DATA-DRIVEN KNOWLEDGE ENGINEERING

Anthony T.C. Cowden
Sonalysts, Inc., Waterford, CT

Dr. John J. Burns
Sonalysts, Inc., Waterford, CT

ABSTRACT

One of the challenges in developing an intelligent tutoring system (ITS) is the understanding and representation of expert performance. This representation is required to evaluate student performance and support remediation and coaching of the student. The traditional approach to gaining this understanding is through a top-down process of knowledge engineering. In this process, a knowledge engineer observes expert performance, conducts a task analysis, and interviews one or more subject matter experts. However, this process is lengthy and error prone, especially for complex tasks.

A significant cost and time mitigation step for the KE process is to create a virtual environment to support the observation of SMEs performing the task. An added benefit is the generation of detailed digital data representing a wide range of output relative to the performance of the task at hand. The Virtual Environment for Training Technologies (VETT) immersive training simulation has been developed by NAWC-TSD, and has initially been used to support the conning officer shiphandling task of underway replenishment (UNREP). Utilizing students from the Surface Warfare Officer School Command (SWOSCOLCOM), a database of UNREP performance has been developed.

Using traditional data mining techniques, it is possible to develop an understanding of system (that is, student conning a ship) performance from the data generated in the training environment. Data mining is the process of discovering relationships within the data. Depending on the approach, the data mining process itself can result in the creation of a software model of the system. By using a fuzzy logic-based approach, this process also results in a semantic representation of that performance. An additional advantage of a fuzzy logic approach is that the semantic representation can be reviewed by the knowledge engineer and the SME, and easily understood, edited, and retested. This allows for better understanding of system dynamics, as well as a much quicker review, test, and validation process.

Unlike black box approaches to data mining such as neural nets, the fuzzy expert model is eminently traceable. Not only does it report a degree of match between observed and trained performance, it allows for traceability of system operation. Implemented within an intelligent tutoring system, this allows for the remediation process to not only measure the degree to which the student deviates from expected behavior, but also to know in what specific area that deviation occurs.

ABOUT THE AUTHORS

Mr. Anthony T.C. (Tony) Cowden is a Senior Analyst at Sonalysts, Inc., working in Waterford, CT. As the founder and Manager of the Fuzzy Systems Solutions business unit, he is the Sonalysts principal analyst for the development of fuzzy logic-based solutions focused primarily on information processing (database query, data analysis, and data mining), decision support, and training applications. He also has extensive experience in software development project leadership, modeling and simulation, operations research, curriculum development, and technical training. He received his bachelor's degree from the University of Michigan, earning a commission in the Navy through the ROTC program, and is currently pursuing a Master's degree in computer science. On active duty and the Reserves he has served on five different classes of warship.

Dr. John J. Burns is a senior scientist at Sonalysts, Inc. working in Orlando, FL. He has been working closely with NAWC/TSD in research and development of shipboard training systems with a particular focus on the development of performance measurement methodologies and technologies. Most recently Dr. Burns has been involved with the development of the Shipboard Mobile Aid for Training and Evaluation (ShipMATE), a prototype handheld PC designed to support trainers in all aspects of training. Dr. Burns received his Ph.D. in psychology from the University of Massachusetts in Amherst and has 5 years of experience in team training research and development.

DATA-DRIVEN KNOWLEDGE ENGINEERING

Anthony T.C. Cowden
Sonalysts, Inc., Waterford, CT

Dr. John J. Burns
Sonalysts, Inc., Waterford, CT

BACKGROUND

The US Navy Naval Air Warfare Center, the Training Systems Division (NAWC-TSD), has developed a Virtual Environment for Training Teams (VETT) to investigate the use of virtual software environments in training. In the initial demonstration, an underway replenishment (UNREP) task, a conning officer drives a warship alongside a replenishment ship while at sea to take on fuel and stores (Crenshaw, 1975). Part of the investigation includes developing a semi-automated, data-driven approach to understanding the actions of the watchstander while developing a cognitive model of that performance.

In traditional top-down knowledge engineering, the knowledge engineer (KE) acquires an understanding of the system by watching it in operation, interviewing subject matter experts (SME), and doing some off-line analysis of data generated by the system. From these efforts, the KE builds a description of how the system operates, and from that builds a computer model of its operation. While this leads to a good (depending on the skill of the KE and the complexity of the system) understanding of the operation of the system, it is a lengthy, time-consuming, expensive, and error-prone process. Validation, correction, and re-testing of the model can also be a complex and lengthy process, and in the worse case, primarily due to the complexity of the system, it is an entirely intractable problem.

Data-driven knowledge engineering (DDKE) is the process of using data generated by a system to help in understanding the dynamics of that system. At its heart is the ability to generate semantic rules that are descriptive of the system. While these rules form the model of the system, they also allow for the KE, SME, behavorial scientist, and other analysts to understand the operation and dynamics of the system.

DDKE does not represent a total departure from traditional, top-down knowledge engineering. The significant value of DDKE is in "priming the pump" of the knowledge engineering process, giving the KE and SME a frame of reference and common representation within which to understand and model the system. In this way DDKE is able to shorten the development time of the initial system model, and dramatically shorten the time it takes to review, adjust, and field the model.

THE SYSTEM

Central to the process of data-driven knowledge engineering (DDKE) is the system that generates digital data to be used in the process. In the case of VETT, this "system" consists of the conning officer using VETT to perform some task (UNREP, pier work, harbor transit, etc.). It is important that the system be

configured to capture as much digital data as possible to support the DDKE process.

DATA GENERATION

The DDKE analysis that was conducted was limited to the alongside phase of the UNREP evolution. During the alongside phase the conning officer must keep his ship alongside a much larger ship, on a steady course and speed, at a pre-determined separation distance. Data was collected from 19 conning officers who used the VETT testbed to conn (drive) a simulated Navy cruiser alongside an oiler.

DATA PREPARATION

Data preparation is a very important part of the process, and certainly the most time and effort-consuming. Data preparation refers to the preparation of the data generated by the system for the rule discovery phase. Data preparation involves:

- Data cleansing – getting rid of bad, corrupted, incomplete, or out-of-date data.
- Missing data – identifying and obtaining all of the columns of data that are necessary. This may involve re-configuring the system to generate and capture additional data, deriving new data, etc.
- Data derivation – deriving new columns of data from existing columns. For example, in the UNREP study it was necessary to calculate the average relative motion over a 15-second period of time, when the data had originally been recorded at one-second intervals.
- Merging data – moving data from whatever native data storage format it is in (spreadsheet, relational database, object-oriented database, etc.) into a two-dimensional (row/column) data format.

RULE DISCOVERY (MODEL GENERATION)

The rule discovery phase involves setting up and running the rule discovery tool. There are generally two types of rule discovery, supervised and unsupervised. In unsupervised rule discovery the goal is to discover any existing relationships within the data without any preconceived ideas about what the objective function might be. In supervised rule discovery the objective function (rule consequent(s)) is known. The input data fields are selected because they are believed to have a relationship to the objective functions, and supervised rule discovery describes and quantifies the relationship between the input data fields and the consequent data fields. For example, in the UNREP study the objective functions are conning officer course and speed changes; these are the only two control actions the conning officer can take. The input fields are the length of the phone and distance (T&D) line, the direction and speed of relative motion, and the fore and aft alignment of the receiving vessel with the delivery vessel.

In human performance and training applications we are generally interested in using supervised rule discovery. This is because we usually know what actions the human can take, and we are trying to understand under what combination of conditions they take that action. To

set up the rule discovery process, metrics associated with system performance must first be identified. These metrics are based on information that is available and measurable, even in a qualitative way, by the human operator. In the case of the alongside phase of an UNREP evolution, these metrics are associated with the relative position and speed of the two ships: range, alignment, direction of relative motion (DRM), and speed of relative motion (SRM).

The next step is to quantify, even if only in qualitative terms, "good" performance in terms of these metrics. For example, we know that when perfectly on station we expect the range between the two ships to be 120 feet (220 feet if measured between the center points of the two ships). We also know that this range will vary somewhat, given the dynamics between the two ships and the actions of the wind and seas. However, we expect the conning officer to be able to maintain a separation range close to the desired figure of 220 feet. We can qualify this performance by assigning adjectives to the performance metric: "Correct" separation range, "TooClose", and "TooFar". Using the expectations of performance discussed above, we can create fuzzy sets to represent these adjectives, depicted in Figure 1.
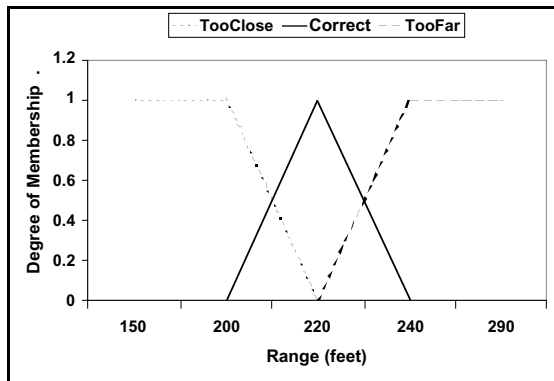
The interpretation of these fuzzy sets is as follows: The degree of membership (DoM) within the concept of being at the correct range is 1.00 (fully within the concept) when the center of the two ships are 220 feet apart (exactly at the correct separation range). As the range moves away from 220 feet, in either direction, the DoM within the concept of being at the correct range decreases. We have shown this decrease in DoM to be a linear function of decreasing/increasing range.

At a certain point, the range has moved far enough away from the perfect separation range so that it can no longer be considered to be correct to any degree. We have created complimentary fuzzy sets, "TooClose" (to the left of the "Correct" fuzzy set) and "TooFar" that overlap the Correct fuzzy set. As the range decreases, for example, the membership in the concept of being TooClose to the oiler increases as the concept of Correct separation range decreases.

Similar to the range fuzzy sets, we can construct fuzzy sets to represent the other semantic terms in the problem, as depicted in Figure 2 through Figure 4.
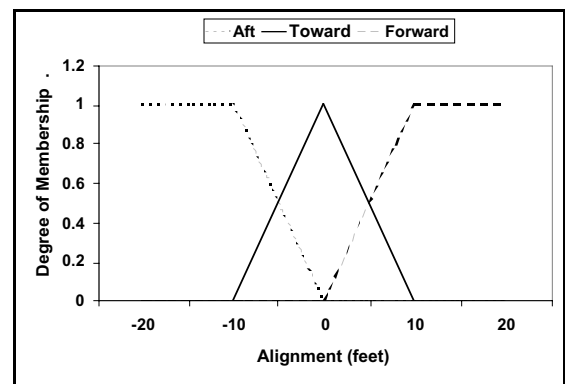

Figure 1. Range Fuzzy Sets

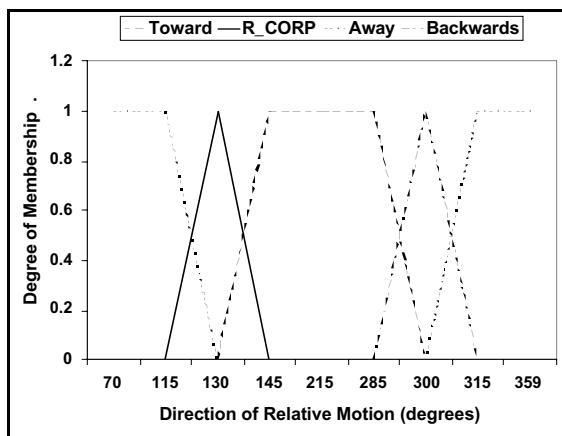

Figure 2. Alignment Fuzzy Sets

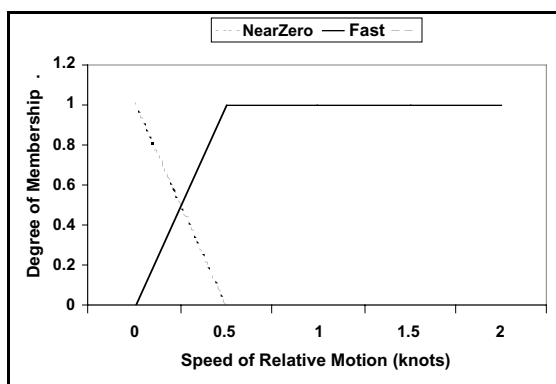**Figure 3. Direction of Relative Motion (DRM) Fuzzy Sets**

Figure 4. Speed of Relative Motion (SRM) Fuzzy Sets

The specific tool used in the rule discovery process was CubiCalc Rule Maker. CubiCalc is a commercially available fuzzy system modeling tool. Rule Maker is a rule discovery add-on to CubiCalc that supports data mining and knowledge discovery. Rule Maker uses an automated process of fuzzy interpolation and correlation analysis "…that automatically creates a fuzzy rule base from user-supplied data" (Hyperlogic, 1994, p.1).

FUZZY DATA MINING AS KNOWLEDGE DISCOVERY

The fuzzy rule base that is the output of the rule discovery process is a model of the system that generated the data. This model consists of fuzzy If-Then rules that represent the discovered relationships between the input (subjective) variables (range, alignment, DRM, and SRM) and the output (objective) variables (the course and speed changes that the conning officer makes). A fuzzy rule discovery approach to data mining, as described in this paper, is truly a knowledge discovery process, since the representation of the discovered relationships within the data consists of a semantic representation of operator performance.

A very important consideration in setting up the rule discovery process is understanding the source of the data and what the data represents. For example, in the UNREP scenario, does the data represent the performance of an experienced conning officer, or does it represent a novice? Under what conditions was the data generated (sea state, visibility, etc.)? How variable is the data? The generated model is only representative of the data analyzed.

In a fuzzy rule discovery process, the If-Then statements that make up the rule base consist of nouns and their associated descriptive adjectives. The number of rules in the rule base is a subset of the fuzzy associative matrix (FAM) that defines the possible solution space of the problem (Kosko, 1992).

The FAM consists of a permutation of all of the possible input conditions, each with an associated output condition(s). The actual number of rules in the discovered rule base is determined by the evidence displayed by the training set of data:

if an input condition is discovered in the training set, then the system learns the associated output. This rule (the relationship between the input conditions and the consequent condition(s)) is then strengthened, modified, or weakened as the rule discovery process gains experience (i.e., processes more input conditions).

The following is a sample rule as it appears in the model:

```
IF
    TD_RNG IS Correct AND
    ALGNMNT IS OK AND
    DRM IS Away AND
    SRM IS Fast
THEN
    ORD_CRSE IS Left_1 AND
    ORD_SPD IS +2RPM
```

RULE REVIEW AND MODIFICATION

The specific value of a fuzzy logic rule discovery process is that the generated model consists of rules that take the form of semantic statements. These rules are readily recognizable by the KE, SME, and behavorial scientist, and form the basis of a shared understanding of the problem domain.

In rule review and modification, the KE and SME review the rules, and with a basic understanding of fuzzy system processing, are able to verify the correctness of the rules. If necessary, the SME and KE can modify the rules. This is sometimes necessary due to the nature of the input data. For example, if the data is inconsistent or contains many errors, then the discovered rules will be inconsistent. The value of the DDKE process is that the rule discovery process will put the rule base (system model) "in the ballpark", and the SME and KE can focus their energy and expertise on understanding the modeled process

and adjusting it, rather than building it from scratch.

Another issue is missing evidence in the data. Each record in the data set represents evidence of some causal relationship between the input fuzzy sets and the objective fuzzy sets. If there is no evidence of a particular relationship in the data set, then it will not be represented by the discovered rule set. Knowledge of the structure of the fuzzy associative memory (FAM) will allow the KE to know where these instances occur, and working with the SME they can decide how to address these occurrences.

One issue is whether or not missing relationships will ever occur. If not, then the model doesn't require that rule to operate effectively. If the relationship could possibly occur, then the KE and SME should provide the missing rule. The specific quantification of the relationship (i.e., the adjectives to use in the fuzzy rule) can be determined through a combination of traditional knowledge engineering and an analysis of nearby rules within the FAM decision space.

SYSTEM MODEL

As discussed, the discovered rule base represents a model of the system that generated the data. This model is based on the relationship between independent and dependent variables in the system, as selected by the KE and SME. The model is the function that connects the independent to the dependent variables

MODEL VALIDATION

The approach to validating the model of conning officer performance was to develop a validation environment using MatLab®. MatLab has an ex-

tensive and powerful simulation capability that was used to model the dynamics of the two ships. The MatLab Fuzzy Logic Toolbox was used to implement the developed fuzzy model (MathWorks, 1999).

In the validation processing, the receiving ship was randomly positioned within an allowable starting area with a defined course and speed (Casey, 2000). The fuzzy model would control the motion of the receiving ship. The test was to see if the fuzzy model could adjust receiving ship course and speed such that the receiving ship maneuvers into the correct alongside position and stays there. In addition, the fuzzy model was evaluated on its ability to maintain the receiving ship's position alongside with minimal course and speed corrections.

An artificiality of this approach is the behavior of the model as it tries to maneuver the receiving ship towards the oiler from a "distant" position. As the receiving ship maneuvers through the geographic plane, in terms of the fuzzy model it is moving through a multidimensional (in this case, fourdimensional) decision space. The direction of relative motion (DRM) and speed of relative motion (SRM) change with changes in either course or speed, and Range and Alignment are affected by the relative change in position between the receiving ship and the oiler.

The effect of this artificiality is that as the ship maneuvers from a "distant" position to the correct on-station position, the model implements a number of course and speed changes as it passes through different regions of the decision space. In reality, an experienced conning officer would determine an initial action to take, and would make adjustments only as necessary. This is not a limitation of the

model as much as it is an application of the model for which it was not intended: the model is designed to make fine adjustments to position, not control a ship through a transit.

If the model were to be implemented operationally or in a training application, a front-end assessment should be implemented to assess the degree to which the current solution (in this case, the receiving ship course and speed) would achieve the correct result, and weigh the output of the model accordingly. One simple way would be to weight the output of the model recommendation with the inverse of a fuzzy "correct action" determination. For example, if the current solution is fully (1.0) within the set of "correct action", then the weight applied to any recommendation by the model would be the inverse of the fuzzy set "correct action" (in this case, 0.0). That way, the recommendation from the model would essentially be the null set.

The model was not expected to be able to maintain the correct alongside position perfectly. This is also true in the case of a real conning officer performing an UNREP. However, position keeping for an "expert" conning officer should be kept within the following acceptable tolerance values:

Variable	Min	Max
RANGE	200	240
RANGE Y	-15	15
SRM	0	.25

"Minimal course and speed corrections" is a little harder to define. In reality, the conning officer will use whatever number of course and speed changes are necessary to maintain the correct position. However, based on the VETT data, the subjects changed course 1.44% of the time

during the alongside phase, and they changed speed 1.73% of the time.

During the first running of the Alongside model, the model was able to maintain the receiving ship alongside the oiler well within the prescribed positional validation parameters. There was initially some underdamped performance in the model, which resulted in the ship position oscillating back and forth. In other words, while position keeping was generally well within tolerance, the model generated an excessive number of conning commands.

MODEL TUNING

Model tuning efforts were undertaken in two areas. The first was to correct the underdamped performance of the model. In general, the rules that applied to the receiving ship being nearly on station were adjusted to take less aggressive corrective action. The second class of tuning efforts focused on the defuzzification method, and is discussed in more detail in the following section.

DEFUZZIFICATION

In the processing of a fuzzy system each rule contributes its "opinion" as to the outcome of the system. These opinions are combined into a single opinion in a process known as fuzzification, and this combined opinion is defuzzified into a single, discrete system output.

There are a number of different ways to defuzzify fuzzy systems to generate model output. In general they fall into two classes: continuous and discrete defuzzification. In continuous defuzzification, the model outputs a precise value that can exist at any point along the entire range of possible output val-

ues. In discrete defuzzification, the model outputs one of a set of discrete possible output values.

In control-related applications, continuous defuzzification is generally preferred. However, the conning officer is limited to a number of specific conning command increments (i.e., two RPMs of speed, 0.5 degrees of course), which would seem to argue for a discrete defuzzification method. While the conning officer model could be considered a control application, the nature of the conning officer command limitations calls for a discrete rule output. During the validation process a number of discrete defuzzification methods were tested. In addition, associating continuous defuzzification with a rounding method to match up the fuzzy model output with the appropriate conning officer command was evaluated.

The following conclusions can be drawn from the successful validation of the Alongside model:

- The process of DDKE can provide a deep understanding of what a subject does by evaluating the data associated with subjects performing that task.
- DDKE is a relatively quick and efficient way of developing expert models of human performance.
- It is believed that, based on this successful demonstration, that the process of DDKE can be extended to other shiphandling tasks, and likely to other problem domains entirely.

MODEL USAGE IN INTELLIGENT TUTORING

One of the areas in which the fuzzy expert technology that underlies the

DDKE approach may be of great value is in its ability to evaluate student performance. It is hypothesized that the expertise resident in a fuzzy model (such as that generated by the DDKE approach) can be used to effectively measure student performance. For example, a student could be evaluated with regards to how accurately they followed some recommended action (turned in the right direction, turned the recommended amount) or how timely they responded (turned early, turned late, turned on time). While the direction of turn may be a binary value, the other parameters all incorporate a degree of fuzzy membership.

The Alongside model represents the control actions taken by an experienced conning officer to maintain a fine position alongside an oiler during an UNREP evolution. This representation is achieved through a fuzzy rule base that consists of semantic rules that process in parallel to achieve the combined result of expert performance. During this processing it is possible (depending on the specific fuzzy inference engine used) to track the individual contribution (rule firing) of each rule. Of course, the current state of the simulation (i.e., the location and relative motion of the two ships) is also known at any given time. Therefore, the model can be implemented so that we know what the model would do, and essentially why it would do it.

Implemented within a simulation-based training system, model performance could be compared against student performance to form the basis of an understanding and assessment of what the student is doing. The model knows exactly what the student should be doing and the component conditions that exist at that given instant (i.e., the reasons the student should be performing that

action). The training system could compare student action (or inaction) to the model recommendation and measure the degree to which the student's performance matches that of the model. For example, a close match might result in a positive reinforcement comment from an automated "mentor" function. An action that is constructive but not perfect might receive a recommendation for a better, more appropriate action. Finally, an action that is decidedly wrong might receive prompt instruction on the correct action to take.

CONCLUSION

The fuzzy logic rule generation approach to knowledge engineering promises to reduce the time and cost of the traditional knowledge engineering process, while allowing the knowledge engineer to retain control of the modeling process. The principle value in using fuzzy logic rule discovery for analyzing system data is its ability to generate semantic rules that are descriptive of the underlying process. Neural nets, while widely used in automated learning and system modeling applications, can not provide the same kind of descriptive and traceable output that a fuzzy rule discovery process can. In addition, they do not allow for the knowledge engineer to modify the generated model. The ability to review the generated model and edit it as necessary is often desirable, due to the difficulty in acquiring truly representative training data sets. Finally, this ability to trace system processing allows for its use in understanding and quantifying student performance and being the driver for remediation strategies in an intelligent tutoring application.

REFERENCES

Crenshaw, R.S. (1975). _Naval shiphandling_. Annapolis, MD: Naval Institute Press.

Kosko, B. (1992). _Neural networks and fuzzy systems: A dynamical systems approach to machine intelligence_. Englewood Cliffs, NJ: Prentice Hall.

Casey, S. (2000). _The alongside model validation report_. Technical Report in preparation for the Naval Air Warfare Center — Training Systems Division (NAWC-TSD) under contract #N61339-98-C-0073.

Rule Maker for CubiCalc® and CubiCalc RTC. (1994). Escondido, CA: Hyperlogic Corporation.

Fuzzy logic toolbox user s guide. (1999). Cambridge, MA: The MathWorks, Inc.