3-30-1978

Some Comments on Intersection and Interjunction in Junction Grammar

Alan K. Melby
SOME COMMENTS ON INTERSECTION
AND INTERJUNCTION IN JUNCTION GRAMMAR

Alan K. Melby

At the fall 1977 session of the BYU Linguistics Symposium, Michael McOmber read a paper entitled "Some Proposals for Junction Grammar." Although the proceedings of the symposium have not yet been published, Mike has made available the text of his paper and has invited responses to it. The present paper is one man's response.

THREE PROPOSALS

Mr. McOmber's major proposals are summarized at the end of his paper:

(1) "Axiomatic contradictions <transformationality>,
circular reasoning, and redundancy need to be checked for throughout all of Junction Grammar.

(2) The formalism needs to be completed, definitions made, and most of all,

(3) interjunction should be eliminated in favor of intersection."

With point (1), everyone should agree. Every linguistic theory should be checked for internal contradictions, circular reasoning, and unneeded redundancy. As Junction Grammar (JG) evolves, these obvious checks are constantly applied to it. Of course, redundancy is not always bad. Available exposition on JG is certainly not flawless, but I do not believe Mr. McOmber has pointed out any inconsistencies among well-established JG axioms. In a constantly evolving theory like Junction Grammar one does not even find all the "well-established" axioms written down in one place. I will not comment on Mr. McOmber's apparent claim that axiomatic contradictions are equivalent to transformationality because he doesn't give any arguments for it or even explain it.

Point (2), I believe, refers to the formalism of JG presented at the Spring 1974 session of this symposium in a paper by Lytle and Packard. That formalism was largely the result of an interdepartmental seminar in which I participated. I have also worked on other formalisms of Junction Grammar (e.g. Melby, 1972). Junction Grammar is a model of language that was developed intuitively, not by starting with an existing mathematical model and gradually adjusting
Melby: Some Comments on Intersection

it in a formal way. I consider this approach to be one of the strengths of Junction Grammar, but it does raise the problem of how best to formalize it. I am not completely satisfied with any past attempt, but I do believe it is worthwhile at various points during the evolution of JG to attempt a new formalism. To date, none has pretended to be a complete and accurate description of JG. Indeed, I feel it is a serious error to assume that a linguistic theory should fit into available mathematical formalisms. In fact, JG was developed because of the inadequacy of existing linguistic and mathematical models. Therefore, for example, I would not accept as valid any argumentation which assumes that Junction Grammar should be formalized strictly within the bounds of standard set theory.

This concludes the discussion of the first two proposals and brings us to the third proposal, with which I cannot agree at all.

INTERJUNCTION vs. INTERSECTION

Mr. McOmber's major proposal for JG is that "interjunction should be eliminated in favor of intersection." In other words, he proposes that "sick boy" be represented as:

```
  A
 / \_________________________
 B  C
    boy       sick
```

Instead of using the JG representation:

```
  N
 / \_________
 N  SA
  *       PA
  boy     sick
```

He objects to Figure 2 on two grounds: (1) Interjunction requires trees which do not fit the mold of standard n-ary trees. (2) The interjunction version of "sick boy" uses two more nodes than the intersection version, and he suggests that these additional nodes are redundant and unnecessary.

Mr. McOmber's diagram is certainly simpler and we have considered using a diagram of the same form. As a matter of fact, a similar proposal appears in one of Lytle's Linguistics 501 materials, "The Evolution of Junction Grammar." Here is a figure from that paper (which is listed as reference "M" in Mr. McOmber's paper):
On the same page as that figure, Lytle candidly states: "I admittedly had some qualms about using so much structure to represent a phrase with only a few words, but attempts to simplify repeatedly led to the loss of descriptive and explanatory power." Lytle then discusses some of the problems associated with the simplified diagram and ultimately rejects it. I suspect that the main problem with JG diagrams is not that they are too complex but that they are too simple to describe the incredibly complex phenomenon of natural language.

It should also be pointed out that the idea that interjunction involves intersection is not new. In fact, it can be traced back in JG all the way to 1970, as reflected in the title of Daryl Gibb's master's thesis: "An Application to Mechanical Translation of a Recursive Algorithm Based on the Operations of Union and Intersection." (Italics added.)

A NEW PRESENTATION OF AN OLD PROPOSAL

Let us summarize what we have discussed so far. McOmber proposes that the Junction Grammar diagram for relative modifiers be simplified to show only an intersection. We have seen that interjunction has long been recognized to involve intersection and that even the form of Mr. McOmber's diagram has previously been proposed and rejected within JG. So we are dealing with a new attempt to justify an old proposal. Perhaps Mr. McOmber should have made it more clear that his proposal is not new. Nevertheless, he deserves a fair hearing in order to determine whether his paper sheds additional light on the issue of modification or points out flaws in the reasoning used when the proposal was previously considered. Unfortunately, his proposal is lacking in detail, and, in my opinion, sheds no significant new light on the issue.

SOME IMPLICATIONS OF THE PROPOSAL

Before considering Mr. McOmber's arguments, let us consider a few of the implications of his proposal. In the proposed diagram:

![Diagram](image)

Figure 4.

Languages and Linguistics Symposium 1978
the intersection is between the set of boys (B) and the set of sick people (C). Unfortunately, the diagram does not provide a node for the attribute sick. Thus, if one considers the phrase "The very sick boys," there is a problem in finding a place for the quantifier "very." It clearly quantifies the sickness and not the number of persons who are sick. This indicates the flavor of the problems often encountered when one tries to simplify a diagram. The simpler diagram may appear to work in some cases but when it doesn't the simplicity may add complexity in another part of the system or destroy desired distinctions.

Another effect of the intersection diagram is to lose the SX and PX nodes that allow JG to distinguish among various levels of adverbial modification. Consider the sentence:

"Happily, the boy took his medicine." Figure 5.

In one reading, the boy was happy about taking his medicine; in another, the boy was perhaps very unhappy but the speaker was happy because the medicine helped the boy recover. In the interjunction diagram, this and many other semantic distinctions are based on whether the modifier is on the predicate level (PX) or predication level (SX). If one considers the following phrase: "The happily sick boy," he will detect the same two readings for "happily" that were discussed above. This means that the JG interjunction diagram for all modifiers provides one mechanism for certain semantic distinctions which apply across categories. (This works for prepositions too. Consider the various readings of "He flipped the coin on the table."") This approach is a result of one of the basic assumptions of Junction Grammar: syntax is one component of semantics and thus there can be no syntax/semantics dichotomy. Of course, the distinction between the two readings of "happily" can be treated as a word sense distinction and thus be made to fit into the intersection diagram, but that would be against the basic JG principle that point of modification does affect meaning. In other words, my criticism of the intersection diagram is not that it may not be useful in some model of language yet undefined; my criticism is that the intersection diagram cannot work within the basic framework of JG. I will try to clarify this point with another example. Consider the phrase "the fact that John learned." This phrase is a common example of the relative/complement ambiguity. In the relative reading, John learned a fact. In the complement reading, it is claimed to be a fact that John learned <something>. Let us examine the JG method of making this distinction.
Mr. McOmber's proposal (from traditional grammar) is as follows:

Junction Grammar relative diagram shows that the relative pronoun "that" is a fact, and it shows that the relative pronoun is also the direct object of "learned." The JG complement diagram shows that the "that" is then a complementizer of the whole embedded sentence, and it shows that the referential overlap between "fact" and the embedded sentence is total whereas in the relative reading, the overlap is only with the relative pronoun. In other words, in the phrase "the fact that he came," "that he came" is a fact; but in the phrase "the apple that he ate," "that he ate" is definitely not an apple.

The above implications (loss of the PX and SX nodes, and loss of the JG relative/complement distinction) are among the reasons that led to the rejection of this proposal long ago.

In JG, the first crucial break with TG came when it was set down as a JG axiom that ++ the diagrams for the relative and complement constructions must show explicitly what elements overlap referentially. In the intersection approach, it is unclear how these semantic distinctions are made explicit. This is a major gap in Mr. McOmber's presentation. I am not saying they cannot be made; I am simply saying that Mr. McOmber's proposal might be part of another model of language but no part of Junction Grammar.
PROCESSING SPEED

Mr. McOmber makes a brief comment at the end of his paper to the effect that reducing the number of nodes in J-trees would speed up processing of J-trees and therefore reduce computer bills. This claim is rash at best. It has often been my experience that reducing the number of nodes in a tree increases the complexity of the processing and more is lost than gained. In noun phrases, for example, the article node can be hidden in features but overall, the processing of the feature is more expensive than the processing of the extra node. In computational applications the advantage of similar processing among all types of relatives has been found to more than make up for the extra processing on the little used PA and SA nodes of an adjective modifier. In other words, overall, the extra nodes speed up and simplify processing rather than slow it down. Another assumption that Mr. McOmber seems to be making is that speed of processing is more important than ease of processing. If speed of processing were all important, computational linguists would do all their programming in machine language. This would be ridiculous because the difficulty of coding a large system would probably be insurmountable. Most computational linguists would say it is most important to get the job done first and then look for ways of doing it more efficiently, if possible. One quickly verified evidence of this is the fact that much work in computational linguistics is done in the horribly slow language called LISP.

SUMMARY OF THE PAPER UP TO THIS POINT

To this point, the present paper has dealt mainly with the implications of Mr. McOmber's proposal that interjunction be reduced to intersection. I have reviewed several assumptions of Junction Grammar and have shown that Mr. McOmber's proposal is inconsistent with them.

I will conclude this paper with a consideration of a few of Mr. McOmber's specific arguments.

J-TREES AS FORMAL OBJECTS

Before Mr. McOmber begins arguing for a simple intersection approach, he argues against interjunction on formal, not linguistic grounds. I will respond likewise.

He notes that the following binary tree works out:
Then he suggests that the interjunction version would be:

\[
\begin{array}{c}
4 \\
\hline
8 \\
\hline
5 + 3
\end{array}
\]

First, I don't see why J-trees should be expected to work for arithmetic, and second, there are interjunction versions that will work:

\[
\begin{array}{c}
17 \\
\hline
15 + 2 \\
\hline
5 * 3 + 5
\end{array}
\] or

\[
\begin{array}{c}
4 \\
\hline
8 - 2 \\
\hline
5 + 3 + 3
\end{array}
\]

But so what?

Mr. McOmber says there is a difficulty in determining the meaning of the slash symbol in an interjunction rule (e.g. N*N/ SV=N). He says there are two alternatives (syntacto-semantic operator and non-syntacto-semantic operator), but he ignores two other interpretations in which the slash is not an operator at all but a context delimiter:

(1) In a constructive approach to J-tree definition, the rule N*N/SV=N indicated that the following junctions occur all at once:

\[
N \quad \star \quad N \\
\quad + \\
SV \quad PV
\]

There is no problem of missing nodes or double based transformations unless one attempts to artificially force J-rules into the mold of standard phrase structure rules.

(2) In a node admissibility approach, the tree is already built and the rule simply accepts or rejects it depending on whether the N*N is in the context of an SV adjunction. By the way, the function "f" Mr. McOmber complains is undefined is a
general labelling function which is defined in
detail by the particular set of J-rules one uses in
a particular junction grammar. This is further
explained on the page after the one in Mr. McOm-
ber's Figure 10. For example, the N*SV=N in Den-
nis Packard's formalism would be (N*,(N1,N2)),
(SV+, (N2,PV3)), and ((N*,SV+),N2). That is, there
are two dominant nodes (N* and SV+) and a shared
topic or intersect node (N2).

This gives the structure:

```
N*    SV+
\----|----
N1    N2   PV3
```

I think that part of the confusion concerns the distinction
between constructing a J-tree (which is the concern of the
Packard paper) and compiling a J-tree once built. This com-

ciliation process has only been briefly touched upon in the
JG literature and is still in early development.

Mr. McOmber argues against some of the objections to
simplified interjunction.

DERIVATION

Mr. McOmber argues that if N* sick is a derivation
("sickness") then boy * who would be a derivation also.
This supposed inconsistency is not one all all. All deriva-
tions are full subjunctions but not all subjunctions (espe-
cially interjunctions) are derivations. Another common
example of subjunction which is not derivation is quantifi-
cation (e.g. "two boys").

INDIVIDUAL vs. CLASS REFERENCE

Mr. McOmber suggests that there is a confusion over
dominant versus terminal assignment. There is a confusion
but it is Mr. McOmber's. J-tree construction involves only
assignment of category to dominant nodes. ++ It is J-tree
compilation, an entirely separate process, which involves
assignment of referential value to various nodes, terminal
and dominant.

The objection he discusses is not motivated by confu-
sion but by the desire to maintain a parallelism between
relative clause and adjective modifiers. The "who" of "The
boy who had braces" receives a referential value during com-

pilation and so the topic of an adjective modifier is
assumed to also.
THE "PROXIMITY" PRINCIPLE

Mr. McOmber does not explain what he means by the "proximity" principle. Apparently, Mr. McOmber assumes that JG claims all referential overlap to involve a subjunction. JG does not claim this but rather the converse: ++ all subjunction involves some referential overlap. So, his "counter examples" are not counter examples at all. They merely point out other varieties of referential overlap than subjunction.

ORDER OF PROCESSING

Concerning the order of processing objection, Mr. McOmber suggests that Lytle states that intersection doesn't work. He quotes Lytle as follows:

"<intersection> actually won't do".

This is a misquote. Lytle is saying that independent evaluation of the operands of a *- or -* subjunction won't do. On the very next page, Lytle clearly states that in both cases "node 1 is assigned the intersection." What Lytle is saying is that if one can break out of the mold of standard set theoretics and consider the process of evaluating a linguistic intersection, the meaning of "others" in the cases of *- and -* can be explained in terms of order of processing. In other words, "boys who are poor" involves determining which boys are poor. Those boys which are not poor become the remainder. On the other hand, "boys who are poor" involves determining which poor people are boys. Those poor people who are not boys become the remainder. In both cases, we calculate the intersection of "boys" and "who are poor" as the value of the noun phrase but a different value for "others" depending on the order of processing. If the operands are evaluated independently and intersected by standard set theory, "others" must be calculated by some other process.

Mr. McOmber suggests that the value of "others" is calculated "with not operators taking the complement of any emphasized node/set." For the two cases discussed above, Mr. McOmber's method works as well as Lytle's (although differently). Then Mr. McOmber considers two other cases: (1) "the boys who are poor..." (both stressed) and (2) "the boys need money but the others don't" (no relative clause). I am not at all sure that in (1) the meaning of "others" is well-defined. At any rate, it probably does not mean B' W' as Mr. McOmber suggests. His interpretation would exclude rich boys from "others". That is, "rich boys" are not among the "others" who don't need money. Does this mean rich boys do need money (a contradiction)? As for (2), in which there is no relative clause, the value of "others" is calculated
by default as the universe of discourse minus the computed value of the noun referment (U-B), i.e. everyone except the boys. (See the \( r+r' = U \) section in \textit{A Grammar of Subordinate Structures in English}, Lytle, 1974.)

Mr. McOmber's system also works for (2). However, Mr. McOmber does not consider the case where neither operand is stressed ("the boys who are poor need money but the others don't.") In JG, the relative clause uses \( =* \), which generates no remainder. Then the same default calculation applies as for "boys need money...". This "others" is everyone except poor boys (U - (B W)). However, Mr. McOmber's method would seem to set "others" to B W ("poor boys") since there are no stressed words and thus no not operators. This is clearly false but Mr. McOmber neglects to explain how "others" should be evaluated in this case.

**SUMMARY**

In summary I must reject Mr. McOmber's proposal to eliminate interjunction because it is inconsistent with basic assumptions of Junction Grammar. Furthermore, I must conclude that his claims of inconsistency among the axioms of JG are unfounded because his arguments either (1) are based on misunderstandings of the axioms of JG, or (2) fail to consider all alternatives or (3) attempt to force JG into some inappropriate mathematical mold.
REFERENCES


