Chapter XVII
Slicing and Dicing a Linguistic Data Cube

Jan H. Kroeze
University of Pretoria, South Africa

Theo J. D. Bothma
University of Pretoria, South Africa

Machdel C. Matthee
University of Pretoria, South Africa

ABSTRACT

This chapter discusses the application of some data warehousing techniques on a data cube of linguistic data. The results of various modules of clausal analysis can be stored in a three-dimensional data cube in order to facilitate on-line analytical processing of data by means of three-dimensional arrays. Slicing is such an analytical technique, which reveals various dimensions of data and their relationships to other dimensions. By using this data warehousing facility the clause cube can be viewed or manipulated to reveal, for example, phrases and clauses, syntactic structures, semantic role frames, or a two-dimensional representation of a particular clause’s multi-dimensional analysis in table format. These functionalities are illustrated by means of the Hebrew text of Genesis 1:1-2:3. The authors trust that this chapter will contribute towards efficient storage and advanced processing of linguistic data.

INTRODUCTION

This chapter suggests a way in which data warehousing concepts may be used and adapted to store and view complex sets of linguistic data. After explaining and illustrating the concept of a three-dimen-
Slicing and Dicing a Linguistic Data Cube

The clauses constituting a text can be analyzed linguistically in various ways depending on the chosen perspective of a specific researcher. These different analytical perspectives regarding a collection of clauses can be integrated into a paper-based medium as a series of two-dimensional tables, where each table represents one clause and its multi-dimensional analysis.

This concept can be explained with a simplified grammatical paradigm and a very small micro-text consisting of only three sentences (e.g. Gen. 1:1a, 4c and 5a):

- Bre$it bara elohim et ha$amayim ve’et ha’arets (in the beginning God created the heaven and the earth)
- Vayavdel elohim ben ha’or uven haxo$ex (and God separated the light and the darkness)
- Vayiqra elohim la’or yom (and God called the light day)

An interlinear multi-dimensional analysis of this text can be done as a series of tables (see Table 1).

The linguistic modules that are represented here were chosen only to illustrate the concept of an integrated structure of linguistic data, as well as the manipulation thereof, and should not be regarded as comprehensive. In analyses that are more detailed additional layers of analyses, such as morphology, transliteration and pragmatics could be added.

Although such series of tables can be regarded as a databank, if it is electronically available, these tables are not combined into a single coherent data structure and they do not allow for flexible analytical operations. Knowing the advanced ad hoc query possibilities that are facilitated by database management systems on highly structured data, the ability to perform similar operations on implicitly structured linguistic data becomes attractive. Such queries would be facilitated if all the separate tables could be combined into one complex data structure. This is an example of document processing that “needs database processing for storing and manipulating data” (Kroenke, 2004, p. 464).

The obvious suggestion for solving this problem would be to use a relational database to capture linguistic data, but there are some prohibiting factors. There are many differences among the structures of clauses and the result will be a very sparse database (containing many empty fields) if one were to create attributes for all possible syntactic and semantic fields. Even in the event that this could work, an extra field will be needed to capture the word-order position for every phrase. Furthermore, relational database management systems are restricted to two dimensions: “The table in an RDBMS can only ever represent multi-dimensional data in two dimensions” (Connolly & Begg, 2005, p. 1209).

Closer inspection of the above-mentioned two-dimensional clause tables reveals that they actually represent multi-dimensional data. The various rows of each table do not represent separate records (as is typical of a two-dimensional relational database), but deeper modules of analysis, which are related