The SQL Procedure as a Tool for Natural Language Processing

Kurt Schmidt, Research Triangle Institute

Abstract

A SAS® based system for translation of natural language data into standardized forms or codes more readily analyzed by computerized means is presented. In survey research and other areas information is frequently collected as natural language utterances. These either cannot be processed by current data processing tools or have to depend on complex and highly specialized tools. This leads to potential losses of data or high processing costs, as either the questions have to be very limiting, have to be painstakingly translated into uniform form by humans or information has to be foregone. With the advent of SAS® SQL Procedure and SAS® Macros across virtually every platform, it is now possible to implement a SAS based system that performs translation of large percentage of such free-form natural language data into standardized form readily manipulated by other computerized tools, a task previously thought of as an exclusive domain of specialized artificial intelligence tools. The system has an ability to make fuzzy translations and to be incrementally trained. The use of SAS® System as implementation platform to perform analysis makes the system not only portable, but easily maintainable by any user with access to even the simplest SAS tools.

Introduction

The SAS System is a major advancement in data processing. It allows various levels of users, from relatively computer naive to professional programmers, to manipulate data quickly and effectively with a tool that meets most data processing needs. One area of data processing, however, has been largely untouched by SAS® software applications. This area was neglected apparently because SAS System was not perceived as suitable for the application and because the area of data processing was perceived as an exclusive domain of exotic tools and techniques. This area was natural language analysis.

Real time natural language analysis is clearly the next big challenge to be met in computer interfaces and it has been the focus of attention of media and academic research. However, there is an existing application of natural language analysis that has the potential of significant savings in cost of labor, and the associated increased availability of significant amounts of difficult to obtain data, and which does not require real time response. This application is analysis of natural language responses collected during survey.

We may classify data obtained through interview surveys into two kinds of data - standardized and free-form. The standardized data are usually readily machine-analyzable. They include items such as counts, dates, phone numbers and social security numbers. Their form, and to certain extent values, are from a relatively limited set of possible values. Numbers are, of course, easiest to handle using computers. Strings such as phone numbers and Social Security numbers have certain ranges and certain well described, fairly limited forms. Even dates can be expressed in a relatively small number of ways.

The difficulty occurs when free-form data need to be prepared in a form manageable by computer. These data include items like addresses and free form English phrases. The currently frequently used method of dealing with the free-form natural language data is either to employ personnel to painstakingly analyze each response, or set of responses, and assign numerical codes to each statement based on a standardize guide, or to put such data away for later analysis, whereby the data rarely see light again. The employment of personnel to encode such responses is
itself highly imperfect. Among others, it requires significant training time, is prone to personal biases, is prone to errors due to fatigue, monotony and other sources, suffers from lack of immediate and frequently any feedback to the personnel to improve their skills, and thus can be exceedingly costly.

There is an number of computer systems designed to deal with natural language and free-form data. These systems have traditionally relied either on relatively obscure languages associated with Artificial Intelligence, or on specialized implementations in more traditional computer programming languages. Either method suffered from poor availability and significant resources required for development and maintenance (Joshi, 1991; Bethke et al., 1989; Perelman-Hall, 1992).

However, with the advent of SAS SQL Procedure and SAS Macros across platforms it is now possible to develop such systems in an uniform manner across the numerous platforms supported by SAS Software. In addition, the SAS system tools make such natural language tools maintainable and modifiable for various specific applications with ease not readily possible with more traditional approaches. Besides a simplified yet powerful approach to the challenge, this approach gives the user direct control over every key aspect of the system in a friendly and well known environment.

General approach

The system described here was designed to assist with natural language analysis. It does not represent to be able to analyze and assign every phrase that may be encountered. It does, however, strive to lessen the burden of translation. It also allows for improving of capabilities of the system, one might say learning, through feedback into its knowledge database. A feature of major importance is the fuzzy matching ability, allowing it to select the most likely match in the absence of a perfect match; further, the degree of such fuzzy matching is user solicit, allowing of ready modification of the system to meet the goals of a given analysis. Another major result of the system is repeatability and easy quantifiability of the results.

System Components

The major components of the system are a knowledge database and the matching engine. The matching engine is constant, but thanks to the power of the SAS system, which provides the underlying power, the source code is readily adaptable for special purposes. The knowledge database can consist of a stable general component and of a variable component suitable to each particular area covered by the questions eliciting the responses to be analyzed.

Knowledge Database

Initially the reference database is created by assigning to each vocabulary unit an index value indicating the degree of matching that the presence of the given vocabulary unit in a target description would indicate. This allows vocabulary units that occur in fewer target strings to have greater indicative value than units that are common throughout target strings with different translations.

Although the actual method used to build the database would be specific to the target set of the responses to be translated, an example might be indexing based on something as simple as the frequency of words in standardized descriptions. The database is build by assigning each unit, in this case word, in the standardized guide a value reversely proportional to the frequency of occurrence of the word in the guide. For example, if we have a guide that has the following words and their associated frequencies:

<table>
<thead>
<tr>
<th>Word</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baby</td>
<td>1</td>
</tr>
<tr>
<td>sitter</td>
<td>1</td>
</tr>
<tr>
<td>house</td>
<td>3</td>
</tr>
<tr>
<td>own</td>
<td>1</td>
</tr>
<tr>
<td>Mechanic</td>
<td>5</td>
</tr>
<tr>
<td>painter</td>
<td>2</td>
</tr>
<tr>
<td>portrait</td>
<td>1</td>
</tr>
</tbody>
</table>
we would initially assign the indicator values of

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baby</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sitter</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>house</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>own</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mechanic</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>painter</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>portrait</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

It is immediately clear that such simplistic assignment would not necessarily be optimal for all uses. However, it is important to remember that since the implementation is in SAS, the system would have an unprecedented degree of flexibility, giving the end user the power and convenience to modify the values as needed.

Besides the indicator values the knowledge database contains one more item, the standardized phrase translation indicator - a code corresponding to the standardized phrase which it represents. This allows, after final analysis, to determine the standardized phrase to which the free form target phrase should be translated.

Threshold value

An important feature of the system is the threshold value. The threshold value is the minimum acceptable score value. It allows for fuzzy matching of data, discriminating between "good enough" matches and those that would not be acceptable. Let us assume that we have the following description codes and corresponding descriptions:

300 - Baby sitter
301 - Baby sitter - own home
302 - Baby sitter - another private location
303 - Baby sitter - child care center
304 - Baby sitter - licensed

300 - Baby sitter
301 - Baby sitter - own home
302 - Baby sitter - another private location
303 - Baby sitter - child care center
304 - Baby sitter - licensed

Let us also assume that our index rules are as follows:

baby - 1
sitter - 4
own - 3
home - 3
another - 1
private - 2

then for a description "Baby sitter" the index score (weight) would be 5. Other methods of scoring are possible based on other theories and empirical research (Bethke et al., 1989; Knaus, 1987). If we were to set the threshold value of 6, then the code would have to be manually assigned. If we would set the threshold value of 5 or less, then the code would be machine-assigned. This important feature allows the user a significant degree of control over the accuracy of the machine assigned codes. The higher the threshold value the more accurate the coding - at the expense of potentially fewer codes assigned.

Conclusion

The presented approach to analysis of natural language using SAS System offers a flexible means of producing and maintaining the means of significant reduction of labor in translation of free-form phrases elicited in response to survey interview questions into standardized phrases, and thus codes, that make these data available for analysis. Through the advantages of SAS system over specialized application languages it is possible for even relatively inexperienced end-users to achieve significant results. Although the presented approach produced satisfactory results, it offers means of fine tuning its performance to achieve a particular goals. It is important to realize that the weighting method used and the vocabulary unit matching function applied will have the controlling influence on the quality, effectiveness and reliability of the automatic coding system.

Although the system is not likely to replace human intervention within the foreseeable future, the system's feedback feature, utilizing accepted data as source of refining its functionality, as well a feasibility of better quality control over the encoding, promise
significant advantages over systems utilizing humans alone. The use of SAS System also allows the mobility of the system across platforms with only minor modifications. This application of SAS System to an area previously considered the domain of exotic to an average person area of Artificial Intelligence is also encouraging for development of other applications either previously considered unsuitable for SAS System or simply not thought of in terms of data processing areas in which SAS System excels.

Bibliography


Kelly, E. and Stone, P.; Computer Recognition of English word senses; North-Holland Publishing Company; Amsterdam, 1975.


Acknowledgements

The author acknowledges the support of the Center for Survey Research of the Research Triangle Institute in preparation of this work. His thanks especially go to Pat Smith and Danny Allen, whose accommodation with time constrains and encouragement made this work possible; and to Dan Pratt, for his generous help with bibliographical references.

SAS® and SAS/SQL® software are registered trademarks of SAS Institute Inc. in the USA and other countries. *indicates USA registration.

Other brand and product names are registered trademarks and trademarks of their respective companies.

88
vocabulary correlation standardized unit indicator phrase code

Figure 1. Structure of knowledge database

Natural language utterance to code

Knowledge base

→ SQL ←

v

Fully translated utterances

Multiple translations

Fully translated using a single maximum significance index value

Multiple translations with same significance index values

v

Human translations

v

Knowledge database feedback

Figure 2. Translation mechanism
Figure 3. Knowledge database feedback mechanism.