Chapter 4

Experience with an Easily Computed Metric for Ranking Alternative Parses

George Heidorn

Abstract

This chapter describes a metric that can be easily computed during either bottom-up or top-down construction of a parse tree for ranking the desirability of alternative parses. In its simplest form, the metric tends to prefer trees in which constituents are pushed as far down as possible, but by appropriate modification of a constant in the formula other behavior can be obtained also. The chapter includes an introduction to the Epistle system developed at IBM Research and a discussion of the results of using this metric with that system. (Epistle has evolved into the Critique text-critiquing system.)
4.1 Introduction

Heidorn 1976 described a technique for computing a number for each node during the bottom-up construction of a parse tree, such that a node with a smaller number is to be preferred to a node with a larger number covering the same portion of text. At the time, this scheme was used primarily to select among competing noun phrases in queries to a program explanation system. Although it appeared to work well, it was not extensively tested. Recently, as part of our research on the Epistle system, this idea has been modified and extended to work over entire sentences and to provide for top-down computation. Also, we have done an analysis of 80 sentences with multiple parses from our database to evaluate the performance of this metric, and have found that it is producing very good results.

We begin with an introduction to the Epistle system, to set the stage for the current application of this metric. Then the metric’s computation is described, followed by a discussion of the results of the 80-sentence analysis. Finally, some comparisons are made to related work by others.

4.2 The Epistle system

In its current form, the Epistle system (Miller, Heidorn, and Jensen 1981) is intended to do critiquing of a writer’s use of English, specifically with respect to grammar and style, in business correspondence. The central component of the system is a parser for assigning grammatical structures to input sentences. This is done with PLNLP, a LISP-based natural language processing system which uses augmented phrase structure grammar (APSG) rules (Heidorn 1975) to specify how text is to be converted into a network of nodes consisting of attribute-value pairs and how such a network can be converted into text. The first process, decoding, is done in a bottom-up, parallel processing fashion, and the inverse process, encoding, is done in a top-down, serial manner. In the current application the network which is constructed is simply a decorated parse tree, rather than a meaning representation.

Because Epistle must deal with unrestricted input (both in terms of vocabulary and syntactic constructions), we are trying to see how far we can get initially with almost no semantic information. In particular, our information about words is pretty much limited to parts-of-speech that come from an online version of a standard dictionary of over 100,000 entries (Webster’s Seventh New Collegiate Dictionary), and the conditions in our 250 decoding rules are based primarily on syntactic cues. We strive for what we call a unique approximate parse for each sentence, a parse that is not necessarily semantically accurate (e.g., prepositional