The Roof Maintenance Problem – a Fuzzy Expert System

Samuel Moyle and Mike Watts
Department of Information Science, University of Otago

Abstract

Roof maintenance is an issue that has on-going ramifications for property owners, all over the world. Determining accurately what maintenance should be taken is difficult, with often-conflicting evidence further complicating the task.

A system has been developed for use by Roof Maintenance Experts. The expert can input information about the condition of the roof then a Fuzzy Neural Network makes an assessment, returning probable roof maintenance options.

This is a non-trivial problem from the real-world domain. Often, many combinations of possible maintenance can occur and, as individual parameters change, so does the prime (or most important) option.

A fuzzy neural network system was developed for assessment, running on a handheld device that could be taken into the field by a roof maintenance expert.

Keywords: fuzzy systems, fuzzy neural network, GUI, integration, roof maintenance, expert system

1.Introduction

The application of Artificial Intelligence systems to real-life problems is not new. Many people are familiar with fuzzy controllers of lifts, automobile transmissions and air conditioners. However, these are seen as being 'high technology', only for the elite, or as a novelty.

Before widespread acceptance of Artificial Intelligence systems by mainstream computer users or system developers will occur, they must be:

- robust
- accessible, and
- seen to have more than novelty value. That is, be useful in solving a problem that would otherwise be considered a difficult problem to solve.

In addition to these issues, it is desirable to simplify a task that is technically complex.

Often the use of technology needs to be transparent or ubiquitous – users don't care about how the job is done, only that it is done. The advent of the palm-top device has meant that this type of computing power is now relatively portable, inexpensive, and accessible.

Any system developed for palm-top devices needs to be fast and compact, the aim being to provide a portable, real time, software 'tool' for daily use. In order to meet these combined conditions, a novel, real world, problem was approached.

1.1.Why a Palmtop device?

In the beginning, there was paper. Paper is useful for many reasons - it is cheap, made from renewable resources, can be recycled. People are fettered only by artistic ability. Anything that can be represented in imagery or text can be captured.

However, there are as many drawbacks. When gathering data, forms can be used, and often people using forms do not think outside the square(s). More importantly, additional tools are required in order to make the paper truly useful - a clipboard or other portable writing surface, a pen or pencil. If the user is in an out-door situation the paper can be rendered useless by weather conditions like wind or rain.

Exceptions to these situations abound - at some significant cost. Use of plastics and specialist pens or pencils were popular briefly. Waterproof notepads or folders are also common. But all are cumbersome and inconvenient in many instances.

Take for instance our intrepid roof maintenance experts. When climbing a roof, the expert has to carry a folder or clipboard, a writing implement, perhaps even a tape measure. On a windy or slightly wet day this becomes somewhat dangerous.

Now picture a different situation. The expert climbs the ladder with both hands free. Points of interest are recorded and notes written directly onto the surface of a device the same size as the experts' hand. They visit all corners of the roof, gathering point data using an integrated GPS system. This device, when finished with, can be slipped back into the pocket from whence it sprang.

Science fiction? A little Trekkie? Well, no actually. In fact, just this sort of scene is played out in many corners of the world now. People everywhere are beginning to use tools that are specialised for their work that work well, and don't pretend to do everything. In fact the basic premise behind the Palm platform is simplicity. Jeff Hawkins, the motivating force behind the Palm phenomenon, recognised two prime forces - people are smarter than computers and that a palm sized device will be treated as an extension of your desktop, not a replacement [1].

How does this relate to our friendly roofing man? Well, if a Palm sized computer has the ability to take notes, allow pictures to be drawn, can be connected to a GPS, and maybe even intelligent enough to do some of the work, then there is no reason why the small device cannot replace the increasingly dangerous and inconvenient pen and paper. When back at the office, the data collected can be sent to the desktop computer and any reports can be printed - more professional than handwritten reports!

1.2.The Problem:

Determining the condition of a roof can be difficult, especially for those not having a technical background. Deciding the most appropriate course of maintenance for a roof is dependent on a number of factors, a combination influencing the final decision. Often there is not a sufficiently high correlation for any one condition to be used, making the final decision complex.

A Fuzzy Neural Network system has been developed that returns appropriate maintenance options. This system builds on work begun in 1999 [4]. The FuNN used is that available in FuzzyCOPE 3, as developed by the University of Otago [3].

2.Method

Neural networks are excellent tools for divining rules from a data set [2], as was the case in this instance. As there was no rule-set to begin with, training of the network was imperative for the success of the project. To this end, an expert was engaged to determine expected input properties and build a suitable test data set.

Random input values were created, based on the expected inputs. The expert then assessed these inputs to give a determination of the expected output. The resulting data sets were used as the training and test values for the FuNN. Having formatted the results, training data (75% of the total data set) was passed through the FuNN. The remaining 25% was used to test the FuNN.

Once training was complete, rules were extracted from the FuNN. These rules were then formatted, coded and inserted into a prototype desktop system.

3.Expected results – maintenance options and conditions

Each roof type has different properties, maintenance requirements and wear characteristics. Possible input parameters identified by the expert are listed as in Table 1, along with the possible maintenance options.

In real life, it is possible that more than one result may be appropriate for a particular roof. Fuzzy systems have the ability to determine when more than one result may be appropriate, making this a useful technology to use for this problem.

It is also possible that some inputs may appear contradictory – another condition for which Neural Networks can be trained.

It is possible that input and result values may have a high or low degree of membership. Input values can be represented by the position of a 'slider' (refer 'Growth' or 'Wear' sliders in Figure 1). The possibility for multiple output options is also displayed in Figure 1.

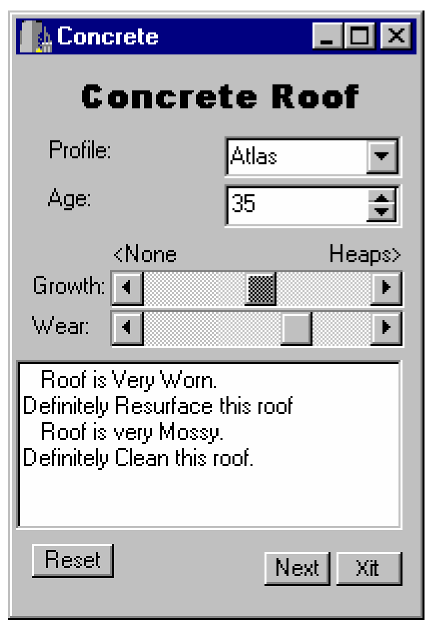

Figure 1 – Using a slider to set input values

Concrete Tile	
Input	**Result**
Wear	Do Nothing
Growth	Clean
	Re-surface
	Replace
Pressed Metal Tile	
Input	**Result**
Chip Loss	Do Nothing
Base Damage	Clean
Rust – Pans	Re-surface
Rust – Valleys	Replace
Unpainted Iron	
Input	**Result**
Surface Oxidisation	Do Nothing
Point rust – white	Re-surface
Point Rust – Red	Replace
Faded ColourSteel	
Input	**Result**
Oxidisation – Colour	Do Nothing
Oxidisation – Base	Re-surface
Rust – white	Replace
Rust – Red	

Table 1 – Possible Input and Output values. Note: Input and result possibilities are not related in any way.

Input and output values tend to fall on a non-linear continuum. For example, a roof may be described as being 'very worn' rather than 'worn'. The result could be 'Definitely

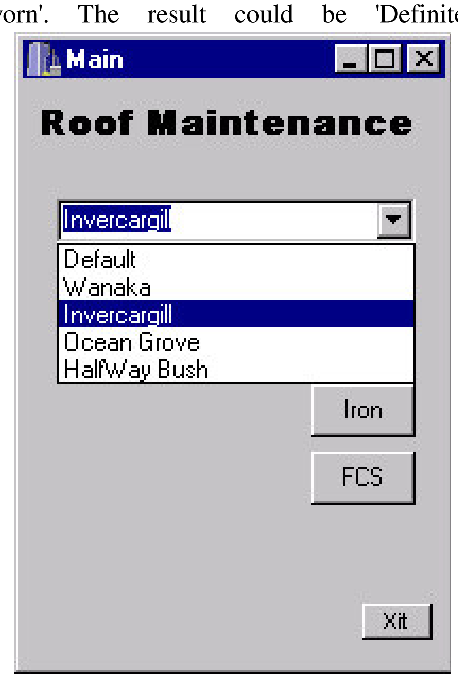

Figure 2 – Choosing a location.

Resurface the roof', falling between 'Re-surface the roof' and 'Replace the roof'. Taking time as one axis, the action of weather will affect the alternative axis (whether that be the amount of wear exhibited, amount of fungal growth etc).

This system also takes into account two other parameters – 'Location' of the roof and its 'Profile'.

Location (refer Figure 2) plays an important role in the life span of a roof. For example, a roof in Invercargill has a dramatically shorter life than the same roof in Dunedin.

The profile (Figure 3) also plays a part (although generally of lesser importance than location). There are, however, some notable exceptions that need to be taken into account.

A desktop-based prototype was built using Borland C++ builder. The prototype, in incomplete form, was then given to a group of roof maintenance experts for comment. Most of the comments received were expected given the incomplete nature of the system, but one logic error was noted. After completion, the prototype was again submitted for expert assessment, with favourable comment.

In order for a software tool to be used successfully in the Palmtop environment, suitable consideration needs to be given to:

- the small screen size (a PalmOS device is limited to 160x160 pixels),
- limited ability to input text on the handheld,
- seamlessly synchronise to a program on a desktop computer,
- be small, and
- be fast. [5]

PalmOS has been developed with the use of the tool in mind. It complements the need for

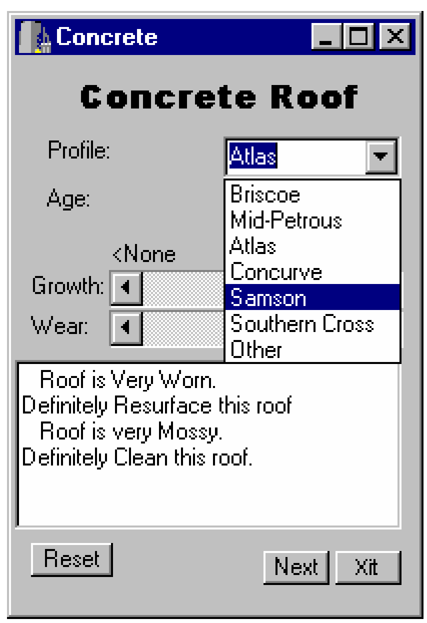

Figure 3 – Choosing the roof profile

applications that are fast to load, simple to use, and can be considered an extension of the desktop computer rather than a replacement for it.

Language and built-in control limitations also exist. This means that compact, accurate, code is necessary.

Taking these factors into account, the desktop-based prototype was used as the basis for a further prototype built for the PalmOS tool. Palm-top devices hold great promise for work improvement over time, in particular for those gathering or generating data in real time, as in this instance.

3.1.Code size limitations in the Palm Environment

A major limitation to be overcome relates to the actual memory size found in a Palm device. Many devices have 2MB of memory, while newer, more powerful, examples have 8MB or more.

In order to control the use of memory, strict code limitations have been implemented. In most instances, 32kb code chunks limit is imposed (although later devices can explicitly use larger sizes). This is implemented through a process known as 'code chunking'. This is a means by which an entire project is split into several pieces. Those components, or chunks, then pass messages between them, as described by Object Oriented design specifications.

In this project, it is expected that more powerful machines would be used, as further development work is expected to take place. However, it is prudent to maintain tight code in the interests of maximising use on any PalmOS machine.

The desktop prototype took up 4mb with 4 Neural Networks running concurrently. Considering the GUI overhead, a not inconsiderable 3.2mb, this was considered acceptable.

The PalmOS front-end resulted in two 30k code chunks, with the balance of memory allocation devoted to the Neural Network processes.

There was also a conscious decision made to limit operations to a single Neural Network at any given time. This allowed two things to be done:

- ❑ a re-use of common code meant that there was a much smaller stored memory overhead, and
- ❑ faster processing as fewer processes will be run at any given time.

The resultant code runs in as small a memory space and as quickly as possible while still giving accurate results.

4.Results

The completed prototype confirms that Artificial Intelligence systems can solve real world problems. Combined with advances in hardware, a truly useful tool was built.

As a result of this development, roof maintenance experts (most of whom are self-confessed technophobes) are now becoming more excited about the possibilities technology can bring to their work environment.

Further developments should eliminate the need for clipboards and paper having to be taken onto a roof. The integration of a GPS device would limit the need for tape measures. The option to save results and roof details in a database for the next visit is potentially very useful and deserving of further development work.

Automated pricing of maintenance options, based on results, is also possible. This can save the expert time, effort, and be more accurate with calculations.

5.Conclusions

Roof maintenance rules are not easily determined by industry experts, or comprehended by trainees. The creation of an 'automated' tool has the potential to greatly decrease the time required to train new staff members.

Ultimately, Artificial Intelligence systems can become an integral part of work processes and overall work systems. In particular, the ability to save-to and draw-from databases appeals to data integrators.

The tool developed simplifies the process involved in assessing the condition of a roof, a process currently undertaken by a roof maintenance expert. By providing useful hardware and software tools, people become used to small, fast, robust and accurate Neural Network systems. Familiarity is likely to lead to greater acceptance of all Artificial Intelligence systems by general computer users.

The creation of an Artificial Neural Network that can be implemented on a Palm-top device also opens possibilities for more complex developments in the future. This particular model, where the Neural Network is built separately then 'attached' to a front-end, means that updating the Neural Network can be done independently of front-end requirements.

References

[1] Dillon, P. "The Next Small Thing", Fast Company Magazine, issue 15, Page 97, June 1998

[2] Kasabov, N. "Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering", MIT Press, 1996.

[3] Kasabov, N., Kim, JS, Watts, M. and Gray, A. "FuNN/2 - A fuzzy neural network architecture for adaptive learning and knowledge acquisition". Information Sciences 101(3-4): 155-175 (1997)

[4] Moyle, S. "The Use of Fuzzy Systems for Small Scale Decision Support Systems", Proceedings of the ICONIP/ANZIIS/ANeural NetworkES'99 International Workshop, P241, 1999.

[5] Rhodes, Neil and McKeehan, Julie. "Palm Programming: The Developer's Guide", O'Reillys, 1999.