

An Evaluation of the Accuracy of Capturing User Intent for Information Retrieval

Hien Nguyen

Dept. of Math and Computer Sciences
University of Wisconsin-Whitewater
Whitewater, WI 53190

Eugene Santos Jr.

Thayer School of Engineering
Dartmouth College
Hanover, New Hampshire 03755-8000

Abstract *This paper reports our evaluation of the accuracy of capturing a user’s intent in an information-seeking task. Specifically, we would like to assess how accurately a user’s short-term goals, methods, and context in an information seeking task have been captured. Our method is to compare a machine-generated model against a human-generated model with 5 users using the CACM collection. Our results demonstrate a good coverage of human-generated user models by machine-generated models in terms of short-term goals, methods and context. The similarity between a real user and our approach in terms of context agrees with the similarities of human-generated ontologies using the same metrics in existing studies from the ontology community. Furthermore, our results show that the similarities between a human and machine in terms of short term goals and methods, which affect the relevancy assessment process, agree with the overlap among people while assessing relevancy documents in the existing study from the information retrieval community.*

Keywords: User Model for Information Retrieval, Evaluation of User Models

1 Introduction

Empirical evaluations of systems using user models are useful to both the user modeling (UM) community and the researchers in the area of the target application. Using the results of such an evaluation, researchers and designers can verify whether their user models are working in the ways they expected. The important goals for the researchers conducting such empirical evaluations are to answer these two questions: (1) Does a user model capture and infer a user’s cognitive states *accurately*? (2) Does a user model improve a user’s performance *significantly*?

The first question focuses on the *inputs* or *internal* states of a user model while the second question

places an emphasis on the *outputs* or *external* impacts of a user model on the target application. In this paper, a user’s cognitive states are used to refer to a user’s state of mind related to the task at hand (e.g. the topics that a user is focusing on reading). Most of the empirical evaluations so far have emphasized answering the second question, (for example: [3, 6, 4]). Even though, the problem of assessing the accuracy of capturing a user’s cognitive states is considered as one of the major issues in the UM community since the early 90’s [21, 2, 14, 24], there are very few studies that include the assessment of the internal accuracy of a user model. This is unfortunate because without an evaluation of the accuracy of a user model, we cannot convincingly justify the impacts that the user model has on the target application.

In this paper, we report our experiment to evaluate the accuracy of our user model which captures and infer a user’s intent in an information seeking task. This is a challenging problem because of the following reasons. First, a user’s intent primarily resides in the mind of the user and thus difficult to observe. Furthermore, even when the user is willing to share his intent, it is often a challenge for him to accurately describe. Second, there are also significant differences in terms of representation and granularity between a machine-generated model (computational model), especially by using artificial intelligence techniques, and a user’s cognitive states [20]. Therefore, it is hard to find an appropriate representation scheme as well as metrics for measuring the similarity between a machine-generated model and a human-generated model. Moreover, even people do not always agree with each other in terms of context and cognitive styles in general [5, 10]. Therefore, in such an evaluation, a machine model is considered to capture accurately a human model if the overlap between these two models is appropriate (i.e. comparable with

the overlap among people on the same aspect).

In our approach, we capture a user’s intent in an information seeking task by finding the commonality of the retrieved relevant documents and use this information to modify a user’s query proactively [17, 12]. In this paper, we assess the accuracy of the process of capturing and inferring a user’s intent by determining to what extent a model created by a human is covered by a model created by using our approach. This experiment helps us to justify our earlier results on the impacts of our user model [13, 18]. We partition a user’s intent into three formative components which are Interest, Preferences, and Context. Therefore, we compare these three sets indicated by a user (*human-generated user model*) against the sets automatically generated by the system (*machine-generated user model*). We measure the similarity between the machine-generated Context and the human-generated Context using the similarity measures from the ontology community [10]. We hypothesize that the human-generated user model will be appropriately covered by the machine-generated user model. In the long run, we expect that the machine-generated user model will fully cover the human-generated user model as the user keeps using the system.

Our results have shown a good coverage in terms of Interest, Preference and Context, which represents the agreements between a real user and our approach in terms of short-term goals, methods, and contexts used in an information seeking task. Moreover, the similarity of the lexical parts of the machine-generated and human generated Context agrees with the similarity of the lexical parts between two humans using the same metrics in the another study [10]. The lexical parts refer to the concepts generated by either a real user or by our approach while constructing the corresponding Context. Furthermore, our results show that the similarities between a human and machine in terms of short term goals and methods, which affect the relevancy assessment process, agree with the overlap among people while assessing relevancy documents in the existing study from the information retrieval (IR) community. The significance of these results is two fold. First, the models being compared in our experiment are more fine-grained granularity than the ones in the existing studies. Second, our results agree with the similarities of the existing studies among human with regards to cognitive searching styles and ontologies. This paper is organized as follows: We start by reviewing some related work on evaluating the accuracy of a user model. Next,

we briefly present the architecture of our approach. The experimental setup, procedures, metrics, and results will be discussed. Lastly, we conclude by the future work.

2 Related Work

Although the problem of assessing the accuracy of a user model’s inference mechanism based on information about a user has been considered important since the early 90s [21], very little attention has been paid to this until the last few years when the layer evaluation approach was proposed [2, 14, 24]. In the layer evaluation approach, the four main evaluation steps are evaluating of a user model’s inputs, inference mechanism, adaptation decisions, and the changes in a user’s behaviors and a machine’s behaviors when a system adapts. Some studies have applied this framework, such as the evaluation of *HTML-Tutor* - an online adaptive course [24], and the assessment of a web-based application using ACT theory [7]. In these evaluations, the correctness of a user model’s internal states or the inference mechanism has been assessed.

There are two main methods to evaluate the accuracy of the internal states and inference mechanism of a user model. In the first method, a user’s data is divided into two sets. One set is used as the training set and the other is used as the test set. The user model will be trained with the training data. We determine how much the data from the test set has been correctly inferred by the user model. Examples of this method can be found in [11, 1].

In the second approach, a user is asked explicitly to construct his/her user model and that user model will be compared with the machine generated user model using the same interactions from the user. Examples of this method can be found in [22, 23]. In this paper, we apply the second method to evaluate the internal states and inference mechanism of a user model for IR. We assess how our model keeps track with a user’s cognitive states in an information seeking task and collect data for further evaluation. The novelty in our experiment lies with the finer-grain granularity in the structure of a user model being constructed by a real user and by our system. In addition to evaluating the similarity between the topics that a user is currently interested in and the topics generated by our approach (*lexical level*), we push our efforts further to evaluate the similarity of the relationships between

these topics (*syntactic level*). To the best of our knowledge, this has not been done before for any user model embedded in an IR application.

3 IPC User Model

We capture a user’s intent by finding the commonality of the content of the retrieved relevant documents and use this information to modify the user’s query proactively. We partition a user’s intent into three formative components: Context, Interests and Preferences. The Context provides insight into the user’s knowledge and represents the connections between a user’s goals for easy explanation. It is captured in Context network (C). C consists of concept nodes and relation nodes. A concept node is a noun phrase. We capture two types of relation nodes: set-subset (denoted as “*isa*”) and related to (denoted as “*related_to*”). C is created dynamically by finding the set of common subgraphs in the intersection of the retrieved relevant documents. The concepts and relations with bold outlines are the nodes from the intersections of the retrieved relevant documents.

The Interests capture the focus and direction of the individual’s attention and is stored in the interest set (I). Each element of I consists of interest concept (a) and interest level ($L(a)$). An interest concept represents the concept that a user is currently focusing on while an interest level is any real number from 0 to 1 representing how much emphasis he places on this particular concept. I is initially determined from the current query, and the set of common subgraph.

Lastly, the Preferences describe the actions needed to perform the proactive query modifications to better meet the user’s goals/interests. We capture Preferences in a Bayesian network [8] which consists of three kinds of nodes: pre-condition, goals, and action nodes. Precondition nodes represent the requirements to achieve the goal nodes. Goal nodes represent the tools that are used to modify a user’s query. We currently have the two tools: filter which narrows down a query semantically and expander which broadens up a query. Figure 1 shows an example of I , P , and C . The shaded nodes represent the current pre-condition nodes set as evidence. We perform belief updating to find the tool with the highest marginal probability. The system uses the information from the user model to modify a query by making the concepts of the original query more specific or more general. For more detail, please see [17, 12].

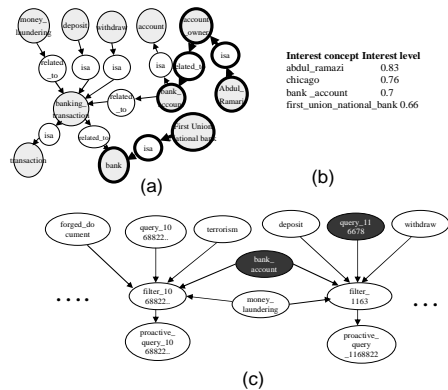


Figure 1: (a) Context network (b) Interest set (c) Preference network

4 Experimental setup

In this experiment, the sets of Interest, Preferences, and Context indicated by a real user (*human-generated user model*) are compared with the sets of Interest, Preferences, and Context automatically generated by the system (*machine-generated user model*). The model generated by a user is explicitly described in an intermediate questionnaire and the model generated by our approach is kept in the files that contain the user model.

4.1 Procedure

Five graduate students in Computer Science and Engineering at the University of Connecticut participated in the experiment. Our goal is to assess the accuracy of the internal state of our user model. Each participant performs a search with the same set of ten questions extracted from the CACM collection. The CACM collection contains 3204 documents and 64 queries in the computer science domain [15]. The CACM collection is chosen because it matches the background of these five participants. The total of 10 queries has been found to be a reasonable number of queries in the evaluation of our user model’s external impacts [18]. By fixing the queries, we avoid introducing more variables into our experiment such as error in natural language processing. It allows us to focus on our main objective of this experiment. The procedure goes as follows:

Step 1: Each participant is required to fill out an entry questionnaire which contains five questions: two questions concerning the user’s expertise on using search engines and using IR applications with relevance feedback; two questions concerning the user’s expertise on the testing domains

which are optimization and distributed computing; and one question about the user’s searching style. Each participant rates his/her expertise on a 7 point Likert scale with 1 being the most experience and 7 being the least experience. Each participant also chooses among 3 options to describe his/her searching styles: *global focus* (approaching a searching topic from general concepts to specific concepts), *local focus* (approaching a searching topic from specific concepts to more general ones), or *random*(using both global or local focuses depending on the searching topics).

Step 2: Each participant is given a list of 10 queries. The first five queries are about “distributed computing” while the last five queries are about “optimization”.

Step 3: Starting with the first query on the query list, each participant runs the program. The program returns a list of the documents whose similarity to the current query is non-zero.

Step 4: By reading the first 15 returned documents, each participant assessed each document to see if it is relevant to the current query. Then each participant is required to construct and list his/her interests, preferences and context on a paper. For example: the user (shown in Figure 2) listed his interests as: “*distributed computer network*”, and “*distributed computing*”; his preferences as filter (which is used to narrow down a query); and one relation in his context as “*topology information - related to - distributed computer network*”.

Step 5: After marking all the checkboxes next to the retrieved relevant documents, each participant is asked to press **Put Feedback** button to put his/her feedback to the system. Then he/she runs the query again. This time, the user’s query will be modified by the user model constructed from the user’s feedback.

Step 6: After the search is finished, the current user model is backed up.

Step 7: Each participant is asked to go back to step 3 and run the next query on the list.

4.2 Metrics

Denote I_1 , P_1 , and C_1 as Interests, Preferences and Context generated by a human and I_2 , P_2 , and C_2 as Interests, Preferences and Context generated by the machine. I_1 and I_2 are two sets in which each element contains an interest concept. The similarity that determines to what extent I_1 is covered by I_2 after running each query is defined as :

$$sim(I_1, I_2) = \frac{(|I_1 \cap I_2|)}{|I_1|} \quad (1)$$

Interaction sheet

Query index	Query text
23	<i>distributed computing structures and algorithms</i>

After reading the content of the relevant document found in the first 15 documents, please determine your:

Interest: (concepts that best describe your focus on this query)

distributed computer network
distributed computing

Preferences: (methods that you would like to perform to get better results) (put \surd next to your choice)

\surd Filter Expander

Context: (relations that best describe your focus on this query)

topology information|related_to|distributed computer network|
distributed computer network|isa|network|

Figure 2: A sample of a user’s note taken during the experiment

where $|I_1 \cap I_2|$ is the number of common interest concepts shared by both I_1 and I_2 , $|I_1|$ is the number of interest concepts generated by a user. We compute the similarity between interest concepts generated by machine and generated by a user using two methods: exact match and flexible match. We removed stop words and applied Porter stemming algorithm for human and machine generated interests. In exact match, two interest concepts are considered matched with each other if and only if they are spelled exactly the same. In flexible match, each Interest set is converted into a vector in which each element is one single word. We compute the similarity by applying equation (1) for these two vectors. These two types of matches are needed because a user’s interests can be represented in a number of ways but a user may only list one way while participating in this experiment. By converting each Interest set into a vector of single words, we relax the syntactic constraints while still reinforcing the lexical constraints. The similarity of Interest sets between a user and a system over n queries is computed by taking the average of all values of $sim(I_1, I_2)$.

A user’s Preferences is represented in a Bayesian Network with a basic function to elicit a user’s preferences over a set of methods being used to modify a user’s query. In this experiment, human generated Preferences (P_1) is directly elicited by asking the user to choose which method he/she would use to modify a query. Therefore, P_1 is a set with a single element representing a method that a user would use to modify a query. To make it comparable, P_2 represents one method with the highest probability (*top 1*) determined by the model. The similarity between these two sets after each query

is defined as:

$$sim(P_1, P_2) = |P_1 \cap P_2| \quad (2)$$

where $|P_1 \cap P_2|$ is equal to either 0 or 1.

The similarity of Preferences generated by a user and Preferences generated by a system over n queries is computed by taking average of all values of $sim(P_1, P_2)$.

For the similarity measure between two Contexts, we borrow the lexical and taxonomy measures from the Ontology learning and Semantic Web communities as presented in [9, 10]. The sets of concepts (*lexical parts*) in C_1 and C_2 are L_1 , L_2 , respectively. The lexical parts refer to the concepts generated by either a real user or by our approach while constructing the corresponding Context. An example of L_1 from the human-generated Context network shown in Figure 3 would be “*topology information*”, “*distributed computer network*”, and “*network*”.

The average string matching algorithm is used to determine the extent to which L_1 is covered by

$$SM(L_1, L_2) = \frac{1}{|L_1|} \sum_{L_i \in L_1} \max_{L_j \in L_2} SM(L_i, L_j) \quad (3)$$

In which

$$SM(L_i, L_j) = \max \left(0, \frac{\min(|L_i|, |L_j|) - ed(L_i, L_j)}{\min(|L_i|, |L_j|)} \right) \quad (4)$$

where $|L_i|$ is the length of a string L_i , and $ed(L_i, L_j)$ is the edit distance which measures the minimum number of insertions, deletions and substitutions required to transform one string to another using dynamic programming.

The measure to compare two relation structures of two Contexts C_1 and C_2 is given by:

$$TO(C_1, C_2) = \frac{1}{|L_1^C|} \sum_{L \in L_1^C} TO(L, C_1, C_2) \quad (5)$$

where $TO(L, C_1, C_2)$ is the overlapping measure between two relation structures of C_1 and C_2 and is computed as follows:

$$TO(L, C_1, C_2) = \begin{cases} TO'(L, C_1, C_2) & \text{if } L \in L_2^C \\ TO''(L, C_1, C_2) & \text{if } L \notin L_2^C \end{cases}$$

where $TO'(L, C_1, C_2)$ is the ratio between the number of concept nodes in the intersections of the two Context and the number of concept nodes in the union; and, $TO''(L, C_1, C_2)$ is the maximum overlap given a fictive membership of a node in C_1 to C_2 (please see [9, 10] for more details about

these similarities). Note that we apply this similarity measure for all the nodes involving both “*isa*” and “*related_to*” relationships.

Similarly to Interest and Preferences, the similarity of Context generated by a user and Context generated by a system over n queries in terms of lexical and relation is computed by taking average of all values of $SM(L_1, L_2)$, and $TO(C_1, C_2)$, respectively.

5 Results and Discussion

We compile the entry questionnaires for all five users. The average number shows that all users are skillful in using search engines (average=1.8) but did not have much experience with IR applications using relevance feedback (average=4.0). All five users are relatively familiar with the search domains (average=3.4 for distributed computing and average=3.2 for optimization domains). All five participants indicated that when they search for information, they start with specific concepts relating to the search topic.

User	P	I	LC	RC
1	20%	7.77% (48.70%)	25.97%	3.19%
2	80%	17.96% (50.5%)	20.16%	2.44%
3	90%	33.3% (66.67%)	27.62%	9.06%
4	50%	45.8% (72.5%)	41.87%	15.22%
5	40%	19.7% (38.54%)	35.40%	10.22%
Avg.	56%	24.89% (55.38%)	30.2%	8.02%

Table 1: Results for 5 participants in the experiment

The results of this experiment are reported in Table 1. Note that in this table, P stands for Preferences, I stands for Interest, LC stands for lexical part of the Context, and RC stands for relation part of the Context.

The similarities of human-machine Interests (flexible match - 55.38%) and human-machine Preference (56%) represent the agreements between a real user and our approach in terms of short-term goals and methods used in an information seeking task. A user’s Preferences and Interests reflect his/her cognitive searching styles which have been found to influence the judgments of document relevance [5]. The similarities between a human generated user model and machine generated user

model in terms of Preferences and Interests fall into the range of the agreement among people in terms of relevancy assessment for a document (which is 40% to 75%, as reported in [19]). The similarity of human-machine Interest using exact match is 24.89% because: First, the interest concept described by a user may not be necessarily from the set of the retrieved relevant documents but from the user's previous domain knowledge (which has been confirmed by our informal exit interviews with all participants) while the ones generated by our approach created from the retrieved relevant documents.

Second, depending on a user's expertise, the user's descriptions of his/her interests may contain either non-technical terms or technical terms. Therefore, some concepts are captured in our model which are very close to what a user describes but are not exactly the same. For example: "*Less space*" is an interest concept generated by a user and "*space efficiency*" is an interest concept generated by the system. Another example is "*concurrent program*" is generated by a user and "*concurrent process*" is generated by the system. The similarity in both exact match and flexible match however only measures how similar two sets of Interests are at the syntactic and lexical levels. A quick computation using a subjective semantic similarity is done, in which if two Interest concepts are topically similar, they are considered a match. For example: "*less space*" is considered as a match for "*space efficiency*". The average similarity then is 64.9%. Our user model contains the words that have the same meaning with what a user is looking for. Therefore, when a query is modified proactively using our user model, it helps retrieve more relevant documents to each user.

The similarity of lexical is very similar to the results using the same metrics to compare two users' ontology structures described in [10]. The average lexical similarity for 4 users in the experiment done in [10] is 36% (ours is 30.2%). The low results for relation structure deserves more study on how to relax lexical and syntactic constraints for computing the similarity measure of two relation structures. The metrics used in this experiment measure the similarity at the lexical and syntactic levels. They can be considered as an exact match for Context because they only consider the direct neighborhood (grandparents and grandchildren of a concept node). Users are very good at connecting two concepts which may be remotely related with each other. Therefore, even though two concepts are included in our Context, but if they are

not directly connected, they are not counted as a match.

6 Conclusions

To recap, our method is to determine to what extent a user's model explicitly generated by a user is covered by our approach. Our results confirm our first hypothesis that at the lexical and syntactic levels, the human generated Interests, Preferences and Context are appropriately overlapped with the machine generated Interests, Preferences and Context. The degree of overlapping also agrees with the previous studies of agreement among people in terms of ontology and relevancy assessment.

There are several problems that we continue to focus on from the results of this work. First, the semantic similarity measures which provide flexible yet informative information about how similar two sets of Interests, or two Contexts are, need further investigation. Second, to collect enough data to verify our second hypothesis, we need to determine the appropriate amount of time a user needs to use the system to measure the long-term effect of the user model. Thirdly, five users are still a small sample size, we wish to increase the sample size to something larger to determine how the similarity has been changed. In addition, five users participated in this experiment happened to be a local focus when it comes to searching. It would help us to know if the similarity is changed for a group with different cognitive searching styles. Lastly, we found out that the size of human generated Context network affected by a user's tiredness. Thus, we would like to further investigate the effect of human factors such as pressure and fatigue on the process of relevancy assessment. This would help us to find the solutions to help the users in critical situations.

References

- [1] D. Billsus and M. Pazzani. A hybrid user model for news story classification. In *Proceedings of the 7th International Conference UM'99*, pages 98–108, 1999.
- [2] P. Brusilovsky, C. Karagiannidis, and D. Sampson. Benefits of layered evaluation of adaptive applications and services. In *Proceedings of Empirical Evaluation of Adaptive Systems workshop (held at UM01)*, pages 1–8, 2001.

- [3] D. Bueno and A. A. David. Metiore: A personalized information retrieval system. In *Proceedings of the 8th International Conference, UM'01*, pages 168–177, 2001.
- [4] A. T. Corbett. Cognitive computer tutors: Solving the two-sigma problem. In *Proceedings of the 8th International Conference, UM01*, pages 137–147, 2001.
- [5] D. Davidson. The effect of individual differences of cognitive style on judgments of document relevance. *Journal of the American Society for Information Science*, 28:273–284, 1977.
- [6] G. Fischer and Y. Ye. Personalizing delivered information in a software reuse environment. In *Proceedings of the 8th International Conference, UM2001*, pages 178–187, 2001.
- [7] D. Iglezakis. Is the act-value a valid estimate for knowledge? an empirical evaluation of the inference mechanism of an adaptive help system. In *Proceedings of the Fourth Workshop on the Evaluation of Adaptive Systems (held at UM'05)*, pages 19–26, 2005.
- [8] F. V. Jensen. *An Introduction to Bayesian Networks*. Univ. College London Press, London, 1996.
- [9] A. Maedche. *Ontology Learning for Semantic Web*. Kluwer International Series in Engineering and Computer Science, 2002.
- [10] A. Maedche and S. Staab. Measuring similarity between ontologies. In *A. Gomez-Perez and V. R. Benjamin (Eds.), EKAW 2002*, pages 251–263, 2002.
- [11] B. Magnini and C. Strapparava. Improving user modeling with content-based techniques. In *Proceedings of the 8th International Conference, UM'01*, pages 74–83, 2001.
- [12] H. Nguyen. *Capturing User Intent for Information Retrieval*. PhD thesis, University of Connecticut, 2005.
- [13] H. Nguyen, E. J. Santos, Q. Zhao, and C. Lee. Evaluation of effects on retrieval performance for an adaptive user model. In *Adaptive Hypermedia 2004: Workshop Proceedings - Part I*, pages 193–202, 2004.
- [14] A. Paramythis, A. Totter, and C. Stephanidis. A modular approach to the evaluation of adaptive user interfaces. In *Proceedings of Empirical Evaluation of Adaptive Systems workshop (held at UM'01)*, pages 9–24, 2001.
- [15] G. Salton and C. Buckley. Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41(4):288–297, 1990.
- [16] E. Santos, H. Nguyen, Q. Zhao, and W. Hua. User modelling for intent prediction in information analysis. In *Proceedings of the 47th Annual Meeting for the Human Factors and Ergonomics Society*, pages 1034–1038, 2003.
- [17] E. Santos, H. Nguyen, Q. Zhao, and E. Pukinskis. Empirical evaluation of adaptive user modeling in a medical information retrieval application. In *Proceedings of the 9th User Modeling Conference*, pages 292–296, 2003. Johnstown, PA.
- [18] E. J. Santos, Q. Zhao, H. Nguyen, and H. Wang. Impacts of user modeling on personalization of information retrieval: An evaluation with human intelligence analysts. In *Proceedings of the Fourth Workshop on the Evaluation of Adaptive Systems (held at UM'05)*, pages 27–36, 2005.
- [19] T. Saracevic. The concept of relevance in information science: A historical review. *Introduction to Information Science*, 1970.
- [20] P. Thagard. Cognitive science. *The Stanford Encyclopedia of Philosophy (Winter 2004 Edition)*, Edward N. Zalta (ed.), 2004.
- [21] P. Totterdell and E. Boyle. The evaluation of adaptive systems. In *In D. Browne and P. Totterdell and M. Norman (Eds.), Adaptive User Interfaces*, pages 161–194. London: Academic Press, 1990.
- [22] M. Virvou and B. du Boulay. Human plausible reasoning for intelligent help. *User Modeling and User Adapted Interaction*, 9(4):323–377, 1999.
- [23] A. Waern. User involvement in automatic filtering: An experimental study. *User Modeling and User Adapted Interaction*, 14(2/3):201–237, 2004.
- [24] S. Weibelzahl. *Evaluation of Adaptive Systems*. PhD thesis, Pedagogical University Freiburg, 2002.