Evaluation of the Quality of Authoring Systems *

Avram ESKENASI and Roumen RADEV
Software Engineering Department, Institute of Mathematics, Bulgarian Academy of Sciences, Academician Bonchev street bl.8, 1113 Sofia, Bulgaria

The quality of educational software is to a large extent decided by features and capabilities of authoring systems. An attempt has been made to evaluate the quality of several more or less known authoring systems with an easy-to-use evaluation instrument. The approach is based on a general method for software quality evaluation developed by the authors. A relatively large set of about 60 characteristics of authoring systems has been determined. Several well-known authoring systems have been screened on these characteristics. Each of these sample systems has been classified as excellent, good, or poor, according to the authors' views and on the basis of available references. Any new authoring system screened against the same characteristics can automatically be classified as excellent, good, or poor. By making use of properties inherent in the method, the set of characteristics used in the evaluation could significantly be reduced. By interchanging samples with authoring systems to be evaluated, as well as by other experiments, the reliability and stability of this approach were proved.

Keywords: Authoring system, Evaluation, Characteristics of authoring systems.

Avram Eskenasi was born in Varna (Bulgaria) in 1946. He received his M.Sc. in Mathematics in 1969 and Ph.D. in Mathematics in 1978, both from the University of Sofia. He is currently head of the Software Engineering Department at the Institute of Mathematics of the Bulgarian Academy of Sciences. He is the Bulgarian representative to IFIP TC 2 (Programming).

* The work presented here was partly supported by the Bulgarian Ministry for Culture, Science and Education under Contracts Nos. 36 and 390.

Education & Computing 5 (1989) 43–47
Elsevier

Introduction

The quality of educational software depends mainly on the software tools used to produce the software. Authoring systems probably form the most important class of these tools. Therefore, the higher the quality of the authoring system, the better the chance of developing good educational software. But the problem of evaluating the quality of authoring systems seems to be an open one: "there is not an extensive amount of published material available ... on the objective evaluation of author languages for CAI" [3]. The present authors hope that they have proposed an appropriate approach to this evaluation, based on a general method for software quality evaluation, in the development of which they have participated.

The Method of Quality Evaluation

In the following, a synopsis of the method is given; a more detailed description is given in [4]. Software quality is understood to be a set of characteristics of the software product or service under scrutiny that reflects the capacity of the product or the service to satisfy certain needs. Classical, hierarchical quality evaluation methods are rather subjective, expensive and cumbersome. In many cases, the exactness of the result of these methods is not required, and approximate information as to whether the product is excellent, good, or poor, will satisfy the user. On the other

Roumen Radev was born in Rousse (Bulgaria) in 1956. He graduated from the University of Sofia and received an M.Sc. in Mathematics in 1981. From 1981 till 1983, he was with the Soviet-Bulgarian Research Institute "Interprogramma". In 1987, he received his Ph.D. in Mathematics from the University of Sofia. Currently, he is a researcher in the Software Engineering Department at the Institute of Mathematics. His scientific interests are in the field of authoring tools, expert systems and their application in education.

hand, the professional only wants to know how the product is positioned in comparison with several already well-known products of the same type. These considerations have led to an attempt to develop a method for software quality evaluation based on pattern recognition theory. Let us consider a given type of software product (text editor, authoring system, etc.). For this type of product we determine a set of characteristics. Each characteristic can be assigned the values:

1 if the product does possess the corresponding property,
0 if it does not, and
x if no information is available.

Let us also suppose that there are several very well-known products of the same type and that for these sample products the value of each characteristic can be determined. Then, on the basis of the opinion of users and professionals, these samples are classified according to quality. In practice, two (good and poor), or three (excellent, good, and poor) classes are the most often used, but a higher number of classes is equally possible. The characteristics of all samples can be represented in a table, called the 'teaching' table. Let us now suppose that a new product E becomes available. For this new product we determine the value of each characteristic. The main idea of the proposed method is that, on the basis of the teaching table T and the characteristics of E, the new product is classified by various procedures into one of the predefined classes. From this classification follows information about the quality of E; for most purposes, this information is sufficiently detailed.

Authoring Systems

To produce educational software, teachers need special software tools—the so-called authoring tools. Two main groups of such authoring tools exist:

(1) authoring languages (AL), and
(2) authoring systems (AS) [6].

Authoring languages can be regarded as a family of specialized, higher order application languages, facilitating the production of courseware. Authoring systems are complex software tools, enabling authors with little or no knowledge of programming more easily to create courseware. In comparison to authoring languages, these systems have two additional functions: capabilities for course management and an author's environment. From the point of view of an end-user, authoring systems provide more facilities than authoring languages and support most of the aspects of the courseware producing process. That is why we have limited our investigations to authoring systems only.

The Set of Characteristics

The initial step in the application of the quality evaluation method is to determine a set of characteristics which describes authoring systems as exactly as possible. In order to compile such a set of characteristics, we examined various sources of information: we studied interviews with teachers who frequently use authoring languages or authoring systems, articles from scientific journals and conference proceedings, as well as the documentation of specific authoring languages and authoring systems. We adopted the following strategy: we tried to obtain as many characteristics relevant to authoring systems as possible, which were then introduced into our experimental quality evaluation program; our goal was to select from these characteristics a subset that would later be used for the evaluation of the quality of authoring systems. From our investigation of the available sources, we obtained a primary set of 63 characteristics relevant to authoring systems. We broke this primary set of characteristics down into 6 groups relating to the functions of the authoring system; the groups contain characteristics relevant to:

(1) Interaction with the author (9 characteristics).
(2) Interaction with the student (14 characteristics).
(3) The evaluation of student answers (7 characteristics).
(4) Individualization of education (3 characteristics).
(5) Provision of information (14 characteristics).
(6) Common software features (16 characteristics).

Because of the limited length of this paper, a full description of the characteristics cannot be

given here; such a description is available on request. To give some idea of what is meant, let us consider the third group of characteristics as an example. This group includes characteristics of the evaluation of student answers; possible evaluations are: true/false, multiple-choice, free format, on keyword, matching, numeric and evaluation of answers in the cursor position. It is assumed that an authoring system under consideration possesses a particular characteristic, if the courseware generated by this authoring system can accept and handle the type of evaluation specified. Each characteristic is assigned the value 1, if a particular authoring system possesses the corresponding property, and the value 0, if this is not the case. Some characteristics can be assigned more than 2 values. For example, the characteristic 'matching', belonging to the third group, can take on four values:

0 if the characteristic is not present,
1 if exact matching is available,
2 if only matching by prefix is available, and
3 if only matching by suffix is available.

For the sake of uniformity, in the context of this paper, all characteristics have been assigned the value 1 (if the corresponding property is present) or 0 (if not present).

The Teaching Table for the Samples

The next step was to determine the sample set of authoring systems to be evaluated. Using the data published in [1,2,5,7], we chose the following systems: PROCAL 2, MicroText, MICROTICCIT, Combat, Easytrain, MAS, SAM, TenCore, AERO, TopClass, Domino, Mentor II, AULA, Mumedala, IVL, Regency USE, Wise. Further on, we shall refer to these systems by using the numbers 1 to 17. These numbers do not imply correspondence to the order of listing above. Because of the fact that the running time of our program increases exponentially with the number of characteristics, we had to reduce the primary set of characteristics. On the basis of our own views, of the opinion of experts and of available references, we formed a working set of 36 characteristics. Each of the 17 sample authoring systems was described by means of a binary vector, such that the ith element of the vector corresponds exactly to the ith characteristic in the working set. This element can get either the value 1, if the ith characteristic is present in the authoring system, or 0, if this is not the case. All vectors together form the so-called teaching table.

We then defined three classes of quality: excellent, good and poor by selecting three well documented authoring systems as samples for each of the classes. This is probably the most subjective part of the application of our method.

Table 1
Results of quality evaluation of authoring systems (samples are not included)

Number of authoring system	Experiment 1	Experiment 2	Experiment 3
02	Poor	Poor	Poor
04	Poor	Poor	Poor
05	Poor	Poor	Poor
09	Poor	Poor	Poor
11	Excellent	Excellent	Excellent
12	Good	Good	Good
15	Poor	Poor	Poor
16	Good	Good	Poor

Experiments and Development

With our program we have carried out three experiments. By 'experiment' we mean the following:

- the setting up of the proper set of characteristics,

Table 2
Characteristics used in the first experiment after first reduction

1. Graphics available	*19. Network facility
2. Colour facilities provided	20. Route through course
3. True/false answer	21. Timing
4. Multi-choice answer	22. Scoring
5. Free format answer	23. Management system
6. Keyword answer	*24. Student responses
7. Match variants	*25. Modelling/simulation
8. Numeric answer	26. Help facilities for author
*9. Answer accepted from cursor position	27. Highlighting
10. Printer	28. Reverse video
*11. 35 mm slide projector	29. Underscore
*12. Light pen	30. Blinking/flashing
*13. Touch screen	*31. Multiple character set
*14. Mouse	*32. Zoom
*15. Bit pad/digitizer	*33. Rotation
*16. Audio tape	*34. Scaling
*17. Video tape	*35. Animation
18. Video disk	36. Windowing

Table 3
The final teaching table

Characteristics ⇒	Samples' binary vectors															Number of sample authoring system
	09	11	12	13	14	15	16	17	19	24	25	31	32	33	34	35	
Class of quality	1	1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	6
	0	1	1	1	1	1	1	1	1	0	0	1	1	1	1	1	17
excellent	1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	7
Class of quality	1	0	0	1	1	1	1	1	0	1	0	1	0	0	0	0	13
	1	0	1	0	1	1	0	0	0	1	0	1	0	1	1	0	14
good	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0	0	8
Class of quality	0	0	0	0	1	0	1	1	0	1	0	0	0	0	0	0	3
	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
poor	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10

- the selection of the samples and the creation of the teaching table,
- the running of the program to compute the subsequent evaluation,
- the running of the program in order to evaluate other authoring systems (non-samples),
- discussion of the results (and, if possible, reduction of the set of characteristics).

In all experiments we have used the same samples. For all experiments the results of the evaluation are summarized in Table 1. The first set of characteristics we used was the working set with 36 characteristics (see Table 2).

The first experiment showed that 11 characteristics are irrelevant in the evaluation. The explanation is that all authoring systems so far observed possess these properties (the corresponding positions in the binary vectors are 1). By removing these characteristics from the set, we obtained a new and reduced set of characteristics which we used in the second experiment.

A warning is in order here. If a new authoring system possesses all the properties corresponding to the characteristics removed, the method will work correctly. If, however, any of the characteristics removed is not a characteristic of this new authoring system, the final evaluation results could be affected. The construction of a new teaching table, including the last authoring system as a sample, could become necessary.

The second experiment produced results similar to those from the first one (Table 1). We decided to go on removing characteristics from the working set as long as the results were not affected. Using weights, determined by the evaluation program for each characteristic, step by step we removed the characteristics with the least weights. In this way, the third and final working set of 16 characteristics (marked by an asterisk in Table 2) was obtained and the last experiment carried out. The final teaching table appears in Table 3.

The results again proved to be very similar to those from the previous experiments, except for authoring system numbered 16. Please note that in Table 1 it is only authoring system number 16 which has different evaluation results in one of the experiments.

Conclusions

By following the strategy described, we obtain a set of characteristics useful for the evaluation of the quality of authoring systems. It is our opinion that these characteristics make a clear distinction between the three quality classes possible. It is crucial to note that the characteristics belonging to the final set are not necessarily the most important ones; these characteristics only 'describe' the differences between classes in a better way in comparison to the other characteristics. For this reason, the final set (Table 3) might appear to be somewhat strange. The method can be used for the evaluation of the quality of authoring systems at different stages of their development. Please remember that each system is described by means of a binary vector. By varying the ones and the zeros within the vector, a clear picture of different versions of an authoring system can be obtained. Thus it is possible to evaluate the quality of each version of the authoring system and to decide on the most appropriate one.

References

[1] *The Authoring Language AERO, User's Guide* (Sofia University, 1987) (in Bulgarian).

[2] Ph. Barker, "MUMEDALA—An approach to multi-media authoring", In: *Proceedings of the 4th Conference on Computers in Education,* IFIP, Norfolk, Virginia (North-Holland, Amsterdam, 1985) 165–172.

[3] Ph. Barker and H. Yeates, *Introducing Computer Assisted Learning* (Prentice-Hall, Englewood Cliffs, NJ, 1985).

[4] A. Eskenasi, "Evaluation of software products quality by means of classification methods", *Journal of Systems and Software,* to appear, 1989.

[5] "Factors influencing the development and use of technology based training", In: *Technology Based Training*, Infotech State of the Art Report 14:8 (Pergamon Infotech Ltd., 1986) 229–249.

[6] G. Kearsley, "Authoring systems in computer based education", *Communications of the ACM* 25 (7) (1982) 429–437.

[7] R. Radev, *Expertise and Informativeness in Computer Based Education,* Ph.D. Thesis (Sofia University, 1987) (in Bulgarian).