A DEPENDENCY SYNTAX FOR
THE SURFACE STRUCTURE OF SENTENCES

Leonard Laske: Syracuse University

Abstract
The paper deals with a dependency-based formalism for representing the structure of natural language. The dependency syntax was first introduced by DL in 1963 and the paper is an extension of that work. The present paper is an attempt to provide a formalism for the surface structure of sentences, by extending the dependency formalism of TL to include the surface structure of sentences. The paper introduces the concept of a dependency tree, and the syntax of the dependency tree is described. The syntax of the dependency tree is then used to describe the surface structure of sentences.

Key words: natural language, syntax, dependency and constituent grammar, generating problem.

1 Introduction
The concept of "parse" has been introduced by many researchers, for example, the PDP and the SRL systems. Each of these systems is based on a different formalism. The present paper extends this work by introducing a new formalism, called the "dependency formalism," which is based on the concept of a dependency tree.

In dependency formalism, the structure of the sentence is represented by a tree, where each node represents a word or phrase, and the edges represent the dependencies between the nodes.

2 The Dependency Structure

2.1 The formation

The paper deals with the basic modules of the grammar in a dependency framework. The first section defines a dependency structure, and the second section describes the rules for generating sentences in a dependency framework. The third section describes the application of the grammar to a particular sentence, and the fourth section describes the limitations of the grammar.

2.2 The dependency tree

The dependency tree is a formalism for representing the structure of a sentence. The dependency tree is a directed graph, where each node represents a word or phrase, and each edge represents a dependency between the nodes.

2.3 Other formalisms

The dependency formalism is not the only formalism that has been used for representing the structure of a sentence. Other formalisms include the constituent grammar, the phrase structure grammar, and the dependency-based grammar. The constituent grammar represents the structure of a sentence as a tree, where each node represents a word or phrase, and the edges represent the dependencies between the nodes.

2.4 Generating problem

The generating problem is the problem of generating sentences that conform to the grammar. The generating problem is solved by using a set of rules, called the generating rules, which are used to generate sentences from the grammar.

2.5 Conclusion

The paper has introduced a new formalism for representing the structure of sentences, called the dependency formalism. The dependency formalism is based on the concept of a dependency tree, and it is used to represent the surface structure of sentences. The dependency formalism is a powerful tool for representing the structure of sentences, and it has many applications in the field of natural language processing.
While qualifying adjectives may add to a word's meaning, it is important to note that they may also introduce a nuance. For example:

- *inao* vs. *inao* [adjective]
- *inao* vs. *inao* (adjective)

The difference in meaning can be significant, as each term carries with it a different connotation. In the former case, the adjective modifies the noun, while in the latter, the noun modifies the adjective. This distinction is crucial in understanding the context in which each word is used.

Each term is followed by an example sentence to illustrate its usage:

- *inao* vs. *inao* (adjective) [example sentence]
- *inao* vs. *inao* (adjective) [example sentence]

In the context of these examples, the difference in meaning is highlighted, providing a clearer understanding of each term's role.

For further reading, please refer to the supplemental material provided.
In the image, there is a page of text that appears to be discussing a technical or scientific topic. The text is relatively dense and uses technical jargon. The content seems to be part of a larger document, possibly a research paper or a textbook. The page contains paragraphs with complex sentences, and there are references to tables or diagrams that are not visible in the image. The text discusses a topic that involves technical analysis or experimental results, but the specific content is not clearly visible due to the formatting and resolution of the image.
### Table 1. The Immediate Derivatives Table

<table>
<thead>
<tr>
<th>Val</th>
<th>Base</th>
<th>Prep</th>
<th>Adj</th>
<th>Chert</th>
</tr>
</thead>
<tbody>
<tr>
<td>(head)</td>
<td>(head)</td>
<td>(head)</td>
<td>(head)</td>
<td>(head)</td>
</tr>
<tr>
<td>Noun</td>
<td>(noun)</td>
<td>(noun)</td>
<td>(noun)</td>
<td>(noun)</td>
</tr>
<tr>
<td>Vb</td>
<td>(vb)</td>
<td>(vb)</td>
<td>(vb)</td>
<td>(vb)</td>
</tr>
<tr>
<td>Adv</td>
<td>(adv)</td>
<td>(adv)</td>
<td>(adv)</td>
<td>(adv)</td>
</tr>
</tbody>
</table>

### Table 2. The Long-Distance Table

<table>
<thead>
<tr>
<th>Val</th>
<th>Base</th>
<th>Prep</th>
<th>Adj</th>
<th>Chert</th>
</tr>
</thead>
<tbody>
<tr>
<td>(head)</td>
<td>(head)</td>
<td>(head)</td>
<td>(head)</td>
<td>(head)</td>
</tr>
<tr>
<td>Noun</td>
<td>(noun)</td>
<td>(noun)</td>
<td>(noun)</td>
<td>(noun)</td>
</tr>
<tr>
<td>Vb</td>
<td>(vb)</td>
<td>(vb)</td>
<td>(vb)</td>
<td>(vb)</td>
</tr>
<tr>
<td>Adv</td>
<td>(adv)</td>
<td>(adv)</td>
<td>(adv)</td>
<td>(adv)</td>
</tr>
</tbody>
</table>

Fig. 3. The Immediate Derivatives Table.

Table 1, as you can see, shows the immediate derivatives that are formed directly from the base word. These include nouns, verbs, adjectives, and adverbs. Each column represents a different type of derivative:

- **Val** represents the value of the word.
- **Base** shows the base form of the word.
- **Prep** indicates the prepositional form.
- **Adj** denotes the adjectival form.
- **Chert** signifies the chert form.

These forms are derived from the base word and can be used in sentence construction. For example, a verb can be transformed into a noun, an adjective, or an adverb through these derivational processes.

Fig. 4. The Long-Distance Table.

Table 2, on the other hand, illustrates long-distance derivatives. These are formed through more complex processes and involve a greater degree of transformation. Similar to Table 1, each column represents a different derivational type:

- **Val** represents the value of the word.
- **Base** shows the base form of the word.
- **Prep** indicates the prepositional form.
- **Adj** denotes the adjectival form.
- **Chert** signifies the chert form.

These long-distance derivatives typically involve more syntactic and semantic changes compared to the immediate derivatives. For instance, a verb might be transformed into a noun through a process involving the addition of a prepositional phrase.
The framework introduced here is the basis for an automatic system of language comprehension. In the first half, we have explored a model of syntactic dependency which underlies the semantic representation of a sentence. The model is based on the idea that the syntactic structure of a sentence can be represented by a network of nodes and edges, where nodes represent words and edges represent grammatical relationships between them.

4 Conclusions

We have presented a dependency grammar that captures some basic phenomena of the English language. The grammar is represented via EBNF rules that are the basis for formal linguistic analysis and computer programs.

Many phenomena, as well as the ambiguity of grammatical relations to the dependencies in the dependency tree in the interpretation of language, are possible to be analyzed by the syntactic dependencies and their patterns.

Many phenomena, as well as the ambiguity of grammatical relations to the dependencies in the dependency tree in the interpretation of language, are possible to be analyzed by the syntactic dependencies and their patterns.