
An Attempt to Generalize AI Part 9: Improving the Exploratory Relevance Process

By Paul Almond

13 May 2010

Website:

<http://www.paul-almond.com>

E-mail:

info@paul-almond.com

This is the ninth in a series of articles attempting an overview of how minds may work and how similar systems could be implemented in computers. Previous articles 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 specified in a very general way, to provide a very general ontology. A *basic, exploratory relevance process* has been described, which achieves relevance of the hierarchy by removing low-relevance pattern instances while the hierarchy "grows", so that it will tend to retreat from low-relevance regions and grow into high-relevance ones. The basic, exploratory relevance process uses a *relevance measurement process*, which back-propagates relevance through the hierarchy. The basic, exploratory relevance process also performs forgetting – the removal of obsolete pattern instances from the hierarchy. This is done in the relevance measurement process, by taking account of obsolescence. The basic, exploratory relevance process is only a simple exploratory relevance process, with about the minimum of processing needed to function. This article describes possible improvements to the basic, exploratory relevance process, to produce a more sophisticated exploratory relevance process. Possible improvements are varying update frequency, making pattern instance addition explicitly dependent on relevance and examining relevance for regions.

Table of Contents

1 Introduction	4
2 Weak Geometry	6
3 Update Frequency Dependent on Relevance	8
3.1 The General Idea of Update Frequency Dependence on Relevance	8
3.2 Update Frequency for Probabilistic Propagation Dependent on Relevance	8
3.3 Update Frequency for Relevance Back-Propagation Dependent on Relevance	9
3.4 Long and Short-Term Relevance	9
3.5 Will this be of any use?	10
4 Pattern Instance Addition <i>Explicitly</i> Dependent on Relevance	11
4.1 The Basic Approach	11
4.2 Clarification	11
4.3 A Possible Change	12
5 Relevance Computed for <i>Regions</i> of the Hierarchy	14
6 Conclusion	15
7 Bibliography	17

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 ninth 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. The fourth article, *An Attempt to Generalize AI - Part 4: Modeling Efficiency*, which is at <http://www.paul-almond.com/AI04.pdf>, suggested that pattern instances should be allowed to have *incompletely specified pattern inputs*, so that it would be practical for the hierarchy to be “pruned” by some process seeking to maximize its relevance.⁴ This required a *completely* probabilistic hierarchy, an issue 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 made a process to provide relevance in the hierarchy feasible. A relevance measurement process (RMP) was needed first. The sixth article, *An Attempt to Generalize AI - Part 6: Measuring Relevance*, which is at <http://www.paul-almond.com/AI06.pdf>, described a back-propagation process for measuring relevance in the hierarchy.⁶ The problem is made tractable by the way in which the *action selection process*, described in the second article, *An Attempt to Generalize AI - Part 2: Planning and Actions*, works. 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

¹ Almond, P., 2010. *An Attempt to Generalize AI - Part 1: The Modeling System*. paul-almond.com. <http://www.paul-almond.com/AI01.pdf> or <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> or <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> or <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> or <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> or <http://www.paul-almond.com/AI05.doc>.

⁶ Almond, P., 2010. *An Attempt to Generalize AI - Part 6: Measuring Relevance*. paul-almond.com. <http://www.paul-almond.com/AI06.pdf> or <http://www.paul-almond.com/AI06.doc>.

An Attempt to Generalize AI – Part 8: Improving the Exploratory Relevance Process

what is most relevant and allows us to describe a process of relevance back-propagation from these pattern instances, through the rest of the hierarchy, allowing the relevance of every part of the hierarchy to be measured. The seventh article, *An Attempt to Generalize AI – Part 7: A Basic, Exploratory Relevance Process*, which is at <http://www.paul-almond.com/AI07.pdf>, described the *basic exploratory relevance process* (BERP), which uses this measuring process to direct the growth and pruning of the hierarchy.⁷

The change to a completely probabilistic hierarchy meant that the approach to forgetting in *An Attempt to Generalize AI - Part 3: Forgetting*⁸ had become unusable in its current form. The eighth article, *An Attempt to Generalize AI – Part 8: Forgetting as Part of the Exploratory Relevance Process*, which is at <http://www.paul-almond.com/AI08.pdf>, resolved this issue by incorporating forgetting into the basic, exploratory relevance process, or any other exploratory relevance process (ERP).⁹ This was done by modifying the relevance measurement process to take account of obsolescence.

The exploratory relevance process previously described is only the *basic*, exploratory relevance process: a simple exploratory relevance process described to show how an exploratory relevance process can work. In this article, ways in which more sophisticated exploratory relevance processes could work will be described.

⁷ Almond, P., 2010. *An Attempt to Generalize AI - Part 7: A Basic, Exploratory Relevance Process*. paul-almond.com. <http://www.paul-almond.com/AI07.pdf> or <http://www.paul-almond.com/AI07.doc>.

⁸ Almond, P., 2010. *An Attempt to Generalize AI - Part 3: Forgetting*. paul-almond.com. <http://www.paul-almond.com/AI03.pdf> or <http://www.paul-almond.com/AI03.doc>.

⁹ Almond, P., 2010. *An Attempt to Generalize AI – Part 8: Forgetting as Part of the Exploratory Relevance Process*. paul-almond.com. <http://www.paul-almond.com/AI08.pdf> or <http://www.paul-almond.com/AI08.doc>.

2 Weak Geometry

This article will refer to the idea of “regions” in the conceptual hierarchy, as well as related concepts such as “nearness” of pattern instances. When I refer to concepts that might *seem* to relate to geometry, such as “region” or “nearness” in this article, as in previous articles, a spatial analogy is *not* being used.

Most of the hierarchy is *not* explicitly spatial: Pattern instances do not have coordinates in some space. Instead, a pattern instance is described in terms of its internal logic and state and *the pattern instances to which it is connected*. This does not preclude the use of a weak, geometrical analogy: We can still apply some idea of proximity to the hierarchy, without space, as I explained in *An Attempt to Generalize AI – Part 4: Modeling Efficiency*, where I stated the following.

“Nevertheless, we can still apply a weak, geometrical analogy, using the concept of logical distance between pattern instances. If one pattern instance were directly connected to another, using it as a pattern input, we would say that those two pattern instances were “close together”. Two pattern instances connected together through an intermediate pattern instance would be less close. We could generalize this by saying that the logical distance between any two pattern instances is the number of connections between pattern instances which must be traversed to move between them. Logical distance is not an arbitrary concept. This is only one way, and the simplest, of determining logical distance. A more sophisticated method might involve averaging over different paths, or determining the logical distance between two pattern instances by determining the logical distance between each and some third pattern instance, and doing this a number of times, changing the third pattern instance each time. However we compute it, the concept of logical distance allows us to impose a weak, geometrical analogy on the system: It allows us to look at a pattern instance and say which pattern instances are “nearby” in some sense, without having to impose any well-defined spatial relationships on the system.”¹⁰

Some readers might ask how any information about geometry manages to enter the hierarchy in the first place. It should be noted that on the bottom level of the hierarchy, the pattern instances would often have some relationship to each other corresponding to the real world: For example, if two light sensitive elements are next to each other in a camera, this needs to be represented in the hierarchy. As an exception, the construction specifications for pattern instances would be able to access information for bottom-level pattern instances that was directly derived from the real-world arrangement of the corresponding input/output devices. Immediately above the bottom level, information derived from spatial relationships would not be explicitly available, but it should still be

¹⁰ Almond, P., 2010. *An Attempt to Generalize AI - Part 4: Modeling Efficiency*. paul-almond.com. <http://www.paul-almond.com/AI04.pdf> or <http://www.paul-almond.com/AI04.doc>. p. 11.

An Attempt to Generalize AI – Part 8: Improving the Exploratory Relevance Process

reflected in the *logical* structure of the hierarchy, as it has been derived directly from a structure with explicit spatial relationships. Further above the bottom level, things would get further away from the simple, spatial relationships on the bottom level, and relationships between pattern instances would become increasingly abstracted into ones that cannot be well-represented by simple, spatial coordinates.

3 Update Frequency Dependent on Relevance

3.1 The General Idea of Update Frequency Dependence on Relevance

The basic, exploratory relevance process (BERP), described in *An Attempt to Generalize AI – Part 7: A Basic, Exploratory Relevance Process*, achieves relevance by removing pattern instances, with the relevance of a pattern instance determining its likelihood of remaining in the hierarchy: The lower the relevance of a pattern instance, the more likely it is that it is removed at any time.¹¹

A more sophisticated exploratory relevance process (ERP) might treat pattern instances differently with regard to propagation of information, with propagation occurring less frequently for low-relevance pattern instances. The idea of this would be that a lower-relevance pattern instance is less important, so we should not be spending as much computing power on it. This could apply to probabilistic propagation through the hierarchy and relevance back-propagation.

3.2 Update Frequency for Probabilistic Propagation Dependent on Relevance

Probabilistic information is propagated through the hierarchy by the processes of logic application and statistics application.¹² Provided that a pattern instance exists in the actual hierarchy, it is updated as often as any other pattern instance. In a more sophisticated ERP, the frequency with which pattern instances are updated in probabilistic propagation may take account of relevance, with frequency of update decreasing as relevance decreases. High-relevance regions of the hierarchy would be updated often, but low-relevance regions might be updated infrequently.

This could mean that probabilistic information is propagated *from* a pattern instance with a frequency dependent on its relevance, so that low-relevance pattern instances affect others by probabilistic propagation less often, or it could mean that probabilistic information is propagated *into* a pattern instance with a frequency dependent on its relevance, so that low-relevance pattern instances are affected by probabilistic propagation from others less often.

¹¹ Almond, P., 2010. *An Attempt to Generalize AI - Part 7: A Basic, Exploratory Relevance Process*. paul-almond.com. <http://www.paul-almond.com/AI07.pdf> or <http://www.paul-almond.com/AI07.doc>. pp. 13-14.

¹² Almond, P., 2010. *An Attempt to Generalize AI - Part 1: The Modeling System*. paul-almond.com. <http://www.paul-almond.com/AI01.pdf> or <http://www.paul-almond.com/AI01.doc>. p. 15, 19.

3.3 Update Frequency for Relevance Back-Propagation Dependent on Relevance

This is similar to what was just discussed for probabilistic propagation, in 3.2, but it is applied to relevance. The frequency with which pattern instances are updated in relevance back-propagation may depend on relevance, with frequency of update decreasing as relevance decreases. High-relevance regions of the hierarchy would be updated with new relevance values very often, but low-relevance regions might be updated very infrequently.

This could mean that relevance is back-propagated *from* a pattern instance with a frequency dependent on its relevance, so that low-relevance pattern instances affect others by relevance back-propagation less often, or it could mean that relevance is back-propagated *into* a pattern instance with a frequency dependent on its relevance, so that low-relevance pattern instances are affected by relevance back-propagation from others less often.

3.4 Long and Short-Term Relevance

An obvious way to use approaches like those just described, in 3.1, to 3.3, is to consider the relevance for different timescales when removing pattern instances in the ERP and when determining what their update frequency in probabilistic/relevance propagation is.

- The ERP, when it prunes the hierarchy, may base its decisions on relevance over a *long* period of time, so that the chance that any pattern instance is removed is based on the average of its relevance over a long period of time: The lower this value is, the more likely a pattern instance would be to be removed at any time.
- The frequency with which pattern instances are updated, in probabilistic propagation or relevance back-propagation, may be based on relevance values averaged over a *short* period of time – or even on single relevance values: The lower the average of the recent values of a pattern instance's relevance is, the lower would be the frequency with which that pattern instance is updated or updates others.¹³

¹³ This could be done for one or both of probabilistic propagation and relevance back-propagation. Different timescales might be used in each case.

3.5 Will this be of any use?

Whether it is *useful* to vary the update frequencies for pattern instances depends on how the computing hardware is put together.

- In some systems, all the storage and processing associated with a single pattern instance might be provided by hardware dedicated to that pattern instance, so that none of the computing resources used by a pattern instance, while it still exists, can be made available to any other pattern instance. That is to say, “storing” a pattern instance would use as much computing power as “running” it. In such systems, approaches like the ones just discussed, in 3.1, to 3.4, would be useless, because the computing resources freed when a pattern instance was not being updated, or updating others, would not be available for any other pattern instances.
- In other systems, some of the computing resources used in propagation by a pattern instance might be made available to others when the pattern instance was not being updated or updating others. That is to say, just “storing” a pattern instance would use less computing power than was needed to “run” it. In such a system, approaches like those just discussed could be more useful, with pattern instances having computing resources gradually withdrawn as they become less relevant, ultimately being removed completely if the ERP makes the decision to do this.

At one extreme, with a massively parallel computer, in which a computational element was assigned to each pattern instance, and specific hardware was dedicated to each connection between pattern instances, with pattern instances not existing in any other way, approaches like those discussed would have no benefit.

At another extreme, with a computer built according to the Von Neumann architecture, with information about the pattern instances and their connections being stored in memory, separate from processing, and propagation through the hierarchy being done by sequential processing, approaches like those just discussed would significantly reduce the processing time needed for infrequently updated pattern instances, although the same amount of memory would still be required to store the information about them.

A real computer might be somewhere between these two extremes. It may have many components working in parallel, but there may be some way in which a less frequently updated pattern instance is using fewer resources. For example, pattern instances might have dedicated computational units to store their states, while the processing to propagate information between pattern instances is performed on hardware that is shared among “local” pattern instances.

4 Pattern Instance Addition *Explicitly* Dependent on Relevance

4.1 The Basic Approach

The idea of the BERP is that the density of the actual hierarchy is reduced where it seems insufficiently relevant to justify a high density, and increased where it seems particularly relevant. As part of achieving this, new pattern instances should generally be added in higher-relevance regions of the actual hierarchy, to cause it to “grow” into high-relevance regions. The BERP already does this, because the pruning by the BERP already selects the pattern instances of the actual hierarchy for high relevance, and pattern instances are connected to the actual hierarchy in random regions. As the higher-relevance regions of the actual hierarchy will be denser than low-relevance regions, it is will be more likely that new pattern instances get connected here.

4.2 Clarification

Some clarification about the previous discussion on pattern instance addition may be useful here. In *An Attempt to Generalize AI – Part 7: A Basic, Exploratory Relevance Process*, I stated the following about addition of new pattern instances.

“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 ... 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.”*¹⁴

There does not need to be any preference about where in the actual hierarchy new pattern instances are connected, because the hierarchy’s pattern instances have already been selected for high relevance: If a region of the hierarchy was really bad for adding new pattern instances, the hierarchy would probably have withdrawn from it, or would be of such low density there that new pattern instances would be rarely added there. For that reason, we can add pattern instances at random “places” in the hierarchy.

¹⁴ Almond, P., 2010. *An Attempt to Generalize AI - Part 7: A Basic, Exploratory Relevance Process*. paul-almond.com. <http://www.paul-almond.com/AI07.pdf> or <http://www.paul-almond.com/AI07.doc>. p. 7.

Some qualification of this is needed. It might be taken as implying that all the labeled pattern inputs of a new pattern instance are connected randomly, so that there is no organization at all in how a pattern instance is connected. This would not be correct: The way in which the labeled pattern inputs of a pattern instance are connected is controlled by the construction specification. It would be more accurate to understand this as meaning that there is no preferred part of the hierarchy in which to connect pattern instances, even though the use of particular pattern inputs for some pattern inputs implies that others will be used for other pattern inputs.

This might be done in a number of ways. One way would involve randomly selecting a pattern instance in the actual hierarchy at random. The construction specification for the relevant pattern would then select a pattern instance as near as possible to it as the first labeled pattern input. Once the first labeled pattern input had been connected, the construction specification would then control selection of the pattern instances to be used for the other pattern inputs. The result would be a pattern instance connected in a random region of the hierarchy, but with the construction specification still controlling individual selection of pattern instances for pattern inputs. The pattern instances serving as pattern inputs of a single pattern instance might be required to be logically “near” each other most of the time.¹⁵

4.3 A Possible Change

As just discussed, in 4.1 and 4.2, the BERP will automatically tend to add pattern instances in high-relevance regions of the hierarchy, because these are the ones that will be most represented in the actual hierarchy. It may be desirable, however, to exaggerate this, with the ERP *explicitly* taking account of relevance when adding new pattern instances.

If such an approach is used, when the ERP is adding a new pattern instance, the relevance values of pattern instances will be examined, and the probability that any single pattern instance is used as the point where the new pattern instance is connected increases as its relevance increases. Exactly how this works will depend on how the construction specification functions, but the general idea should be that preference is given to high-relevance pattern instances when it comes to connecting new ones. This could involve an approach similar to that just discussed, in 4.2, with some modification to prioritize pattern instances according to relevance.

This could also be combined with approaches in which update frequency is based on relevance, as previously discussed in Section 3: Update Frequency Dependent on Relevance, on page 8: As well as the update frequency for probabilistic propagation or relevance back-propagation depending on relevance, the frequency with which pattern instances are added in a given region of the hierarchy could also depend on relevance.

¹⁵ using a non-spatial understanding of the word “near”.

An Attempt to Generalize AI – Part 8: Improving the Exploratory Relevance Process

It is also possible to examine relevance averaged over different periods of time when adding and removing pattern instances, with the removal of pattern instances being based on relevance averaged over a longer period of time than it is for pattern instance addition. Whether this would be useful is debatable.

5 Relevance Computed for *Regions* of the Hierarchy

The BERP examines relevance values for single pattern instances only, but some of the processes occurring as part of the BERP might be more effective if they take account of the relevance for a *region* of the hierarchy, instead. As previously stated in Section 2: Weak Geometry, on page 6, the hierarchy is not spatial, but the idea of a “region” could still be applied to it by considering the pattern instances which are logically “near” a particular pattern instance. For example, to obtain the relevance of a region we might obtain the relevance of some pattern instance, then take less account of the relevance of those logically closest to it, and then still less account of the relevance of those next closest to it, and so on, combining all this information to obtain a single relevance value.

This could be done when determining update frequency for probabilistic propagation or relevance back-propagation, when removing pattern instances or when adding pattern instances; however, the argument for doing it may not be very strong in all these cases.

The argument for taking account of regional relevance might be strongest in relation to addition of new pattern instances. The idea, when adding a new pattern instance, should be for it to have a high relevance. After it has been added, the ERP can examine it and remove it (or make it more likely that it will be removed) if its relevance is too low, but by then, in one respect, it is too late: Computing resources have already been spent on adding the pattern. Adding patterns randomly this way, and hoping to find high-relevance ones occasionally, would be too wasteful. The exploratory nature of the process already deals with this: The previous removal of low-relevance pattern instances means that any addition of a pattern instance is more likely to occur in a higher-relevance region of the hierarchy. I have just stated that we may want to give more explicit consideration to relevance when adding pattern instances, in 4, but while the relevance of a single pattern instance may be a useful guide, it will not be a perfect indicator of the relevance that a new pattern instance will have if we connect it logically nearby. Relevance will vary from pattern instance to pattern instance, and we may see that the relevance of a pattern instance is high, and connect a new pattern instance logically nearby, expecting that to have high-relevance too, and find out that it has low-relevance. We do not need to get this right every time, as the ERP’s pruning will remove failures, but we should want to get it right as often as possible. One way of improving this may be to look at the relevance of a region when adding new pattern instances, to obtain a more reliable indication of the likely relevance of local pattern instances than would be obtained by examining a single pattern instance.

6 Conclusion

Limited computing power means that an AI system, or the human mind, needs a way of representing only relevant features of the world. In the context of the kind of system described in this series of articles, this means a way of ensuring the relevance of the actual hierarchy, so that only those pattern instances which are relevant are represented in the actual hierarchy.

Relevance is provided by the basic, exploratory relevance process (BERP) described previously.¹⁶ This requires the relevance measurement process (RMP),¹⁷ which back-propagates relevance through the actual hierarchy, starting with bottom-level pattern instances corresponding to future inputs/outputs that are assigned relevance externally. (Relevance is assigned externally to pattern instances due to their role in the action selection process.¹⁸) The BERP removes pattern instances from the hierarchy, in a way based on the relevance values of pattern instances. The chance of removal for any pattern instance at any time increases as its relevance decreases. As low-relevance pattern instances tend to be removed, when new pattern instances are added to the hierarchy, they tend to be added in high-relevance regions, so that the density of the actual hierarchy decreases in low-relevance regions and increases in high-relevance regions.

The BERP is a minimal example of an exploratory relevance process (ERP) and more sophisticated ERPs to perform the same task can be imagined. This article has discussed ways in which the BERP could be improved to provide a more sophisticated ERP.

One way in which the BERP could be improved is by introducing variable update frequency, with the update frequency of pattern instances for probabilistic propagation or relevance back-propagation through the hierarchy being dependent on relevance, decreasing as relevance decreases. This could also involve averaging relevance over different timescales, with relevance values over a long time being averaged to obtain the relevance values on which pattern instance removal is based and relevance values averaged over a short time being used to determine update frequency. Varying the update frequency would only be worthwhile if the architecture of the computer running the hierarchy allowed some of the computing resources allocated to a pattern instance to be made available to others when it was not being updated.

¹⁶ Almond, P., 2010. *An Attempt to Generalize AI - Part 7: A Basic, Exploratory Relevance Process*. paul-almond.com. <http://www.paul-almond.com/AI07.pdf> or <http://www.paul-almond.com/AI07.doc>. pp. 13-14.

¹⁷ Almond, P., 2010. *An Attempt to Generalize AI - Part 6: Measuring Relevance*. paul-almond.com. <http://www.paul-almond.com/AI06.pdf> or <http://www.paul-almond.com/AI06.doc>. pp. 16-19.

¹⁸ Almond, P., 2010. *An Attempt to Generalize AI - Part 2: Planning and Actions*. paul-almond.com. <http://www.paul-almond.com/AI02.pdf> or <http://www.paul-almond.com/AI02.doc>. pp. 9-16.

An Attempt to Generalize AI – Part 8: Improving the Exploratory Relevance Process

Pattern instance addition by the BERP will tend to occur in high-relevance regions of the hierarchy, because the BERP is removing low-relevance pattern instances, but an ERP may be made more sophisticated by making pattern instance addition explicitly dependent on relevance, so that relevance values of pattern instances already in the hierarchy are examined when adding new pattern instances.

An ERP might be more sophisticated if it examined the relevance of *regions* of the hierarchy, rather than just the relevance of individual pattern instances. This could be particularly useful when adding new pattern instances.

7 Bibliography

Almond, P., 2010. *An Attempt to Generalize AI - Part 1: The Modeling System*. [Online] paul-almond.com. Available at: <http://www.paul-almond.com/AI01.pdf> or <http://www.paul-almond.com/AI01.doc> [Accessed 13 May 2010].

Ibid. p. 15, 19.

Almond, P., 2010. *An Attempt to Generalize AI - Part 2: Planning and Actions*. [Online] paul-almond.com. Available at: <http://www.paul-almond.com/AI02.pdf> or <http://www.paul-almond.com/AI02.doc> [Accessed 13 May 2010].

Ibid. pp. 9-16.

Almond, P., 2010. *An Attempt to Generalize AI - Part 3: Forgetting*. [Online] paul-almond.com. Available at: <http://www.paul-almond.com/AI03.pdf> or <http://www.paul-almond.com/AI03.doc> [Accessed 13 May 2010].

Almond, P., 2010. *An Attempt to Generalize AI - Part 4: Modeling Efficiency*. [Online] paul-almond.com. Available at: <http://www.paul-almond.com/AI04.pdf> or <http://www.paul-almond.com/AI04.doc> [Accessed 13 May 2010].

Ibid. p. 11.

Almond, P., 2010. *An Attempt to Generalize AI - Part 5: A Completely Probabilistic Hierarchy*. [Online] paul-almond.com. Available at: <http://www.paul-almond.com/AI05.pdf> or <http://www.paul-almond.com/AI05.doc> [Accessed 13 May 2010].

Almond, P., 2010. *An Attempt to Generalize AI - Part 6: Measuring Relevance*. [Online] paul-almond.com. Available at: <http://www.paul-almond.com/AI06.pdf> or <http://www.paul-almond.com/AI06.doc> [Accessed 13 May 2010].

Ibid. pp. 16-19.

Almond, P., 2010. *An Attempt to Generalize AI - Part 7: A Basic, Exploratory Relevance Process*. [Online] paul-almond.com. Available at: <http://www.paul-almond.com/AI07.pdf> or <http://www.paul-almond.com/AI07.doc> [Accessed 13 May 2010].

Ibid. p. 7.

Ibid. pp. 13-14.

Almond, P., 2010. *An Attempt to Generalize AI - Part 8: Forgetting as Part of the Exploratory Relevance Process*. [Online] paul-almond.com. Available at: <http://www.paul-almond.com/AI08.pdf> or <http://www.paul-almond.com/AI08.doc> [Accessed 13 May 2010].