Abstract

We address the problem of ordering several circumstantials when generating or revising a clause. This problem occurs in the context of a multi-document summarization system that relies on language generation to incrementally reformulate the wording of fragments of sentences extracted from the documents.

We present the results of an extensive corpus analysis of the relative position of different types of circumstantials. Our approach learns a set of rules using parameters that can be effectively used by our system. Evaluation indicates that these rules, which we have implemented in our text generator, attain a high level of precision (95.4% over a baseline of 78.6%).

Ordering Circumstantials for Multi-Document Summarization

Michael Elhadad*, Yael Netzer*
Regina Barzilay**, Kathleen McKeown**
(*) Dept of Computer Science
Ben Gurion University of the Negev
84105 Beer Sheva, Israel
(elhadad—yaeln)@cs.bgu.ac.il
(**) Dept of Computer Science
Columbia University
New York, NY 10027, USA
(regina—kathy)@cs.columbia.edu

April 2, 2001

1 Introduction

While a language generation grammar controls ordering of words and constituents within a sentence, there are phrases whose position in a sentence is underconstrained by grammar. These phrases can be freely ordered within a sentence, but only to a certain extent. Circumstantials (sentence modifiers), adjectival modifiers of nouns, and prepositional phrase modifiers of nouns all fall within this category. While circumstantials (*e.g.*, time and location modifiers of the main sentential proposition) typically appear at either the beginning or end of the sentence, when there are multiple circumstantials, a generator must determine which ones appear at the beginning and which ones at the end and in which order at beginning and end.

We are exploring the problem of ordering phrases for language generation in the context of a system which generates summaries of the similarities be-

tween multiple input documents [1, 9]. To generate a summary, this system finds similar sentences across articles, using statistical techniques to extract sentences which essentially talk about the "same thing." It then uses a deeper analysis of the sentences, using intersection of the parsed sentences to find phrases within the sentences that are repetitive. The selected phrases then form input to the generator which must combine the phrases to form a coherent sentence for the output summary. In this process, the sentence generator must determine the appropriate ordering for the phrases extracted from multiple sentences, rewording them when necessary to form fluent output.

This application places a number of demands on the problem. Because we are developing robust, domain independent techniques for document summarization, analysis of the input text must necessarily be relatively shallow. As a result, the type of information available for the generator to use in ordering phrases is limited. The system does not have detailed semantic knowledge about the connections between phrases in the input sentence. It does not know, for example, the explicit temporal relations between events that are specified by temporal circumstantials in the sentence.

Similar demands exist for a growing set of applications that rely on revision or re-generation of text (cf. for example, [11], [7]).

In this paper, we study the ordering of multiple circumstantials within a sentence, investigating how temporal, location and other types of circumstantials can combine. While it may appear that circumstantials can appear in any order within a sentence, our study shows that depending upon the situation different orderings are possible. Furthermore, selecting an implausible ordering can result in the generation of unwarranted inferences as the result of ambiguity in addition to merely awkward sentences. In this paper, we describe several constraints on ordering of circumstantials, based on a corpus analysis, and present a revision algorithm in FUF/SURGE, a robust language generator, that combines circumstantial phrases with a main semantic proposition, deriving and representing ordering constraints.

Figure 1 indicates the types of decisions our system must face. In this case, a base clause (*An OH58 military helicopter made an emergency landing*) must be revised to include a time modifier (*at 9:15pm*), location (*in North Korea*) and source (*DoD*). A variety of orderings can be generated. However, only a subset of those orderings preserve the correct meaning of the sentence. While the first and second sentences have different semantic nuances, the third sentence may mislead the reader by creating the wrong attachment

An OH58 military helicopter made an emergency landing at about 9:15pm in North Korea the Department of Defense said.

In North Korea at about 9:15pm an OH58 military helicopter made an emergency landing the Department of Defense said.

The Department of Defense said at about 9:15pm an OH58 military helicopter made an emergency landing in North Korea.

Figure 1: Different circumstantial ordering

relation between the source and time modifiers.

In the following sections, we first present background data on the ordering of circumstantials within a clause and state the problem we address more precisely. Then, we present a set of parameters corresponding to factors identified in linguistic studies and that are readily available in the input to our summarization system. We have extracted these features from a large scale corpus of news reports and run a learning algorithm (the RIPPER system). We present overall statistics characterizing the corpus in Section 5.

The baseline for the problem of correctly predicting the placement of a circumstantial within a clause is around 78.6%, assuming that it is always placed in the end position. We then present the set of rules learned by our automated analysis. These rules predict the correct placement of circumstantials in 95.0% of the cases. We identify the most significant features in the rules and conclude with a discussion of future work.

2 Problem Description: Circumstantials Ordering

We present in this section the terminology and basic hypotheses that have been advanced in linguistic studies concerning ordering of circumstantials and, using this terminology, formulate the problem.

Within the syntactic structure of a sentence, it is customary to distinguish between arguments and circumstantials (also known as adjuncts) which are less closely related to the clause. While English grammars strongly constrain the position of arguments, the position of circumstantials is much more underconstrained. Several factors, however, beyond pure syntax, do constrain

-BNF	benefactive (verbs with dative shifts)	*I baked a cake* ***for Doug***
-DIR	direction (from where, to where)	*I flew* ***from Tokyo*** *to Paris*
-EXT	extension (spatial)	*She walked* ***5 miles***
-LOC	place/setting location of event (also metaphorical)	*I'll be happy on the moon*
-MNR	manner and instrument	*She hit the nail* ***impatiently*** *with a hammer*
-PRP	purpose or reason	*I called Maya* ***to check if she's home***
-TMP	temporal or aspectual adverbials (when, how often, for how long)	*It was nice meeting you* ***in the morning***

Table 1: PENN TREEBANK Circumstantial types

the position of circumstantials.

2.1 Types of Circumstantials

While several arguments for semantic classifications of circumstantials have been proposed in the literature, we rely on the empirical results of the PENN TREEBANK annotation team that has identified a set of seven circumstantial types that have been reliably recognized by annotators. These circumstantial types are presented in Figure 2.1.

In addition to their semantic tags, circumstantials are characterized by their syntactic tags, namely adverbial, prepositional phrase or clause.

2.2 Position of Circumstantials

When referring to the position of a circumstantial in a clause, we consider (1) where in the clause it is located and (2) within each particular location, its relative position with respect to other circumstantials. [10] distinguish 4 options:

I – Initial (before the subject)

M1 – Middle – right before verb

M2 – Middle – right after verb

E – end position (after all arguments)

Hasselgard [5] extends the possible positions in a clause, as shown in the following example:

(I) the book (iM) must (M) have (mM) been (eM) placed (iE) on the shelf (E)

Our preliminary empirical investigation indicates that the distinction between the different types of "middle" positions is quite difficult; that is, it seems that circumstantials tend to move quite freely among the different middle slots. To avoid this issue, we have simplified the categorization of locations to only three positions: [I] Initial (before the subject), [E] End (after all arguments) and [M] for any other position.

Hasselgard [5] discusses another aspect of circumstantials ordering, related to multiple circumstantials:

Clusters: multiple circumstantials occur in the same slot (I, M or E).

Combinations: multiple circumstantials are distributed among several slots (I, M and/or E).

Using this terminology, the problem we address in this paper can be reformulated more precisely as follows:

- When a single circumstantial is to be added to a clause, in which slot is it assigned (I, M or E).
- When a circumstantial is to be added to a clause, does the presence of other circumstantials in different slots constrain the creation of a combination?

2.3 Parameters Influencing the Position of Circumstantials

Linguistic grammars have remained quite fuzzy when determining the relative position of circumstantials within a clause. For example, according to [10], 3 main principles govern the relative position of circumstantials:

1. the unmarked order (in our case, placement of most circumstantials in end position) can be changed when needed,
2. a subordinate clause usually appears after other constituents,
3. longer circumstantials follow shorter ones.

Hasselgard [5] has inspected a corpus of various genres focusing on the position of time and location circumstantials. She found that the end position is clearly unmarked for most types of time and space adverbials. She proposes a classification of the factors that influence the ordering along the following classes:

Syntactic: clause type (main or subordinate) and type of the circumstantial obligatoriness.

Semantic: semantic class of the circumstantial, attachment, anaphora/antecedent relation, scope.

Pragmatic: focus, information structure (given/new), cognitive order

Textual: end-weight (sentence balance), clarity, genre.

While Hasselgard's classification of parameters influencing circumstantials ordering is useful, it is not immediately operational for an automated text revision system. This justifies the exploration of corpus-based alternatives, which have provided robust solutions to similar problems. For example, [12] exploited corpus-based methods for determining ordering among premodifiers in noun phrases with high precision.

3 Parameters

Our method starts with the definition of a set of parameters that correspond to features that can be readily computed from our input and verified on our corpus.

The features we have selected are shown in Table 3.

The semantic types we use are exactly those available in the PENN TREEBANK annotations. The decision to use this quite general classification instead of a more precise one (for example, distinguishing among frequency, duration, location instead of only temporal) is related to 2 characteristics of our application: first, we haven't found a reliable method to tag the more precise categories in our training corpus; second, when the text generator is to insert a time modifier inside a sentence, a precise semantic tag for the new circumstantial is available (it comes from a semantic representation), but the tag of the existing modifiers in the clause, as they have been parsed in the

Feature	**Example Value**
Semantic type of circumstantial	`LOC, TMP`
Syntactic tag of circumstantial (tag)	`ADVP, S`
Parent tag (parenttag)	`S`
Grand-parent tag (grand)	`S, none`
Lexical head of circumstantial (head)	*previously*
Depth of attachment (nest)	integer
Number of siblings (siblings)	integer
Number of sibling circumstantials (brothers)	integer
Number of words in circumstantial (wc)	integer
Syntactic depth of circumstantial (mydepth)	integer

Table 2: List of features used in the experiment

original sentence fragments of the source texts is not precise. In fact, our parser [2] produces the same PENN TREEBANK tags.

The description of the syntactic configuration in which the circumstantial is to be inserted is also constrained by the type of input we are expecting. The training corpus distinguishes between S-like categories and VPs. We first trained the system using as a feature the syntactic tag of the parent constituent of circumstantials. It turned out that the determining feature was the value of the syntactic tag: if S, the position of the circumstantial is [I]nitial, if VP, the position is [E]nd. This rule, however, is of little use for a generator: the decision whether to attach a circumstantial under S or under VP actually corresponds almost exactly to the decision to position the circumstantial in initial or final position. Our input cannot assume that this decision is already made. Instead, we adopt the systemic approach of ignoring the VP level in the clause, placing all arguments and modifiers directly under the S node. Accordingly, when analyzing the corpus, we computed the "parent-tag" and "grand-parent-tag" features by ignoring all VP nodes and jumping up to the dominating S node.

Length of the circumstantials also plays a role in how they are ordered. When two circumstantials appear side by side and both modify the proposition, longer, heavy circumstantials tend to be placed after shorter ones. This corresponds to a general stylistic preference [6]. Since we were not sure which parameter would best characterize weight, we computed a set of related features: number of words, number of siblings (other constituents under the

Total clauses	37,203	100.00%
Clauses with 1 circumstantials	28745	77.26%
Clauses with 2 circumstantials	6821	18.33%
Clauses with 3 circumstantials	1409	1.20%
Clauses with 4 circumstantials	203	3.70%
Clauses with >5 circumstantials	25	0.06%
<TMP>	21765	45.8%
<LOC>	9009	18.9%
<MNR>	4813	10.1%
<BNF>	54	0.1%
<EXT>	2348	4.9%
<DIR>	5866	12.3%
<PRP>	3742	7.9%

Table 3: Size of the corpus

same parent node) and syntactic depth of the circumstantial.

4 The Corpus

The corpus we used in our experiment is the Wall Street Journal corpus from the PENN TREEBANK [8]. We have extracted all clauses (trees marked as S-nodes in the PENN TREEBANK) which contain marked circumstantial, specifically 37,203 clauses. Table 4 indicates the number of circumstantials found in the corpus by semantic type, and shows the distribution of circumstantials over the clauses.

To establish a baseline to the problem of assigning a position to a circumstantial, we computed the distribution of the position of circumstantials as a function of their semantic type (cf. Table 4). The End position appears as a clear baseline, with an overall precision of 78.6%. For the most frequent circumstantial types, however (TMP and LOC which are the most important for our application), the precision is lower (67.8% and 76.2% respectively).

Type	Init	Middle	End
TMP	22.0% (4795)	10.2% (2222)	**67.8%** (14748)
LOC	22.4% (2017)	1.4% (125)	**76.2%** (6867)
MNR	2.1% (102)	19.6% (945)	**78.2%** (3764)
PRP	10.4% (389)	2.3% (10)	**89.3%** (3343)
BNF	0.0% (0)	0.0% (0)	**100.0%** (54)
EXT	0.1% (2)	0.1% (3)	**99.8%** (2343)
DIR	0.0% (1)	0.0% (0)	**100.0%** (5865)
Total	15.4% (7310)	6.0% (2876)	**78.6%** (37411)

Table 4: Distribution of position per semantic type

Circum. Type	Precision	Baseline
TMP	94.8% ±0.20%	67.8%
LOC	97.6% ±0.08%	76.1%
MNR	83.0% ±0.72%	78.2%
PRP	98.8% ±0.50%	89.3%
Average	**95.4%**	**77.9%**

Table 5: Precision of the learned rules over all circumstantial types

5 Results

Using our dataset, we calculated for each clause all the features described in Section 3. Because the distribution among circumstantial types is different, we learned the rules for each category separately. Otherwise, frequent types would have dominated over others, and less frequent types would not have been considered by the learning algorithm. We excluded from our dataset BNF, EXT and DIR, since they always occur in the end position. We used RIPPER with ten-fold cross validation to learn rules that predict the position of a circumstantial given the value of the parameters listed in Table 3.

The set of rules learned provide us with an overall precision of 95.0% (over a baseline of 78.6%). The precision for each semantic type is shown in Table 5.

Some of the most significant rules are shown in Table 5.

We analyzed 70 rules learned by the system in order to find the impact

middle	$\vdash$ parenttag $\sim$ S, head $\sim$ recently, wc$\leq$1, siblings$\leq$3
init	$\vdash$ parenttag $\sim$ S, wc$\geq$2, nest$\leq$2

Table 6: Sample rules learned by the system

of each feature. The features with major effect were:

sentence complexity: The number of siblings has a major impact on circumstantial placement. This feature was included in 50 rules. Surprisingly, it has a different impact on different types of circumstantials. For temporal circumstantials, when number of siblings is bigger than 3, it tends to be in the initial or end position. However, manner circumstantials in this situation are placed in the middle position.

weight features: Two features in this category, depth of circumstantial and number of words, were the strongest features for all types of circumstantials. For instance, 23 rules regarding middle position involve restriction on number of words and depth: wc$\leq$2 and depth$\leq$1.

lexical features: For temporal and manner circumstantials, 30 rules derived by RIPPER include the `head` feature. We found that many lexical heads, picked up by RIPPER as predictors of middle position, belong to the *frequency* subtype of temporal circumstantial, *e.g.*, *"never", "often", "rarely"*. Lexical features are useful for our task, since they provide an approximation of the circumstantial subtype.

parenttag: The syntactic category of the node which is modified by the circumstantial has a strong impact on circumstantial placement. For instance, for LOC circumstantials, `SINV` parent tag is sufficient predictor for the init position.

On the other hand, nesting level of the clause and its grandfather played much smaller roles.

We also concluded from rule analysis that semantic type of circumstantial is important. While some placement rules are applicable for all types, some rules are type specific. Therefore, it is important to provide this information to the generation component.

We also examined whether the placement of the circumstantial is influenced by other circumstantials in that clause. For this purpose, we introduced

the feature *brothers*, which stands for the number of sibling circumstantials. This feature was included in the majority of middle position rules with the same value — zero. It shows that when circumstantials of more than one type are present in the sentence, none of them tend to use middle position.

Another issue that we examined in our analysis is why prediction of manner circumstantial placement has such a low accuracy (83.5% over baseline of 78.2%). We found that the placement of manner circumstantials is particularly free; for example, in the sentence *"... David carefully analyzed the data" carefully* can be moved to end position: *"... David analyzed the data carefully"*. Informal evaluation with native English speakers validates this hypothesis: on many cases of disagreement between the learned rules and the corpus, human judges found the rule-predicted placement acceptable as well as the corpus observed one.

6 Implementation in FUF/SURGE

In order to be used for multi-document summarization, the generator in our application, (FUF/SURGE) [4], has been modified to support a non-standard hybrid input which includes both full phrases (described by syntactic trees) and semantic constituents (described by more abstract semantic values). Similarly, the implementation of the circumstantial rules in FUF/SURGE must encode syntactic features (semantic category, number of words and depth) that we expect to receive for a semantic input.

The resulting grammar places circumstantials using the directives encoded by the rules learned in our experiment. Some of the features used in the rules are problematic. We addressed these cases using the following special techniques within FUF/SURGE:

Weight features: The features *depth of circumstantial* and *number of words* refer to properties of the syntactic representation of the constituent to be inserted in the base clause. However, the syntactic representation is not available before the constituent is generated. We have implemented the rules referring to these features in two ways: using delayed unification (FUF's: wait control annotation [3]), we require the test of the depth/number of words to be delayed until the constituent has been unified. Second, to avoid the inefficiency of a pure generate-and-test approach, we have added pre-emptive tests in the grammar that ap-

proximate the syntactic depth with the depth of constituents in the generator input.

Sentence complexity: There is no built-in feature in a unification-based formalism to count the number of siblings under a constituent. We use FUF's `control` feature to implement this test procedurally instead of using a pure unification-based test.

Parent tag: the syntactic tags used in the Penn Treebank do not correspond one on one with those used in SURGE. As a consequence, we had to translate the tests on tags like SINV to tests on several features in SURGE (*e.g.*, features cat and mood).

7 Future Work and Conclusion

We have addressed the problem of deciding where a generator should place a circumstantial in the context of summarization systems that "re-generate" text. The method is based on a corpus-based analysis of the features that influence placement of circumstantials. The rules learned predict accurate placement of circumstantials in over 95.0% of the cases (compared to a baseline of 78.6%). These good results have been achieved with a small set of features (mainly, weight of the added constituent, sentence complexity and syntactic category of the base constituent). These features are readily available in the input of our application scenario and the rules have been implemented in our generation grammar.

In the future, we will address issues regarding ordering problem within the cluster. That is, when several circumstantials are placed inside the same slot, additional rules must be learned to determine in which order they should be placed. Our method is not directly applicable for this task because of a sparse problem data: there are not enough observed cases of clusters in the corpus to learn reliable rules about the interaction among different circumstantials within a single slot.

Second, we plan to identify additional features that could have an impact on the placement of circumstantials: we have empirically hypothesized that the specificity of a geographical location seems to be very predictive of its placement (less specific locations appear first). We expect the semantic subtype of temporal modifiers (frequency, duration, location) to also have

an impact on the relative position within clusters. Our objective is to learn good predictors (lexical) for this subtype.

References

[1] R. Barzilay, K. McKeown, and M. Elhadad. Information fusion in the context of multi-document summarization. In *Proceeding of ACL'99*, Maryland, June 1999. ACL.

[2] Michael Collins. A new statistical parser based on bigram lexical dependencies. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, Santa Cruz, California, 1996.

[3] M. Elhadad and J. Robin. Controlling content realization with functional unification grammars. In R. Dale, H. Hovy, D. Roesner, and O. Stock, editors, *Aspects of Automated Natural Language Generation*, pages 89–104. Springler Verlag, 1992.

[4] M. Elhadad and J. Robin. An overview of SURGE: a re-usable comprehensive syntactic realization component. In *INLG'96*, Brighton, UK, 1996.

[5] H. Hasselgard. Time and space adverbials in english and in norwegian, with special reference to initial position. *Norsk Lingvistisk Tidsskrift*, 15, 1997.

[6] J.A. Hawkins. *A performance theory of order and constituency.* Cambridge: Cambridge University Press, 1994.

[7] Hongyan Jing. Summary generation through intelligent cutting and pasting of the input document. PhD thesis proposal, 1999.

[8] M.P. Marcus, B. Santorini, and M.A. Marcinkiewicz. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19, 1993.

[9] Kathleen R McKeown, Judith Klavans, Vasileios Hatzivassiloglou, Regina Barzilay, and Eleazar Eskin. Towards multidocument summarization by reformulation: Progress and prospects. In *Proceedings of the*

Seventeenth National Conference on Artificial Intelligence (AAAI-99). ACL, July 1999.

[10] R. Quirk, S. Greenbaum, G. Leech, and J. Svartvik. *A comprehensive grammar of the English language*. Longman, 1985.

[11] Jacques Robin. *Revision-Based Generation of Natural Language Summaries Providing Historical Background: Corpus-Based Analysis, Design, Implementation, and Evaluation*. PhD thesis, Department of Computer Science, Columbia University, NY, 1994.

[12] J. Shaw and V. Hatzivassiloglou. Ordering among premodifiers. In *Proceedings of the 37th Annual Meeting of ACL*, Maryland, June 1999. ACL.