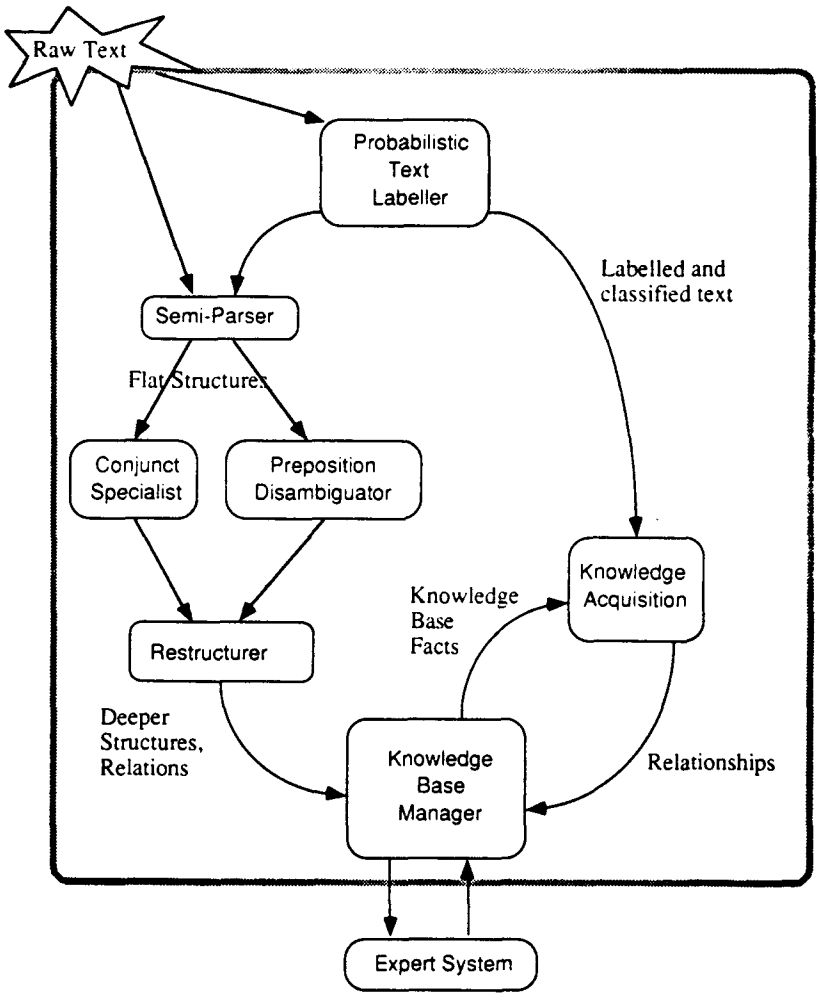

Figure 4: Overall System