Artificial Intelligence

by Dan Taylor

What is Artificial Intelligence?

This article is about artificial intelligence (or AI as it is often known). I'll talk about what makes a computer system artificially intelligent and how to tell a genuinely intelligent system from one which is just "a bit clever". I'll be giving a brief introduction to a number of new and trendy AI disciplines as well as some of the older areas of the subject. I won't, however, be providing many implementation details. That shouldn't disappoint you though; AI is not a complicated subject. Any half-competent programmer could knock together a simple genetic algorithm, rule based classifier or neural network in an afternoon. Hopefully you'll read part of this article and say "I could use that", and maybe you'll even implement something.

Artificial Intelligence is one of the most researched fields in Computer Science. Thousands of coffee-hungry postgraduates all over the world are working on AI and artificially intelligent systems are operating in many aspects of modern life. Despite its ubiquity, the general public only ever really hear about artificial intelligence when Hollywood brings it to their attention through films like "I, Robot". This is quite depressing to me, as the academic community, for the most part at least, is *not* working on robots which can fight Will Smith. As a result of the media's bizarre slant on AI, the public have very strange opinions on what it actually is. Big companies have taken this on board, and as we've moved into the third millennium every product name contains the word "intelligent". Please allow me to set the record straight, before we get on with the rest of the article…

Imagine that you have just gone to the shops and bought a brand new "Intelligent Fridge". You get it home, plug it in and put your cold food in it. After a few days you start to run low on milk and as you put the carton in the bin you notice a flashing message on the front of the fridge saying, "You need more milk". You press "OK" on the fridge's touch-screen and it adds milk to your shopping list at some online supermarket. Now the manufacturers of the fridge might say that that behaviour is intelligent, but it just isn't. In fact it's no more intelligent than a thermostat with a SOAP client attached.

In order to make our fridge demonstrate real intelligence we'd have to fit it with some mechanism to predict your requirements or to make recommendations. So a fridge which said things like "Your auntie Marge is visiting this week, shall I buy extra milk?" or perhaps "Your eggs go out of date next week and I know that you like omelettes before you go to the pub. Shall I order some cheese and bacon for Friday?" would be intelligent. Though it sounds quite flippant to suggest a system which acted in that way, it really wouldn't be that hard. Of course you'd need to add a link to your calendar and scan best before dates, but neither of those things are difficult to do.

The intelligent fridge example should prove two important points about intelligence in general, firstly that there is no cut-and-dried way to define what it is that makes something intelligent and secondly that humans are incredibly good at judging what is intelligent and what isn't.

Alan Turing tried to define artificial intelligence before computers as we know them even existed. He proposed the "Turing Test", which calls for a human judge to have a text-based chat with an unknown test subject in another room. The test subject can be either a human or a computer and the judge is not aware which he is speaking to. The computer is judged to be intelligent if the judge is unable to differentiate between it and the human test subject.

The Turing test is fine, in a way, and many AI systems have passed it. However, there are many other aspects of AI that the Turing test ignores: generalisation, planning, inference and learning all spring to mind. Most researchers now agree that the aims of artificial intelligence are not to mimic human behaviour, but to solve problems.

Prediction, Classification and Planning

Most AI applications perform either classification, prediction or planning. Classifiers are, for example, used to read number plates in speed camera photos, to spot substandard products on production lines and to identify tumours in MRI scans. Prediction systems are used by the Met Office to produce the weather forecast, they're used on oil rigs to stop drill heads before they break and at call centres to predict workload. Finally, planning systems are used by mapping software to find a route between towns, the Mars rovers use planning software to navigate between rocks and game playing software (like IBM's Deep Blue) use planning techniques. In the latter half of this article we'll look at different AI techniques, each of which has a different level of suitability to classification, planning and prediction problems.

AI Techniques

Decision Trees

Decision trees are a simple, "brute-force" problem solving method and are especially suited for game playing systems. Imagine we want to write an AI player for a game of noughts and crosses. When the game starts the board is blank. We plot this state as the root node of our decision tree. The root node, in this case, has nine branches, corresponding to the nine possible opening moves in the game. If we follow one of those branches, by placing a symbol on the board, we arrive at a new node which has eight branches, corresponding to each of the eight possible moves.

Decision tree systems use a variety of algorithms to plot the entire tree for a particular problem. Every possible move which can be made in our game of noughts and crosses is mapped. The system can then asses which branch to follow from its current state by looking down the decision tree and evaluating how likely it is to win if it goes that way. The most simple way to do this is to sum the number of "win" states which exist along each branch.

IBM's "Deep Blue" chess-playing computer used a decision tree to play chess. The computer had to be absolutely vast in order to cope with the huge numbers of branches which can be followed from each state.

As well as game playing and planning problems, decision trees can be used in classification systems.

Neural Nets

Neural nets mimic the design of the human brain. Neurons are linked together by synapses in a network. Input values are presented to neurons on one side of the network and one or more outputs can be read from the other end. Due to space considerations I won't describe exactly how neural nets operate, but if you are interested you can read more on my website.

The great thing about neural nets is that they are able to learn. A set of known "training data" can be used with one of a huge variety of learning algorithms to allow the network to model the problem in hand. In an image classification problem we might require our network to accept a photo of an apple or an orange as input and output either 1 for apple or 0 for orange. We then train the network with a database of a few hundred photos of apples and oranges. After the network is trained it won't just be able to classify the photos it has seen, but should also be able to correctly classify any number of other, unseen, photos (as long as they contain an apple or an orange!).

Generalisation is the key advantage of neural nets. It allows us to take a problem we know relatively little about and train a system to be able to make predictions or classifications on it.

Evolutionary Algorithms

Evolutionary algorithms (or genetic algorithms as they are also known) are another biologically inspired artificial intelligence technique. However, unlike neural networks, which mimic the design of creatures in order to solve problems, evolutionary algorithms mimic the problem solving method which nature has employed to create life in the first place: survival of the fittest.

A simple evolutionary system for timetabling in a school would consist of a population (list) of candidate timetables and some means of assessing their quality – for example a measure of how many double bookings exist in each timetable. Using this quality measure we can "kill" the worst few timetables in our population. Deleting the worst individuals increases the average quality of the population. To replace the individuals we killed we breed some new timetables by selecting high quality parents from the remaining population and combining them (taking three days from one and four from another perhaps). We can also apply small mutations to our new individuals (moving double maths to Monday morning for example), to introduce new behaviours.

By repeating this process a few times we eventually reach a point where we can select the best individual from the population and use it to solve our problem. In the school timetable problem we might have a number of different solutions in which there are no lesson-clashes, in which case we just pick one at random.

Genetic Programming

Genetic programming is part of the broader field of evolutionary algorithms. Instead of breeding solutions, as we did in the previous section, we breed actual executable code. We create a population of functions, generally written in a high level interpreted language, and asses their quality by running them and seeing

how well the solve our problem. We can breed them together by copying “if” statements, “for” loops and function calls from the parents into the new individual. Mutations can be performed by swapping plusses for minuses and changing the order of assignments.

Genetic programming probably has fewer uses in the real world due to the huge number of problems causes by having to check syntax and make sure programs terminate properly, but many good results have been obtained using it.

Conclusion

My motivation for writing this article was to try to get people interested in using AI in their projects. Writing artificial intelligence systems is not hard and can give incredible benefits in terms of the features it allows you to offer to your customers. Due to space and time considerations I haven’t had time to talk about many of the less well known sides of AI, including Artificial Immune Systems, which are my personal field of research. If you’re interested in knowing more about AI then I really couldn’t suggest any better place to look than the internet. I’d also like to recommend the popular science book “Blondie 24” by David Fogel, which provides a better explanation of artificial intelligence than I ever could.

I hope you have found this article inspiring. If you have then I suggest you head to my website to read some of the longer articles I have written to support my talks at Developers Group meetings in the past. If you’re so inspired by what you’ve read here that you actually implement something then I’d really love to hear about it!

Dan is a part time postgraduate at the university of Reading where he researches into the detection of faults in real time systems using artificial intelligence techniques. He also works for JTL Systems Ltd, where he writes software to protect food in supermarket refrigeration systems.

Dan is spending a week walking 130-150 miles along the South West Coast Path for charity - for the MS Society. You can read more about it and sponsor Dan on http://logicalgenetics.com/longwalk.php. Please support Dan’s effort: he does a great deal for the group and it would be nice if we could repay him by everyone making a contribution, however small.

Winners of the thirteenth annual Shareware Industry Awards

- Best Overall Utility - WinZip by WinZip Computing, Inc. (www.winzip.com)
- Best Application - askSam by askSam Systems (www.asksam.com)
- Best Graphics Program or Utility - Paint Shop Pro by Jasc Software, Inc. (www.jasc.com)
- Best Desktop Enhancement - Microangelo Creation by Eclipsit Corporation (http://microangelo.us/tour)
- Best Photo Program or Utility - ThumbBuddy by Lincoln Beach Software (www.lincolnbeach.com)
- Best Application Using .NET - SourceGear Vault by SourceGear, LLC. (www.sourcegear.com)
- Best Utility Using .Net - FILEhand Search by Filehand, LLC. (www.filehand.com)
- Best Sound Program or Utility - Blaze Media Pro by Mystik Media (www.mystikmedia.com)
- Best Vertical Market Program or Utility - Help and Manual by EC Software (www.ec-software.com)
- Best Business Application or Utility - InnoSetup by Jordan Russell (www.jrsoftware.com)
- Best Educational Program or Game - Cherokee Trails by Pharos Games (www.pharosgames.com)
- Best Hobby or Personal Interest Program - Word Search Construction Kit by Insight Software Solutions (www.wordsearchkit.com)
- Best Internet Enhancement or Utility - FeedDemon by Bradbury Software (www.bradsoft.com)
- Best Internet Communication - Trillian by Cerulean Studios (www.ceruleanstudios.com)
- Best Web Enhancement or Utility - The Bat! by RITLABS (www.ritlabs.com)
- Best Non-Action Game - Jig Jag! Gold by JigJag Ltd. UK (www.jigjag.com)
- Best Action/Arcade Game - Snood by Word of Mouse Games (www.snood.com)
- Best Program or Utility for PDAs - TealDoc by Teal Point Software (www.tealpoint.com)

Working from home

a kind of blog by Joanna Pooley

It's a tribute to modern technology and to enlightened employers that it's possible for developers to combine their chosen lifestyles with their work. We have members all over the world who work from home. We now even have UK members who work from home all over the world. Apart from the, at least, two members I know of who spend part of their year working from their second homes in France, we have three current escapees. Adrian is about to work from home in Austria, Steve is dividing his time between England and Denmark. The prize goes to Martin, though, who is going back to his native Australia … without leaving his UK job.

So, I thought it might be amusing – at least for me – to have a regular column in which members could tell us about their lives. Let us bring out the Bill Bryson or the Peter Mayle in you. Tell us about a day in your life, your impressions of settling or resettling in another country, or maybe about some unusual experience.

Is it my feet?

I thought I'd kick the column off (sorry) by telling you what attractive feet I have. You may not think so, and I certainly don't, but recent experience is proving otherwise.

My desk faces the garden and a convenient little retaining wall on which the local wildlife can practice its climbing techniques, and can promenade along the top. Wendy's desk, by the way, looks over the pond onto the patio, so we have the area pretty well covered between us. I will confess that, when you phone us, you may not always have our undivided attention. But I won't bother you right now with all the birds that feed in our garden, or the occasional field mouse or other little beast that comes past. Or the blackbird that throws itself down in the sun like a bundle of rags, and sits panting with its beak open. Or the frog that appeared in the middle of our barbeque party or the hedgehog that trundled past in the shadows.

My feet seemed to start doing their Pied Piper act two weeks ago, when I had to evict a toad from under my desk. It didn't want to go – it was nice and cool under there – but I insisted. A few days later, I walked out onto the patio and nearly fell over a frog. We both jumped in surprise: in fact, it was so surprised that it jumped straight onto my foot.

Two days later, I saw a hedgehog in broad daylight. Going over to investigate, I nearly fell over its three siblings. They were tiny little things: all four would have fitted into my cupped hands. Luckily, they were already weaned, but were obviously in the process of discovering what was and wasn't edible. All our hedgehogs are called George (Poles will understand why), and I watched George IV try to eat a white feather nearly as large as himself. Desperately hoping that they wouldn't disappear as quickly as they'd arrived, I dashed inside to fetch the camera, and then settled myself on the grass for a session of Bill Oddie impersonating.

I was in luck. They took absolutely no notice of me, so I sat for an hour, taking dozens of photographs. None of which have come out very well, let me tell you. Maybe it's because of the contrast between the dark bodies and the light bristles, or because they were moving between bright sunlight and dappled shade, but my frog-loving camera just doesn't do hedgehogs. So I sat there, snapping away and even phoning my sister in Norway (as you do) to tell her about it. (She and I compete in who can see the best wildlife.) Then George IV started rolling towards me. Have you seen the way they walk, like little tanks? They're especially funny at that age, because they fall over a lot, particularly when they try to walk and scratch themselves at the same time. I wondered what would happen when he finally realised there was a large and potentially dangerous creature in front of him, but still he kept coming. Then he reached my outstretched foot. He sniffed it, he licked it – they have huge pink tongues – and then, predictably, I suppose, he tried to eat my toe. I was very brave about it.

After this unique experience, I thought I'd be lucky to see them again, but the very same thing happened at the same time the next day. This time, I was a lot bolder and sat myself much closer to them. Again, one of them (and I'm ashamed to say that no amount of Bill Oddieing can make me tell one from another) decided to investigate my foot, first trying to climb onto it, then squeezing under it, before sinking its teeth in. I wasn't quite so brave this time.

Since then, I have seen at least one or two of them on a daily basis, but they're growing fast and I don't suppose we'll have the pleasure of their company for much longer. I wonder what we'll get next?

Joanna lives and works in a village at the foot of Salisbury Plain, where bustard (yes, spelt that way!) chicks have just been reintroduced from Russia. A fully-grown bustard stands over a metre tall, so let's hope it doesn't have a foot fetish.