An Attempt to Generalize AI Part 7: A Basic, Exploratory Relevance Process

By Paul Almond

18 April 2010

Website: http://www.paul-almond.com
E-mail: info@paul-almond.com

This is the seventh in a series of articles attempting an overview of how minds may work and how similar systems could be implemented in computers. Previous articles have described a probabilistic hierarchy based on *patterns*. A pattern has a specification describing a set, or population, of *pattern instances*, distributed throughout a hierarchy containing the pattern instances of all the patterns. Each pattern's set of pattern instances is used to obtain statistical information for probabilistic predictions. Each pattern's population of pattern instances is to be described in a very general way, to provide a very general ontology. A way is needed of ensuring that the hierarchy only represents *relevant* features of the world. This requires a way of measuring the relevance of individual pattern instances in the hierarchy, or of parts of the hierarchy. The sixth article described such a measurement process. Such a measurement process is made possible by the way that the action selection process, described in the second article, requires probabilistic predictions of specific pattern instances corresponding to future evaluation function score values. This enables a measurement process in which these pattern instances are regarded as relevant, relevance then being back-propagated from them through the rest of the hierarchy. In this article, this measurement process is used in an *exploratory* process that reduces the hierarchy, by removing pattern instances, where it appears insufficiently relevant to justify its existence, while continually extending the hierarchy so that it grows into relevant regions.

Table of Contents

List of Abbreviations

AI	artificial intelligence
BERP	basic, exploratory relevance process
EFS	evaluation function score
ERP	exploratory relevance process
RMP	relevance measurement process

1 Introduction

This article is the seventh in a series about artificial intelligence (AI) and how our own minds might work. The first article, *An Attempt to Generalize AI - Part 1: The Modeling System*, is available at http://www.paul-almond.com/AI01.pdf.[1] The second article, *An Attempt to Generalize AI - Part 2: Planning and Actions*, is at http://www/paul-almond.com/AI02.pdf.[2] The third article, *An Attempt to Generalize AI - Part 3: Forgetting*, is at http://www.paul-almond.com/AI03.pdf.[3]

These three articles described a hierarchy based on *patterns*, which are sets of *pattern instances*, and were intended to give an idea of how humans may model the world, plan actions and discard information from the model when it is no longer useful. However, some way would be needed of ensuring that only *relevant* information about the world would be represented in the AI system or a human mind: Some way would be needed of ensuring that only *relevant* parts of the hierarchy would be represented.

This issue was discussed in the fourth article, *An Attempt to Generalize AI - Part 4: Modeling Efficiency*, which is at http://www.paul-almond.com/AI04.pdf.[4] It was suggested in that article that pattern instances should be allowed to have *incompletely specified pattern inputs.* The hierarchy might contain information about some of the pattern inputs to a pattern instance, while others, for practical purposes, would be non-existent. This would allow the removal of pattern instances from the hierarchy without having to remove what was "above" them, and it could simplify the connection of new pattern instances into the hierarchy. This would be done so that the hierarchy could be "pruned" by some process seeking to maximize its relevance. For this to be practical, the hierarchy needed to be *completely* probabilistic: It had previously relied on the special case of pattern instances known about with certainty (previously referred to as "fixed" pattern instances) and this reliance needed removing. This issue was dealt with in the fifth article of this series, *An Attempt to Generalize AI - Part 5: A Completely Probabilistic Hierarchy*, which is at http://www.paul-almond.com/AI05.pdf.[5]

That leaves things ready for a process to provide relevance in the hierarchy. Relevance will be provided by an *exploratory process* which extends the hierarchy, by allowing

[1] Almond, P. (2010). An Attempt to Generalize AI - Part 1: The Modeling System. *paul-almond.com*. http://www.paul-almond.com/AI01.pdf. (Also available at http://www.paul-almond.com/AI01.doc.)
[2] Almond, P. (2010). An Attempt to Generalize AI - Part 2: Planning and Actions. *paul-almond.com*. http://www.paul-almond.com/AI02.pdf. (Also available at http://www.paul-almond.com/AI02.doc.)
[3] Almond, P. (2010). An Attempt to Generalize AI - Part 3: Forgetting. *paul-almond.com*. http://www.paul-almond.com/AI03.pdf. (Also available at http://www.paul-almond.com/AI03.doc.)
[4] Almond, P. (2010). An Attempt to Generalize AI - Part 4: Modeling Efficiency. *paul-almond.com*. http://www.paul-almond.com/AI04.pdf. (Also available at http://www.paul-almond.com/AI04.doc.)
[5] Almond, P. (2010). An Attempt to Generalize AI - Part 5: A Completely Probabilistic Hierarchy. *paul-almond.com*. http://www.paul-almond.com/AI05.pdf. (Also available at http://www.paul-almond.com/AI05.doc.)

pattern instances to be added, where it appears relevant,[6] and reducing the hierarchy, by removing pattern instances, where it appears insufficiently relevant to justify its existence. In this way, the hierarchy becomes a dynamic structure, continually changing as inputs/outputs occur and relevance is sought.

Such an exploratory process requires a way of measuring the relevance already provided by any part of the actual hierarchy in an AI system, and a process to achieve this was described in the article immediately preceding this one, *An Attempt to Generalize AI – Part 6: Measuring Relevance*, which is at http://www.paul-almond.com/AI06.pdf.[7] The problem is made tractable because of the way in which the action selection process, described in the second article, *An Attempt to Generalize AI – Part 2: Planning and Actions*, works.[8] The action selection process, which is used to *drive* the system's behavior in a particular direction, relies on predictions of pattern instances which will be used for future input of a continually computed evaluation function score (EFS). The requirement for such specific predictions from the hierarchy provides a natural end-point – what the hierarchy is expected to produce – and a clear goal: The hierarchy needs to minimize the uncertainty in these particular pattern instances. This tells us what is most relevant and allows us to describe a process of back-propagation of relevance from these pattern instances, through the rest of the hierarchy, allowing the relevance of every part of the hierarchy to be measured.

This article will describe how the exploratory process, which uses such a measuring process to direct the growth and pruning of the hierarchy, can work. This, with the previous article, *An Attempt to Generalize AI – Part 6: Measuring Relevance*, will form the main description of how an AI system can be made to model relevant aspects of reality.

---

[6] This can actually be achieved just by extending the actual hierarchy equally everywhere, as will be discussed later.

[7] Almond, P. (2010). An Attempt to Generalize AI - Part 6: Measuring Relevance. *paul-almond.com*. http://www.paul-almond.com/AI06.pdf. (Also available at http://www.paul-almond.com/AI06.doc.)

[8] Almond, P. (2010). An Attempt to Generalize AI - Part 2: Planning and Actions. *paul-almond.com*. http://www.paul-almond.com/AI02.pdf. (Also available at http://www.paul-almond.com/AI02.doc.)

2 The General Ideas Behind the Basic, Exploratory Relevance Process

2.1 Starting with a Basic, Exploratory Process

A process which controls the development of the hierarchy could take a number of forms, and different degrees of sophistication are possible. I want the main idea to be clear, so to start with I will just consider a basic process, which I will call the *basic, exploratory relevance process* (BERP) – about the simplest process that illustrates the main idea of what is going on. Later, possible ways of extending this into a more sophisticated process can be described.

2.2 What Would Happen With an Unrestricted Hierarchy

The hierarchy is based on patterns. Each pattern has a set of pattern instances, and a pattern can cause its pattern instances to be "wired into" the hierarchy. If pattern instances were wired into the hierarchy in an unrestricted way, then the hierarchy would develop without any regard for relevance. The result would be a hierarchy consisting almost completely of very low-relevance pattern instances. A huge amount of computation would be required to compute the hierarchy, just to get results from the small number of pattern instances that happen to be relevant.

2.3 The Objective of the Basic, Exploratory Relevance Process

The objective of the BERP is to ensure that the scenario described above does not happen: that the hierarchy tends to consists of *relevant* pattern instances, rather than irrelevant ones.

Consideration should be given to how far this is taken. We might adopt an approach of only allowing the most relevant pattern instances in the hierarchy, so that if the hierarchy contains n pattern instances then the only way any pattern instance can remain in the hierarchy is by being one of the n most relevant pattern instances known to the system at any time and the system not encountering any pattern instance with higher relevance. A possible issue with such an approach is one of *accessibility*. There could be regions of the hierarchy with very high relevancy, but which can only be reached via intermediate pattern instances with slightly lower relevancy. If such pattern instances are not allowed, such regions could never be reached.[9]

[9] Some readers may see a similarity with the issue of accessibility in Darwinian evolution, here.

For this reason, a better approach may be one that does not always select the absolute "best" pattern instances, but instead *favors* those with higher relevance, so that the number of pattern instances allowed in the hierarchy at any time with a given degree of relevance is related to that degree of relevance, increasing as the degree of relevance increases: We would have a lot of pattern instances with high relevance and fewer pattern instances with low relevance. It should be noted that an approach of selecting only the most relevant pattern instances is really only a special case of this, in which the distribution of pattern instances over a range of relevance is made as narrow as possible.

2.4 What the Basic, Exploratory Relevance Process Will Do

The BERP will attain a relevant hierarchy in two ways, as follows.

2.4.1 Removal of Pattern Instances

The BERP will remove pattern instances from the hierarchy, with the decision to remove a pattern instance based on its relevance value. As previously stated, a good way of doing this may be to allow greater numbers of more relevant pattern instances, so that the chance that a pattern instance is removed is dependent on its relevance, with the chance increasing as relevance decreases.

2.4.2 Addition of Pattern Instances

The BERP will control the addition of new pattern instances to the hierarchy by patterns. Addition of pattern instances should be allowed more in regions of the hierarchy where the new pattern instances are expected to have high relevance. A good start is to assume that higher relevance regions of the hierarchy are the ones where more new pattern instances should be added. As with the removal of pattern instances, this should be done in a continuous way, so that new pattern instances are connected more often to higher-relevance regions of the hierarchy than to lower-relevance ones.

We do not need to go to any trouble to achieve this: The removal of pattern instances by the BERP, as just discussed, above, will actually ensure that this happens anyway, if we just allow new pattern instances to be connected to the hierarchy with equal frequency everywhere, or randomly. This is because the BERP's actions in removing pattern instances based on lack of relevancy will mean that the distribution of pattern instances in the hierarchy will be weighted towards higher-relevancy ones. If we just connect a pattern instance to the hierarchy in some randomly chosen region of the hierarchy, the distribution of pattern instances over relevancy throughout the hierarchy, caused by the removal of pattern instances, will automatically mean that it is more likely to be connected to a higher-relevance region. To put it another way, by taking out low relevance pattern instances, we are removing opportunities to connect new pattern instances to them, and therefore making new pattern instances more likely to be connected to high-relevance regions of the hierarchy.

2.5 Why add more pattern instances in higher relevance regions?

The BERP will try to get relevancy by allowing more pattern instances to be connected to parts of the hierarchy where relevancy is already high, and as just explained, on page 7,[10] the BERP's removal process will tend to ensure that this happens anyway by manipulating the distribution of pattern instances in the hierarchy. Having new pattern instances connected to higher relevance regions of the hierarchy is supposed to make it more likely that they too are highly relevant. This may seem to be assuming a lot. If a part of the hierarchy has high relevancy, why should this mean that any new pattern instances we add there will also have high relevancy? It is not being assumed that this is *always* the case. The idea is merely to increase the chances of new pattern instances having high relevancy.

Any pattern instances being connected "logically close" to ones that are already relevant will be related in some way to ones that are known to be relevant. This should make them more likely to be relevant themselves, and the more direct the relationship, the greater the chance of high relevancy.

As an example, imagine you are a detective investigating a crime. At the crime scene, a receipt was found for purchase of a book about western movies. Two new leads are offered to you, but for some reason you do not have full information about what the leads are, except they are both supposed to relate to the case in some way. You do know that one lead involves someone who rented a western movie from a store nearby, and another lead involves someone who bought a cheeseburger nearby. When you choose which lead you want to follow, you will be given full details.

All else being equal, the lead involving the person renting the western movie is more likely to *lead* somewhere useful. We might justify this by saying that it clearly involves someone who likes western movies, and we think the criminal may like western movies, but the real justification is more general to this: The better lead has the stronger relationship with what we know to be relevant – with our pre-existing structure of knowledge about the case.

This will be more obvious if we consider a more abstract example, one in which we may not have any really good, specific reasons for preferring one lead over another, but in which one lead is clearly more directly related to what we know to be relevant.

[10] Section 2.4.2: Addition of Pattern Instances.

Suppose you are a detective investigating a murder, and the victim was known to be a scholar of Latin. In the course of your investigations, you find two suspicious looking, handwritten notes. One is in Greek. The other is in Esperanto. You have to choose which note to get translated first. All else being equal, you should prefer the note in Greek. Greek is a "classical" language, as is Latin, and so has a stronger relationship with what you know to be relevant. How could it be linked to what is going on? We do not know yet. Maybe the victim wrote it: As a Latin scholar it is reasonable to think he/she may have known another classical language. Maybe someone close to the victim wrote it? A Latin scholar is likely to have associates who know Greek, maybe including personal enemies. Maybe the murderer wrote it? The point here is that we may not know what the specific significance of this note is, but most of you, reading this, would not have needed to know that. You would have immediately decided it was likely to be the more relevant piece of evidence based on its close relationship to what is already known to be relevant. All else being equal, the Latin note is more likely to be relevant because it is in a region of our model of the world that we already know to be relevant.

When we encourage addition of pattern instances in high-relevance regions of the hierarchy (and we can do this just by using the removal process to manipulate the distribution of pattern instances over relevancy within the hierarchy, we are doing the same thing: We are expecting information that relates to what we already know to be relevant to have a higher chance of relevancy than information that does not.

2.6 How the Hierarchy Should Develop

2.6.1 How the Structure of the Hierarchy Changes

The BERP modifies the hierarchy to make it more relevant by removing pattern instances. Pattern instances are selected for removal according to their relevance values, so that greater numbers of higher-relevance pattern instances are allowed to remain. The BERP also allows the addition of new pattern instances by patterns, with more new pattern instances being connected to the hierarchy where it is more relevant. The BERP does allow lower-relevance pattern instances to exist in the hierarchy: It just allows fewer of them.

The idea of this process is that the structure of the hierarchy will be continually changing. Some of this change will be due to the pattern instances themselves moving from the future to the past. A particular, future pattern instance might correspond to future input of an EFS value, so may have high relevance due to its use in the action selection process. However, this relevance comes from it being at some point in the future. As time passes, the instant to which that pattern instance corresponds will get closer to the present and some other pattern instance which is still far enough in the future will have to take its place. It will therefore lose its relevance while another pattern instance gains relevance. The BERP will be continually changing the hierarchy to "keep up", as well as to deal with changes in the kind of situation in which the AI system finds itself. All the time, the BERP is reducing it where it has low-relevance and extending it where it has most relevance.

We might also think of this in terms of *local density* of the hierarchy. The hierarchy is not absolutely prevented from going into regions that give it low-relevance, but the BERP will tend to make its density low in low-relevance regions, and high in high-relevance regions. Low-relevance regions of the hierarchy will tend to be sparse and skeletal, whereas high-relevance regions will be densely populated by a complex network of pattern instances.

2.6.2 Exploration is not random.

An important feature of all this is that the exploratory nature is not random. The hierarchy is not just being randomly extended in the hope that it finds some relevance. The hierarchy gets extended most from where it is already relevant, and this means that high relevance in the hierarchy causes more local exploration there. If that exploration finds more high-relevance pattern instances then the local density of the hierarchy will increase, in turn causing more exploration there.

More exploration, looking for high relevance, occurs "near" high-relevance pattern instances, because the hierarchy is arranged to have more of them in it. A pattern gets high relevance by being on a "high-relevance path" (actually, probably by being on many such paths) between the pattern instances corresponding to previous inputs/outputs and the important, predicted future pattern instances. When the exploratory process finds high-relevance pattern instances it is finding such high-relevance paths. When it finds such paths, the increase in local hierarchy density will cause more exploration locally, so finding high relevance paths will tend to cause more exploration where they are found. That exploration in turn may find further high-relevance paths, and so on. The whole point of this is that finding high relevancy, by affecting the local density of the hierarchy, guides the BERP to extend the hierarchy still further where it found it.

This has some similarity with the exploratory way in which the streamer of a lightning bolt obtains a high conductivity path to the ground. When the streamer encounters air with higher conductivity, it flows into there, which in turn increases the exploration for a path to the ground in that region.

One way of thinking about the low-relevance pattern instances is in terms of "tripwires". The hierarchy will spread out a network of paths featuring low density, low relevance pattern instances, and these will play little part in any of its predictions. Occasionally, something may happen that causes the relevance of some of these pattern instances to increase. This causes the density of the hierarchy to increase locally, with more exploration occurring and more paths being generated locally. This low density, low-relevance part of the hierarchy has acted as a tripwire, ready to detect something interesting impinging on that part of the world view.

2.7 Use of the Relevance Measurement Process by the Basic, Exploratory Relevance Process

The BERP is an exploratory process that extends or reduces the hierarchy based on the relevance of pattern instances in it, but it requires the pattern instances in the hierarchy to be assigned relevance values. This is done by a process of *back-propagation of relevance* through the hierarchy, which was discussed in the previous article, *An Attempt to Generalize AI – Part 6: Measuring Relevance*.[11] This relevance back-propagation process, which I will refer to now as the relevance measurement process (RMP), started with particular, bottom-level pattern instances corresponding to future inputs/outputs, and which had been assigned relevance with reference to something outside the hierarchy. Relevance was propagated back from these pattern instances through the hierarchy, tracing back along the flow of information which determined their probability values.

For this to work, the pattern instances at the start of the back-propagation process, from which all the relevance propagates, need to be assigned relevance. This must be done with reference to something outside the hierarchy. This is made possible by the way in which the system plans actions. The system's actions are selected by the action selection process described in the second article of this series, *An Attempt to Generalize AI - Part 2: Planning and Actions*.[12] In this process, an evaluation function score (EFS) is continually computed, based on recent inputs. Each EFS value is treated as if it were an external input, in that it is encoded as one of more bottom-level pattern instances: It is therefore incorporated into the historical data on which the predictions in the hierarchy are based. When an output is required, the different possible output values (e.g. 0/1) are tried by experimentally updating the corresponding pattern instance(s) as if it had occurred with each value and propagating the effects on probabilities through the hierarchy. The probabilities for one or more pattern instances that will be used for a future input of the EFS are examined, and this gives a probabilistic prediction of the EFS. This allows the merits of each possible output value to be compared, and the output made with the appropriate value.

As was explained in the second article, this is not the real planning process: *That* occurs in the hierarchy itself, when the system models its own behavior and starts to predict a pattern of improving future behavior, which becomes the context in which any individual output is assessed. The purpose of the action selection process described here is to drive the system in a given direction, by establishing a historical pattern of improvement, and to protect against random drift. Nevertheless, the action selection

[11] Almond, P. (2010). An Attempt to Generalize AI - Part 6: Measuring Relevance. *paul-almond.com*. http://www.paul-almond.com/AI06.pdf. (Also available at http://www.paul-almond.com/AI06.doc.)
[12] Almond, P. (2010). An Attempt to Generalize AI - Part 2: Planning and Actions. *paul-almond.com*. http://www.paul-almond.com/AI02.pdf. (Also available at http://www.paul-almond.com/AI02.doc.)

process provides an external reference for assigning relevance to pattern instances. We can say that those pattern instances examined in the action selection process, corresponding to predictions of future EFS inputs are the pattern instances that are relevant, and the system will be functioning best when it is able to produce probabilistic predictions for them which have minimal uncertainty. This would still apply if some more sophisticated version of the action selection process were used; for example, one using multiple EFS predictions. With different versions of the action selection process, the pattern instances to be assigned relevance externally may be selected in different ways – in some versions of it maybe even corresponding to near-future outputs – but the general principle would remain the same.

The action selection process is therefore important to the RMP in defining which pattern instances are relevant at any time. The BERP, which in turn uses the RMP, can then continually modify the structure of the hierarchy to minimize uncertainty in prediction of whatever pattern instances are considered relevant by external standards at the time.

2.8 Why the Hierarchy is Completely Probabilistic

The fourth article in this series, *An Attempt to Generalize AI - Part 4: Modeling Efficiency*, in discussing the need for measures to control the spread of the hierarchy, suggested using a completely probabilistic hierarchy, instead of the hierarchy relying on the special case of “fixed” pattern instances which had been proposed.[13] The fifth article in the series, *An Attempt to Generalize AI - Part 5: A Completely Probabilistic Hierarchy*, modified the description of the hierarchy to make it completely probabilistic, dispensing with any dependency on a special case of certainty.[14]

The hierarchy was made completely probabilistic mainly to make the kind of process discussed in this article feasible. The BERP will involve removing pattern instances from the hierarchy when they have insufficient relevance, and this is likely to lead to pattern instances with incompletely specified pattern inputs. Such pattern instances can be dealt with probabilistically, but it is likely to mean that absolute certainty in the hierarchy is rare, and cannot be relied on as an important part of the way the system works. It would also require that the removal of a single pattern instance was followed by the removal of all the pattern instances that depended, directly or indirectly on it – and there may be many such pattern instances. Making the hierarchy completely probabilistic makes it sufficiently fault tolerant to allow the BERP to modify its structure without causing failure.

---

[13] Almond, P. (2010). An Attempt to Generalize AI - Part 4: Modeling Efficiency. *paul-almond.com*. http://www.paul-almond.com/AI04.pdf. (Also available at http://www.paul-almond.com/AI04.doc.)

[14] Almond, P. (2010). An Attempt to Generalize AI - Part 5: A Completely Probabilistic Hierarchy. *paul-almond.com*. http://www.paul-almond.com/AI05.pdf. (Also available at http://www.paul-almond.com/AI05.doc.)

3 Description of the Basic, Exploratory Relevance Process

3.1 Addition of Pattern Instances

Patterns are allowed to extend the actual hierarchy by continually connecting new pattern instances to it. No particular part of the hierarchy is favored. Each pattern is expected to follow its construction specification when adding new pattern instances, but apart from this requirement, a pattern can connect a new pattern so that its pattern inputs come from any part of the actual hierarchy.

A single pattern instance may be required to have all its pattern inputs coming from the same “region” of the hierarchy; for example, the pattern inputs for a pattern instance may all be required to be within some given logical distance of each other. This, or some similar rule (possibly some statistical version of it in which pattern instances with all their pattern inputs near each other, in logical distance terms, would be more likely) would be to reduce the number of options available for connecting a pattern instance to something manageable.

3.2 Removal of Pattern Instances

Pattern instances are continually removed from the hierarchy. This is in a continuous way, based on the *relevance values* of pattern instances. The lower some relevance value, the less the representation we should want of it in the hierarchy.

There is no cutoff relevance at which pattern instances are not allowed in the hierarchy: A pattern instance’s relevance value just determines the likelihood that it will be allowed to remain when the hierarchy is being “pruned”. The higher some amount of relevance is, the greater the proportion of the population we will want with that relevance value. The idea is to select pattern instances for high relevance, by removing pattern instances so that the pattern instances with the highest relevance values are the most common, followed by those with slightly less relevance, followed by those with slightly less relevance, and so on, until, ultimately, we reach the pattern instances with the very lowest relevance values and these are the least common in the hierarchy.

3.3 Incompletely Specified Pattern Instances

The hierarchy was made completely probabilistic in an earlier article, *An Attempt to Generalize AI - Part 5: A Completely Probabilistic Hierarchy*, to make it practical to use pattern instances with "missing" pattern inputs.[15] The actual hierarchy in the AI system is a representation of the conceptual hierarchy – a structure containing all possible pattern instances of all possible patterns. When a hierarchy has "missing" pattern inputs in the AI system's actual hierarchy, these inputs really connect to pattern instances in the *conceptual* hierarchy that are not being represented in the AI system. Such pattern instances will result from the removal of pattern instances by the BERP in its attempt to make the hierarchy relevant.

A completely probabilistic hierarchy will still be able to function with this going on; however, it is still desirable that "missing" pattern inputs be kept to a minimum. This means that when patterns are adding new pattern instances to the hierarchy, the BERP should ensure that "broken" parts of the hierarchy are repaired, where possible, in preference to adding new pattern instances in the same region of the hierarchy.

[15] Almond, P. (2010). An Attempt to Generalize AI - Part 5: A Completely Probabilistic Hierarchy. *paul-almond.com*. http://www.paul-almond.com/AI05.pdf.
(Also available at http://www.paul-almond.com/AI05.doc.)

4 Forgetting and Bottom-Level Pattern Instances

4.1 Relevance does not mean information is *needed*.

The handling of relevance by the BERP has not dealt with *obsolescence* of pattern instances. The BERP, in its current form, does not provide the *forgetting* required in an AI system and a human mind.

The BERP may seem to be performing forgetting, because "irrelevant" pattern instances are being removed. Does this not mean that a pattern instances only stays in the system until it has become irrelevant, and that is then removed? This is the case – but "irrelevance", with "relevance" defined as it is here, and obsolescence are not synonymous: A pattern instance might be very *relevant* to reducing uncertainty in the prediction for some pattern instance, but it may be *obsolete* because it is not an important part of any *ongoing* computational process being used to generate any new information. We can see this with an analogy involving a spreadsheet.

4.2 The Spreadsheet Analogy for Obsolescence

Suppose there is a spreadsheet, intended to produce some final answer in a single cell. Various values are entered into the spreadsheet's cells, and these are used in formulae to compute the contents of other cells, which are used to compute the contents of other cells and so on.

Suppose that changing the contents of some cell, Cell 1, in the spreadsheet has little or no effect on the answer. This cell would not be very relevant to the answer.

Suppose now that changing the contents of another cell, Cell 2, does significantly affect the answer, by directly affecting another cell, Cell 3, which directly affects the answer. This makes Cell 2 highly relevant to the answer.

Suppose that after the spreadsheet is first made, it is used every day and, at first, Cell 3 changes significantly every day, with Cell 2 playing a significant role in this, and this significantly affects the answer. Cell 2 is relevant and *important*.

Now, suppose that after some time, the situation for which the spreadsheet was designed changes in some way. The spreadsheet is still used every day, but now Cell 3 does not change at all: It has had a value which has not changed for a long time. The formula which computes the value for Cell 3 is no longer doing anything useful: It is generating the same answer every day. We do not really need the formula for Cell 3. We could just replace it with the actual value which it has been generating for a long time. For example, if the formula in Cell 3 has been producing a value of 0.3217 for a long time, we could just take the formula out of Cell 3 and put 0.3217 in it as a constant value. Cell 3 would still be relevant and important, but it would not be changing. Any importance that Cell 2 had from its effects on Cell 3 is now gone. Cell 2 may still be important if it is part of some other, ongoing computation that affects the answer, but if this is not the case, Cell 2 is no longer of any interest.

Assuming that Cell 2 is only relevant due to its effects on Cell 3, it is not the decision to "freeze" Cell 3 that makes Cell 2 unimportant. Even before we decided to do this, Cell 2 had become unimportant, only being needed because we were going to keep using it in the same repetitive flow of information through the spreadsheet. This does not change the fact that Cell 2 is relevant, in the way that we have been using the word "relevant" so far, but "relevant" does not necessarily mean "*still* needed".

4.3 Obsolescence of Pattern Instances in the Hierarchy

What was said for spreadsheets, on page 15,[16] also applies for the hierarchy. Suppose we have Pattern Instance 1, which directly affects Pattern Instance 2, which directly affects some other pattern instances and so on, and Pattern Instance 1, because of its effect on Pattern Instance 2, has high relevance. This does not change the fact that, if Pattern Instance 2 does not significantly change over time, then any computation involved in repeatedly determining Pattern Instance 2's probability is redundant and Pattern Instance 1 is not really playing any important role in doing this.

4.4 The Need for Forgetting

All this indicates the need for some "forgetting" process. If there is a path through hierarchy to some point, and things do not change much at that point, the path leading up to it is of little use: We can just as easily retain the end-point of the path. This agrees with the basic forgetting procedure described in the third article, except that that was described before the hierarchy was made completely probabilistic.[17] A completely probabilistic forgetting process is needed, which causes pattern instances to be "frozen" when they are not changing much, removing their dependence on other pattern instances, which can then be discarded if they are not needed by other pattern instances. There are two ways of doing this: It can work as a separate process in its own right, removing pattern instances as the BERP does, or the existing concept of relevance can be extended to include this. This is a matter for a later article.

---

[16] Section 4.2: The Spreadsheet Analogy for Obsolescence.

[17] Almond, P. (2010). An Attempt to Generalize AI - Part 3: Forgetting. *paul-almond.com*. http://www.paul-almond.com/AI03.pdf. (Also available at http://www.paul-almond.com/AI03.doc.)

4.5 Bottom-Level Pattern Instances

The issue of bottom-level pattern instances needs some consideration. These will tend to be removed by any forgetting process, but some will tend to be removed in the existing pruning by the BERP.[18] Normally, when pattern instances are removed, they can be added again if needed: The hierarchy can be reduced in some region where it is insufficiently relevant, and later regrown if that region becomes relevant again. The information in the hierarchy is based on bottom-level pattern instances corresponding to previous inputs/outputs. If these are removed, the information in them is lost from the hierarchy. It may be advisable to make the conditions for removal of such pattern instances more demanding than for others.

[18] This might be considered a kind of "weak" forgetting process.

5 Conclusion

As discussed in previous articles, the system proposed in this series for AI use and as a description of the way the human mind works needs a way of only representing what is relevant about the world. The system is based on pattern instances which belong to patterns, and relevance means that only certain pattern instances are represented in the system.

A way of doing this, the basic exploratory relevance process (BERP) has been described. This is an exploratory process that continually extends the hierarchy in an exploratory way, and "prunes" it by removing low-relevance pattern instances. In this way, the hierarchy "grows" into high-relevance regions and retreats from low-relevance regions.

The BERP needs a way of determining the relevance of individual pattern instances in the hierarchy. This is provided by the relevance measurement process (MRP), described in a previous article.[19] Starting with particular bottom-level pattern instances corresponding to future inputs that are of interest and that have been assigned relevance by an external process, the MRP back-propagates relevance from these pattern instances, through the hierarchy. Back-propagation is based on the extent to which pattern instances directly affect the probability values of other pattern instances. For this to work, an external process is needed to assign relevance to the particular bottom-level pattern instances of interest that are at the start of the back-propagation process. This is provided by the action selection process, described in a previous article.[20] This requires probabilistic predictions of specific pattern instances corresponding to future evaluation function score values.

The BERP is likely to cause pattern instances to have incompletely specified pattern inputs, by removing pattern instances serving as those pattern inputs. This can be permitted due to the transition to a completely probabilistic hierarchy, made in a previous article.[21] Where possible, however, "broken" connections should be repaired when adding pattern instances.

The existing process does not deal with *forgetting*. Pattern instances should be removed from the hierarchy at some point when they have become *obsolete*. Lack of relevance, as relevance is currently defined, and obsolescence are not necessarily the same: A pattern instance may be highly relevant to the bottom-level pattern instances about

[19] Almond, P. (2010). An Attempt to Generalize AI - Part 6: Measuring Relevance. *paul-almond.com*. http://www.paul-almond.com/AI06.pdf. (Also available at http://www.paul-almond.com/AI06.doc.)
[20] Almond, P. (2010). An Attempt to Generalize AI - Part 2: Planning and Actions. *paul-almond.com*. http://www.paul-almond.com/AI02.pdf. (Also available at http://www.paul-almond.com/AI02.doc.)
[21] Almond, P. (2010). An Attempt to Generalize AI - Part 5: A Completely Probabilistic Hierarchy. *paul-almond.com*. http://www.paul-almond.com/AI05.pdf. (Also available at http://www.paul-almond.com/AI05.doc.)

which the predictions are wanted, but every path through which it affects them might always encounter intermediate pattern instances that have been stable for a long time. This would mean that the pattern instance, despite having high relevance, is not really needed: The stable pattern instances could just be assigned probabilities and any paths leading to them would then become unimportant. This suggests the need for a forgetting process, similar to the one proposed in a previous article, but capable of working in a completely probabilistic hierarchy.[22] This could work as a separate process, or the definition of relevance in the BERP could be modified to take account of obsolescence, the BERP then dealing with this accordingly.

The BERP is only intended as a description of a simple exploratory process. Another, more sophisticated exploratory relevance process (ERP) might be imagined. Such a process might be more sophisticated in a number of ways. It might take account of the distribution of relevance values in the hierarchy when determining where to expend the most computing work in extending it. The frequency of addition of pattern instances in different regions of the hierarchy might be made different for different patterns. The frequency with which the hierarchy is updated, both with regard to propagation of probability values and computation of relevance, might be made different for different regions of the hierarchy.

The approach to relevance in this article and the preceding one [23] has been described within the context of the particular system in this series of articles; however, the same general approach could be used with other systems based on probabilistic hierarchies, to reduce the computing and storage needs. For example, the same general approach could be used with the cruder system in one of my earlier articles.[24] An approach like this could be used with systems designed by other people. For an approach like this to work, there needs to be an external process for assigning relevance to particular, probabilistic elements, suggesting that something like the action selection process, described previously.[25] It is only worth doing this if individual probabilistic elements are not "hard-wired" into the system. For example, with a system consisting of layers of arrays of repetitive, probabilistic elements, it would not achieve anything. Such an approach based on hard-wiring of individual elements is a bad idea: It precludes measures to prune the hierarchy, meaning that the resulting hierarchy must waste almost all of the available computing resources on irrelevancy and must be shallow in its representation of the world.

---

[22] Almond, P. (2010). An Attempt to Generalize AI - Part 3: Forgetting. *paul-almond.com*. http://www.paul-almond.com/AI03.pdf. (Also available at http://www.paul-almond.com/AI03.doc.)
[23] Almond, P. (2010). An Attempt to Generalize AI - Part 6: Measuring Relevance. *paul-almond.com*. http://www.paul-almond.com/AI06.pdf. (Also available at http://www.paul-almond.com/AI06.doc.)
[24] Almond, P. (2006). A Proposal for General AI Modeling. http://www.paul-almond.com/Modeling.pdf. (Also available at http://www.paul-almond.com/Modeling.doc.)
[25] Almond, P. (2010). An Attempt to Generalize AI - Part 2: Planning and Actions. *paul-almond.com*. http://www.paul-almond.com/AI02.pdf. (Also available at http://www.paul-almond.com/AI02.doc.)

6 Bibliography

Almond, P. (2006). *A Proposal for General AI Modeling*. Retrieved 18 April 2010 from http://www.paul-almond.com/Modeling.pdf.
(Also available at http://www.paul-almond.com/Modeling.doc.)

Almond, P. (2010). An Attempt to Generalize AI - Part 1: The Modeling System. *paul-almond.com*. Retrieved 18 April 2010 from http://www.paul-almond.com/AI01.pdf.
(Also available at http://www.paul-almond.com/AI01.doc.)

Almond, P. (2010). An Attempt to Generalize AI - Part 2: Planning and Actions. *paul-almond.com*. Retrieved 18 April 2010 from http://www.paul-almond.com/AI02.pdf.
(Also available at http://www.paul-almond.com/AI02.doc.)

Almond, P. (2010). An Attempt to Generalize AI - Part 3: Forgetting. *paul-almond.com*. Retrieved 18 April 2010 from http://www.paul-almond.com/AI03.pdf.
(Also available at http://www.paul-almond.com/AI03.doc.)

Almond, P. (2010). An Attempt to Generalize AI - Part 4: Modeling Efficiency. *paul-almond.com*. Retrieved 18 April 2010 from http://www.paul-almond.com/AI04.pdf.
(Also available at http://www.paul-almond.com/AI04.doc.)

Almond, P. (2010). An Attempt to Generalize AI - Part 5: A Completely Probabilistic Hierarchy. *paul-almond.com*. Retrieved 18 April 2010 from http://www.paul-almond.com/AI05.pdf.
(Also available at http://www.paul-almond.com/AI05.doc.)

Almond, P. (2010). An Attempt to Generalize AI - Part 6: Measuring Relevance. *paul-almond.com*. Retrieved 18 April 2010 from http://www.paul-almond.com/AI06.pdf.
(Also available at http://www.paul-almond.com/AI06.doc.)