An Expert System for Reviewing Commodity Substitutions in the Consumer Price Index

Frederick Conrad, Richard Kamalich, James Longacre and David Barry

US Bureau of Labor Statistics

Abstract

An expert system is being developed to help assure the quality of data in the Consumer Price Index (CPI). The system replicates the reasoning of Commodity Analysts determining if a substitute product is comparable to a product priced in the previous month. If comparable, its price can be considered in producing the CPI. The system reasons with expertise directly entered by Commodity Analysts. To help the experts (Commodity Analysts) serve as their own knowledge engineers, they are provided with a skeletal system to which they add their knowledge of particular product categories. The system was tested on six months of CPI data. Results indicate that Commodity Analysts can accurately encode their own expertise and use the technology to detect their own oversights. The success of the current project may help other members of the government statistical community collect and review survey data using similar technology.

1. Background

Major decisions in business, government, health care and agriculture are often shaped by information reported in large government surveys. One of the most influential of these is the Consumer Price Index (CPI). The CPI is conducted monthly by the Bureau of Labor Statistics (BLS) to measure the average change in price for a fixed set of goods and services. The index is updated each month by comparing the price of commodities in the current month to the price previously collected. Because of marketplace changes, a product priced in an earlier month may no longer be available. In order to maintain a continuous sample under these conditions, a comparable product may sometimes be substituted for the unavailable product. Commodity specialists review product substitutions to determine the comparability. This paper reports the results of a BLS project to introduce expert system software to the process of reviewing commodity substitution.

This paper reports the methods and results of researchundertaken by staff at the Bureau of Labor Statistics. The views expressed are those of the authors alone, and do not necessarily reflect the views of the Bureau of Labor Statistics.

Each month, BLS Field Representatives collect price information (or quotes) on approximately 78,000 products for the CPI. Approximately 2700 are substitutions for unavailable products. When the Field Representative selects a substitute product, his or her workload and training typically preclude systematic analysis of the products. Instead, the Field Representative follows a basic procedure which is designed to select a product similar to the original product. Each substitution is reviewed by a Commodity Analyst, in a central facility, to determine if the substituted product is comparable to the original one. Such a judgment requires extensive knowledge of the product domain and a meticulous comparison of the two products. To be effective, an expert system would have to replicate this factual base and reasoning capability.

An earlier research effort determined that it was feasible to develop an expert system for product substitution review [8]. The current project is aimed at developing software that realistically can be used by CPI analysts. The software is intended to help analysts evaluate their performance by highlighting any discrepancies between their actual decisions and those reached by the knowledge base they believe they use. A primary function of the project is to document and refine analyst expertise that is otherwise implicit. This makes the knowledge permanent and represents it in a way that allows it to be applied consistently.

2. Rationale for using expert system technology

One reason for applying expert system technology to product substitution review is to document a large body of tacit expertise. Encoding an analyst's commodity knowledge into production rules makes the knowledge available to the rest of the community -- even after the analyst has left that community[1]. In principle, new analysts can become more skilled by working with the expert system and studying the rule set. Once this knowledge is represented in a knowledge base, it

[1] There are currently 29 Commodity Analysts working on the Commodity and Services portion of the CPI. In the last year, four analysts left the program and three joined it. In addition to this turnover, all analysts go on vacation, during which time their review duties must be performed by others.

can form the basis of an Intelligent Tutoring System (*e.g.* [1] and [5]), creating novel training opportunities.

A second reason we are developing an expert system for this task is to exploit the reliability of computing in a situation where precision is essential, yet the opportunity for human error exists. Substitution review is prone to visual search errors as analysts locate information in the product descriptions, comprehension errors as they extract the content of those descriptions, and consistency errors when they irregularly apply a particular rule. Rare as these performance complications may be, they are a potential source of error for the CPI, and the BLS is committed to minimizing such error. Expert systems are not susceptible to these particular limitations. They seek information only where instructed to look, they interpret that information systematically, and they always apply relevant rules because their knowledge base is permanent.[2] To the extent that this technology can help reduce analyst error, it is consistent with the movement in the survey community to reduce non-sampling error [3].

Finally, expert system technology can substantially reduce the burden on analysts. Analysts report that their comparability decisions are painstaking and require on the order of ten minutes per decision. The initial plan is to provide the expert system as a post-production research tool to aid in the goals mentioned above; as such, it will temporarily increase the burden, while providing the benefits mentioned above. It is our hope that based on the success of our system, expert systems will be incorporated into production, pre-screening substitutes and eliminating the need to review many of them.

The substitutions that analysts review fall essentially into two categories. The first category comprises the vast majority of substitutions, and is relatively straightforward. The substitute product is judged either *comparable* to the unavailable product, in which case a direct price comparison is possible, or *not comparable*, in which case it is excluded from the price comparison that contributes to the Index. These decisions are routine, but they still require the analyst's full attention. The second category involves a more elaborate process. In these cases, the analyst can statistically adjust a substitute's price permitting a comparison, where a quality difference would otherwise rule this out [2]. Adjustments like these currently account for approximately 10.5% of the substitutions. These cases require additional analyst activities. The current research is aimed primarily at providing assistance for the class of substitutions which are either comparable or not comparable because they are procedurally simpler than adjustments which, because of their frequency, demand far more analyst time.

3. Commodity substitution review task

When a field representative cannot collect a price for a product that was priced in the previous month, he or she may substitute the price of a similar product. The Commodity Analyst (CA), must decide if the substituted and original products are sufficiently similar that their prices can be directly compared. In order to reach a conclusion about the two products' similarity, the analyst must compare the products on a feature-by-feature basis. These features are enumerated for each product in a document referred to as a "checklist." For some products the checklist includes over 50 features -- or "specifications." Additionally, each possible specification is given a numerical code.

```
Type of Entertainment
    S1     National/International
    S2     Local
    S99    Other
```

In this example from the Theater Admissions category, the specification "Type of Entertainment" is symbolized by *S*, and a value *1* indicates that the particular event is either national or international, *2* indicates that it is local, and *99* stands for "other"; "other" values include a blank where the field representative can insert the specific information in "free text," *i.e.* an uncoded response. The field representative collects information about all of the specifications on the checklist. It is by comparing the value for each specification for the two products that the analyst determines their comparability.

In determining comparability, the analyst must reason about complex patterns of specification data. For certain specifications, a simple mismatch disqualifies the substitution from direct comparison. For others, the decision can be more involved. For example, the analyst might set a threshold for the number of acceptable mismatches for a group of specifications. If that threshold is exceeded, direct price comparison is rejected. Sometimes, the analysts will allow certain mismatches for a specification, but not others. For example, an analyst might treat linen and cotton as equivalent fabrics for certain garments, although they are coded distinctly on the checklist, but enforce the difference between linen and nylon. Rules of this sort are derived from detailed knowledge of what affects the quality and price of certain products. The analyst acquires this knowledge from years of studying industry publications, interacting with manufacturers and retailers, and performing statistical analysis of price data using regression models (in particular, hedonics).

The comparability judgment presupposes that there is complete and consistent information from the field about the old and new products. If there is missing information, *i.e.* if the quote is *incomplete*, a clear comparison is not immediately possible, and so the analyst must infer the missing specification, or classify the substitution as not comparable. It is possible that all the information needed for a comparability judgment is available, but that it is not internally consistent, *i.e.* the quote is *inconsistent*. For example, the field might have reported at one point in the description that a particular fish product is filet, but elsewhere in the description that it is also live. Such an inconsistency might invalidate the substitution even before the comparability of the old and new product is considered.

[2] An additional source of analyst error is misconception. Instead of incorrectly executing an appropriate procedure, the analysts may develop erroneous procedures and then perform them "accurately." An expert system cannot detect logical flaws of this sort but analysts may be more likely to detect them by having made their logic explicit in the knowledge acquisition process.

While the analysts' decision making is rooted in the characteristics of particular product domains, there is a similarity to their decision strategies that transcends any one type of product. Regardless of the type of product, analysts tend to look for evidence that products are not comparable, and only after failing to find such evidence do they accept the substitution as comparable. Such a "self-terminating" comparison process is effective because each checklist includes some specifications or groups of specifications for which a mismatch allows the analyst to immediately reject the substitution. A match, however, does not allow any conclusions until other specifications are compared.

Typically, analysts reach their conclusions on the basis of one or two inferences. The brevity of their inference chains seems to be accentuated by the use of the checklist. By enumerating a product's specifications, the checklist makes explicit those criteria that analysts believe determine a product's value. In substituting, the field representative provides a value for each specification. That knowledge is therefore applied in data collection, *i.e.*, prior to the analyst's decision making about comparability, consistency and completeness.

A practical benefit of pre-analyzing the substitution through the checklist is that it restricts the range of knowledge that must be encoded for our expert system. Yet some product information, such as brand names, cannot be comprehensively codified. Therefore the current checklist design permits the field representative to collect literal, or "free text" information, *e.g.*, the S99 specification above. This poses a challenge in developing an expert system because the identity of this information is, by definition, unknown at development time.

4. System design and development

An eight member team was assembled in the Spring and Summer of 1991 to develop a usable expert system for commodity substitution review. Four team members are Commodity Analysts; the remaining members represent the Quality Assurance and Behavioral Science Research staff in the BLS. The analysts all use SAS fairly regularly, but none are expert programmers. The rest of the team has more computing experience, though none holds a computer science degree.

The team selected 12 product categories from the roughly 350 surveyed for the CPI and set out to develop a knowledge base for each. We wanted to explore the generality of the knowledge across product categories, so we selected three categories from the apparel family (Women's Swimsuits, Women's Pants and Shorts and Women's Nightwear), three from the food area (Cereal, Fish, and Dinner), three from services (Theater Admission, Automobile Finance Charges, and Hospital and Patient Services), three that were unrelated to each other (TV, Automotive Body Work and Ski Equipment).

The knowledge bases for the 12 product categories are designed as 12 independent systems, each of which is composed of a *skeletal knowledge* base and a *domain specific* knowledge base. The skeletal knowledge base is comprised of control knowledge, *e.g.* the knowledge to advance to the next substitution for review, and default knowledge, *e.g.* the knowledge that a substitution is comparable unless proven not comparable. The skeletal knowledge is the same in all twelve systems. Domain knowledge corresponds to the analyst's expertise in determining the completeness, consistency and comparability of substitutions for particular types of products. The domain knowledge in the system for Fish is entirely distinct -- at least on the surface -- from the domain knowledge in the system for Theater Admission, and so on.

This decentralized system architecture has the advantage of keeping the individual knowledge bases relatively small. If there were a single domain knowledge base for the twelve product categories, it would be more complicated to maintain than any one of the individual knowledge bases. When our prototype is implemented for production, a single knowledge base would encompass some 350 product categories, and that would be unwieldy. While considerable progress has been made in the engineering of very large knowledge bases [3], the current design greatly simplifies the problem.

4.1. Implementation

Our expert system for reviewing product substitution is implemented with Level5 Object, a commercial development environment for rule-based systems that runs on IBM compatible PCs, under Microsoft Windows. Level5 Object offers several features that are indispensable to our particular circumstances:

1. It runs on the BLS standard platform. Our decentralized architecture demands that analysts be able to use the software at their desk as part of their overall work.

2. Because of its Graphical User Interface, Level5 Object allows developers to navigate through the environment almost entirely with the mouse. This includes producing runtime displays. End users can interact with runtime software with the mouse. This is essential because Commodity Analysts, who may have limited programming experience, serve both as developers and end users.

3. The production rule language has a simple, English-like syntax, and uses English terms. There are no special characters. Again, this is attractive for users who are not experienced programmers.

4. Level5 Object has both forward and backward inference engines. The substitution review task has certain characteristics that are typically modeled with forward reasoning, *e.g.* a large quantity of data in the form of specification values, and others conducive to backward reasoning, *e.g.* a fixed set of hypotheses about completeness, consistency and comparability. Two inference engines gives us considerable flexibility.

Overall Level5 Object has been a good choice. We decided ultimately to implement a largely backward reasoning system. After conducting protocol studies with analysts, we believe this is consistent with the strategy they actually use.

1. completeness OF quote IS incomplete
2. completeness OF quote IS complete
2.1 consistency OF quote IS inconsistent
2.2 consistency OF quote IS consistent
2.2.1 comparability OF quote IS no
2.2.2 comparability OF quote IS refer
2.2.3 comparability OF quote IS adjust

2.2.4 comparability OF quote IS yes

Table 1. Goal Hierarchy (Level5 Object Agenda)

The system's overall goal structure is presented in Table 1. The system first tries to prove the substitution is *incomplete*. If it succeeds, the substitution is marked as *incomplete* and the next quote is considered; if it fails, then the substitution is *complete* and the system tries to prove that it is *inconsistent*. If that goal is satisfied, the substitution is marked *inconsistent* and control advances to the next quote; if it fails, the substitution is *consistent*, and its comparability can be evaluated.

There are four goals associated with the comparability judgment: the system first tries to prove that the substitution is *not comparable*; barring that outcome, the system seeks evidence that the substitution requires review by the analyst and so it concludes *refer*; if it cannot prove that, it then tries to establish that the substitution can be *quality adjusted*, in which case it is referred to the analyst; if it cannot prove that, it concludes that the substitution is *comparable*. After reaching one of these conclusions about comparability, the system considers the next quote. Where the quote is considered incomplete or inconsistent, it is coded as *refer* for the comparability decision, as the analyst might be able to infer enough from extant information to determine comparability, while the expert system cannot.

We believe that the reasoning required to reach any of these decisions is made simpler and more consistent by the use of the checklist. For example, one criterion used to judge the comparability of many apparel items is their cleaning requirements. The checklist encodes: machine wash, hand wash, dry clean. The information needed to determine the cleaning requirements is also present on the checklist, *e.g.* fiber content, but the analyst does not need to infer one from the other because they are all present. That shortens the comparison process and is at the crux of our self-administered knowledge acquisition procedure.

4.2. Knowledge acquisition

As indicated earlier, an expert system is only practical for CPI production if the analysts can develop and maintain it while continuing to complete their responsibilities in the monthly production cycle. An often cited bottleneck in developing expert systems is knowledge acquisition -- the process of eliciting and encoding expert knowledge [6]. In the conventional approach to knowledge acquisition, a domain expert is intensively interviewed by a knowledge engineer who encodes the expert's knowledge in a rule-base. Often experts are not able to articulate their knowledge because it is second nature. The knowledge engineer must make an educated guess at how the expert performs the task and ask the expert to react. Eventually the knowledge engineer and expert agree on the contents of the knowledge base, though as the domain changes, additional knowledge acquisition is necessary. Clearly, this sort of activity does not fit well within the CPI production schedule.

One approach to reducing the time involved in knowledge acquisition has been to build intelligent systems that interview an expert and construct a knowledge base from the interview [7]. This approach would have been an option for us, but we were not aware of any "off the shelf" tools for building automatic knowledge acquisition systems. Rather than invest in developing such a system, we explored an alternative approach. We asked the experts to be their own knowledge engineers -- to type their knowledge directly into the system, run the resulting knowledge base, and refine it to their satisfaction.

We were fortunate in that the nature of the substitution review task permits the analysts to directly express their knowledge. Each analyst wrote English language rules for completeness, consistency and comparability, and then entered them as production rules. Their access to their own knowledge may well be attributable to the checklist approach. As suggested earlier, the checklist may simplify the reasoning sequences by "pre-applying" knowledge about quality and price, focusing the decisions on specification values. The checklist also provides an abstract vocabulary to describe the domain: the specifications are lettered, and so, for example, the analysts can refer to "the A spec" to mean the style of a swimsuit, "the B spec" to indicate size, etc. This may help make the analysts' reasoning more clear to them, by disassociating it from the content of the domain.

We attempted to reduce the programming burden on the analysts as much as possible by (1) supplying the skeletal knowledge base mentioned earlier, and (2) providing a set of rule templates. The skeletal knowledge base implements the goal structure of the system -- a sort of outline of the system's logic. Because the skeletal knowledge base, in conjunction with the inference engine, provides an overall sequence of rule execution, the analyst can enter rules without considering their position in that sequence. All that the analysts must guarantee is that the terms they use to express conclusions correspond to those used in the skeletal knowledge base, *e.g.*"comparability IS no".

To help the analysts enter such rules, they were provided with a set of rule templates. We constructed the templates by searching the first four knowledge bases for recurring types of rules. Once such rules were identified, they were stripped of references to features of the product category and specific files, and presented together in a text file. A simple template might have the following form, indicating that the "B spec" has changed between the previous and current month.

```
TEMPLATE for comparability of b spec
IF old_b <> new_b
THEN comparability OF quote IS no
```

The analysts edited the templates, specializing them to their particular product category. In deriving a rule from the template above, the analyst would indicate the spec name and the "class" containing "old_a" and "new_a" -- essentially the product category. For example, in the following rule, the spec is "style" and the "old_b" and "new_b" are members of the class dB3ec38033, which the analyst would recognize as the category of women's pants.

```
RULE for comparability of b spec style
```

```
IF old_b OF dB3ec38033 <> new_b OF dB3ec38033
THEN comparability OF quote IS no
```

One type of rule included in the templates is designed to handle free text entries. These are entries such as brand name that are not coded on the checklist but entered in their literal form. The templates contain rules that compare a free text entry to a list of possible entries. That list has been classified in some way by the analyst, for example high quality brands, and is represented as a string attribute. If the entry is found in the list, then the rule can act accordingly. If the free text entry is not found in that or other similar lists, for example, low quality brands, then the analyst must add that free text item to the appropriate list.

4.3 Data

The data used for the current project were stored in database files on individual PCs. The data were extracted from a mainframe database, generated each month in producing the CPI. Specification codes for each substitution and original product, as well as an analyst's comparability judgment were transferred to the database files on the individual machines.

5. Evaluation

The system is being formally evaluated at this time. The evaluation procedure compares the performance of the analyst and the system for a given knowledge base, run on three months of data. The statistics of interest are the proportion of overall agreement for each possible decision: complete/incomplete, consistent/inconsistent, comparable/not comparable/ quality adjust/refer. Once an acceptable level of performance has been attained, the knowledge base is run again on three more months of data, one month at a time. For each additional month, statistics are calculated for the performance of the current knowledge base using the new data. Then, any additional knowledge engineering suggested by the disagreements is performed and final statistics for that iteration are calculated.

We present results below from one of the apparel categories, Women's Pants and Shorts. This is representative of our sample, though several other knowledge bases were tested with considerably smaller data sets..

5.1. Agreement, disagreement and referral

Over the four data sets, the system and analyst agreed on 68% of the trials and disagreed 13% of the time; the system referred 19% of the decisions back to the analyst[3].

An important issue is the coverage of the initial knowledge base for new data sets and the amount of modification needed to maintain a useful set of rules. In short, little was added to the knowledge base beside additional brand names after the initial run on each new data set.

It is worth asking how much exact repetition of brand names occurs across data sets in order to determine if our free text approach is effective. To look at this more closely we calculated the proportion of new brand names to possible brand names (two for each substitution) for the second through fourth iterations. These proportions are .35, .41 and .38 respectively, relatively small given the potential number of brand names. One interpretation is that the set of brand names in the original knowledge base was relatively exhaustive, and that these proportions represent a steady flow of one time only brands and variations in the field data collectors' spelling.

Figure 1 plots agreement, disagreement and referral rate over the four iterations. There is a sharp drop in agreement as each new data set is introduced followed by a rebound in the first three iterations after new brands have been added. The rebound in the fourth iteration is negligible, due to the exceptionally high referral rates. The pattern of referral rates is essentially the inverse of agreement rates, indicating that when the system makes a substantive decision (Yes, No, Adjust), it tends to concur with the analyst.

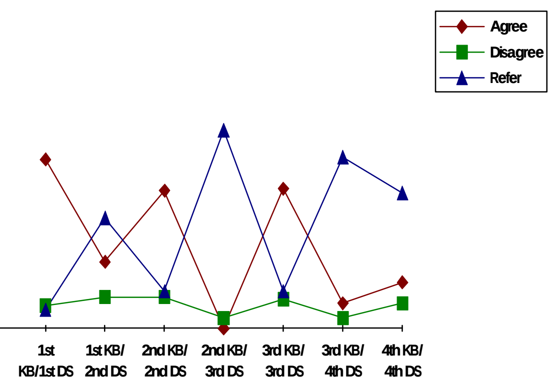

Figure 1. Agreement, disagreement and referral rates at start and completion of each iteration. KB = Knowledge Base, DS = Data Set.

5.2. Substantive Decisions

If we focus only on the substantive decisions, *i.e. Yes*, *No*, *Adjust*, the correlation between the system's and analyst's decisions over the four iterations is .60, p < .0001 (N = 294). The correlations for the first three iterations were .79, .13, and .73. The correlations for the fourth data set could not be calculated due to empty cells. The correlations for Iterations One and Three were significant; for Iteration two, the correlation was small and non-significant. This appeared to be due to a large number of substitutions judged comparable by the analyst but which the system concluded should be adjusted.

Iter-ation	Analyst Decision	System Decision Yes	No	Adjust	N
	Yes	.71	.01	.01	

[3]These results are based only on quotes judged by the system to be both complete and consistent. In all, 92% of the quotes were so evaluated. Two percent were judged inconsistent and seven percent were judged incomplete.

1	No	.05	.01	.02	147
	Adjust	.02	0	.16	
2	Yes	.80	.01	.16	80
	No	0	0	0	
	Adjust	.01	0	.01	
3	Yes	.66	.02	.06	53
	No	.06	0	.04	
	Adjust	0	0	.17	
4	Yes	.64	.07	.28	14
	No	-	-	-	
	Adjust	-	-	-	

Table 2. Analyst-System correspondence for four development iterations.

Table 2 presents rates of correspondence for all combinations of analyst-system judgment, for each of the four development iterations. The main diagonal in the 3 X 3 panel for each iteration contains agreement rates; the off-diagonal cells contain disagreement rates. A high degree of correspondence would appear as large numbers along the main diagonal and numbers close to zero in the off-diagonal cells. Iterations 1 and 3 approach this pattern, but the row for analyst No judgments diverges, showing small numbers in all cells, with the smallest in the diagonal cell. For the second and fourth iterations, there are simply no observations in this row. Our ability to evaluate the system's accuracy for No judgments is compromised by their infrequent occurrence.

5.3. Discrepancies

Not all discrepancies indicate inaccuracy. The analyst classified the discrepancies from the first iteration and from a preliminary run in the second iteration. For the first iteration he identified six discrepancies (out of 17) that were due to (1) analyst error; (2) possible field messages to the analyst -- not available to the system; and (3) peculiar price changes that seemed too rare to warrant special rules. All of these discrepancies involved NO and PP decisions by the analyst, helping to explain the apparent inaccuracy of the system for those items.

For a preliminary run through the second data set, 15 out of 19 discrepancies were attributed to analyst error, and one to a field error that the system, unlike the analyst, could not accommodate. All of these discrepancies concerned analyst Yes judgments. Similar analyses of the later data were not performed. because the Commodity Analyst left the program.

We have encoded the expertise of an analyst for this apparel domain without a professional knowledge engineer. Yet the accuracy of the resulting knowledge base is good. The overall correlation between analyst and system is positive and significant. The major weakness of the study is our inability to evaluate the accuracy of "No" decisions. There are relatively few of these in practice, and a preliminary analysis suggests that those which do occur are prone to analyst oversight.

6. Future

We are optimistic that the commodity substitution review expert system will be valuable to CPI Commodity Analysts. One extension to its functionality that appears to be tractable is to encode the expertise used for Quality Adjustment. Currently the system is equipped with rules to notify an analyst when it encounters a candidate for adjustment. In the proposed extension, the system would estimate the new price, much as the analyst does, and present the estimate to the analyst for review. A further use of the technology is to adapt it for data collection. The CPI field operation is currently experimenting with entering quotes into portable, pen pad computers [9]. Embedding analyst expertise in such systems, might enable field representatives to screen potential substitutions, increasing comparability rates.

Finally, our approach may help to automate a broad class of tasks in which expertise is applied to coded data collected with forms. Applications of this sort exist throughout the statistical establishment. In addition, related tasks exist in many other areas. For example, mortgage applications, medical charts, motor vehicle registration forms, all decompose the information required for a decision into discrete facts that might be relevant. A knowledgeable person reasons about this information and reaches a decision, much as the Commodity Analyst does. To the extent that our approach succeeds, so other tasks of this nature may benefit from similar technology.

References

[1] Anderson, J. R., Conrad, F. G. & Corbett, A. C. (1989). Skill acquisition and the LISP Tutor. *Cognitive Science, 13,* 467 - 505.

[2] Armknecht, P. & Weyback, D. (1989). Adjustments for quality change in the U.S. Consumer Price Index. *Journal of Official Statistics, 5,* 107 - 123.

[3] Bachant, J. & Soloway, E. (1989). The engineering of XCON. *Communications of the ACM, 32,* 311 - 317.

[4] Biemer, P.P, Groves, R.M., Lyberg, L.E., Mathiowetz, N.A. & Sudman, S. (eds., 1991). *Measurement Errors in Surveys.* New York: John Wiley and Sons.

[5] Clancey, W. J. (1987). *Knowledge-based Tutoring: The GUIDON Program.* Cambridge, MA: MIT Press.

[6] Hoffman, R. R. (1987, Summer). The problem of extracting the knowledge of experts from the perspective of experimental psychology. *AI Magazine, Vol. 8*, No. 2, 53 - 67.

[7] Marcus, S. (Ed, 1988) *Automating knowledge acquisition for expert systems.* Boston: Kluwer Academic Publishers.

[8] Tschetter, M. and others (1991). Feasibility of Using an Expert System to Review Product Substitutions. Unpublished report, Bureau of Labor Statistics.

[9] Uglow, D. A. (1992). Design issues in forms-based CASIC instruments. Manuscript submitted for publication.

Acknowledgments

We would like to thank the four Commodity Analysts for participating in the project: Karen Flynn-Huff, Robert Adkins, Joseph Chelena and Jerome Watters. We are also grateful to the following colleagues for their invaluable comments on an earlier draft: Paul Armknecht, Paul Bliese, Cathryn Dippo, Marshall Reinsdorf, Stuart Scott, Marybeth Tschetter, and Rick Valiant.