ACCESS: CONTEMPORARY ISSUES IN EDUCATION
1998, VOL. 17, NO. 1, 28–35

ACCESS

Multimedia in education: An 'x-model' for educational multimedia selection

Jessie Wong Yuk Yong & Victoria Y. Hsui

ABSTRACT
In the drive to implement IT in the classroom, together with the vast amount of software that are currently available in the market, educators are faced with the daunting task of selecting quality software that will add value to learning. How do they identify, evaluate, and select educational software so that the process will result in the acquisition of software that offer optimal value to the curriculum and the institution? This paper focuses on this question by discussing a model, called the 'x-model', for educational multi-media selection. This model, developed with the user as central to the whole process, is an all-purpose, comprehensive model to which users can refer when called upon to select software for use in educational settings.

Introduction

Research has pointed to the benefits of using multimedia resources in teaching and learning. Matthew (1997) investigated two groups of students concerning their comprehension of texts in ordinary print and in CD-ROM format. The results showed that there was no statistically significant difference in their comprehension but there was a significant difference in the ability to retell the story. The CD-ROM format had a positive impact on students' ability in story retelling. Reinking and ChanLin (1994) presented a theoretical framework that described crucial differences between electronic and print texts based on how they were written and read. With electronic texts, the computer controlled the readers' access to the text by limiting the amount of text that was displayed and determining the pace at which the text was displayed. Electronic texts also provided interaction between the student and the text which printed texts were unable to provide. The nonlinear presentation and graphics were powerful ways to motivate learning and to retain what was learned. Because of these differences, they argued that the change in the way materials were presented had an impact on the readers' comprehension of the text. Other studies which also showed that students had better comprehension when reading electronic texts were done by Harper and Ewing (1986), Miller, Blackstock & Miller (1994). In addition, Cockerton and Shimell (1997) observed that children preferred using hypermedia presentations and rated them more positively in terms of ease of use than traditional print format. All these studies indicated that multimedia materials were able to aid learning and therefore they should provide a valuable resource for pedagogical practices.

In the drive to implement IT in the classroom, and supported by research which shows the advantages of using multimedia materials to complement classroom teaching, schools are making concerted efforts to use multimedia resources in classroom pedagogy. Computers have become standard equipment in schools, used to instruct, disseminate content, practice skills, diagnose

learning problems, administer tests, and enrich learning experience through a wide range of educational software. The greatest challenge for teachers lies in the selection of quality software for teaching and learning. How do they identify, evaluate, and select educational software before introducing them for curricular use? What criteria should they use to discriminate the excellent ones from the ones that are merely satisfactory or even mediocre? Is there a selection model that is suitable for all subject areas across the curriculum? This paper focuses on these issues and proposes a model, called the 'x-model' for educational multi-media selection. This model, developed with the user as central to the whole process, is an all-purpose model designed to optimise selection and to add value for money in the purchase and use of educational software.

The 'x-model'

The first step in the 'x-model' (see Fig. 1) consists of the examination of publishers' catalogues for updated information on new software available in the market. Most people search publishers' catalogues to locate courseware that will help meet their objectives. From these catalogues, educators can filter out those which obviously do not match their purposes. Once they have identified a software that seems to match their purposes, they should, preferably, check for independent reviews in journals or magazines or on the internet to see if a review has indeed been made of that particular software. These reviews usually provide input from the users' point of view and feedback on actual use which may be quite different from the publishers' claims or descriptions. Critical reviews are an excellent source of information in the search for quality courseware. The best reviews are those written by educators with experience in the subject area and also with computers. They often include information on field-testing the software with students as well as a summary of the reviewer's opinion. They provide a very useful resource for selecting promising courseware.

Not all software packages, however, are independently reviewed. Therefore in most cases, one will need to carry out one's own informal evaluation which is the next step in the 'x-model.' This process consists of the following steps: first, check for hardware compatibility; second, carefully examine the instructional design; third, examine the programme design. (see sections below for detailed description of each item). These processes should enable the educator to filter out a software which is incompatible with existing hardware or has a poor programme design. Once an appropriate software is identified and before making the decision to purchase it (the 'Decision Box,' Fig. 1), consider another filter in the model, 'Individual Choice.' This is an important consideration because the user is central to the use of the software. No matter how good a software seems in its instructional or programme design, the user/users must feel comfortable with it and believe in its efficacy for learning and teaching purposes. In addition, it is very important for the user to like the software. A dislike means that the programme cannot or will not be used optimally and may even become a white elephant. Once this filter is cleared, the user should decide on the advantages of purchasing the software for a single subject or for sharing among the curricula subjects (see section below for detailed considerations). Once this is done, the user can quite confidently reach a decision to purchase the software.

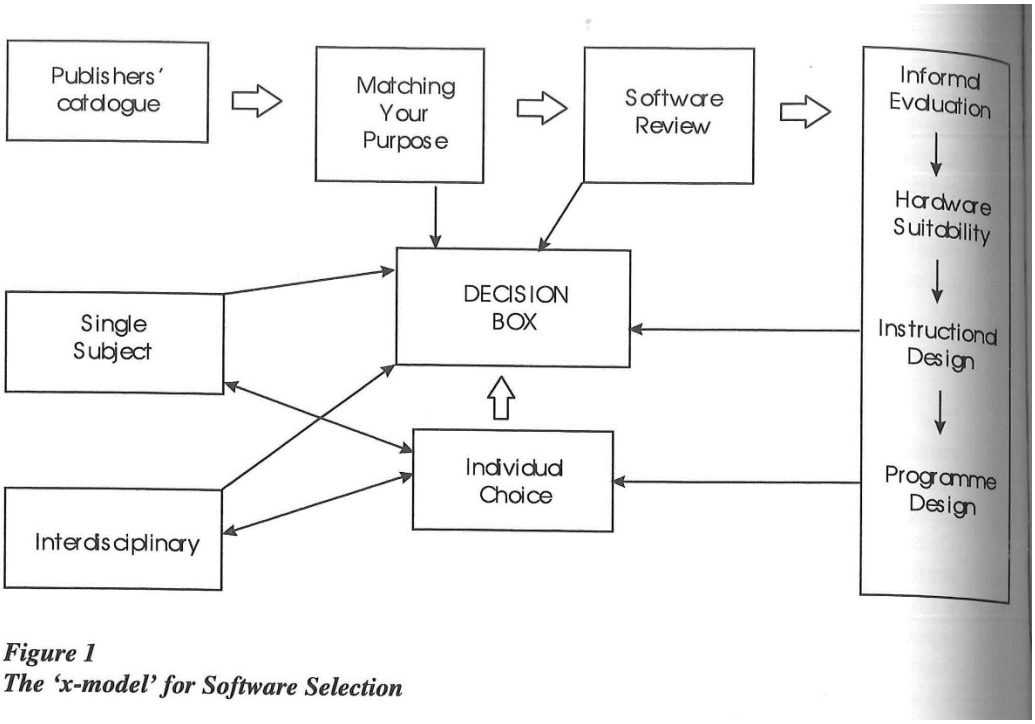

Figure 1
The 'x-model' for Software Selection

The sections that follow discuss in some detail various aspects of the 'x-model'.

The user

The user is a commonly forgotten factor in the process of educational software evaluation. Whether the software would have a positive impact on learning depends on the user - the teacher and/or the students. Whether a teacher feels that the programme actually teaches and helps to improve learning is an important question he/she should address before choosing a software. If the teacher feels uncomfortable with any aspect of the software, he/she should not consider buying it. Where critical reviews are not available and teachers do not have time to carry out a formal testing of each software before they purchase it, they should use their experience and personal judgement to make a decision. In doing so, teachers should ask some vital questions: Will the programme have a positive impact on learning? What does it teach specifically? What are the knowledge and skills that students would actually learn from using the programme? What else can students learn from it compared to another mode of learning? With experience in educational software evaluation, a teacher should be able to answer these questions reasonably well, by reference to the fact that the human factor is as important as any other factor in educational software evaluation.

Once a preview is done, the educator must determine whether the courseware really has a potential usefulness in the school. It is much easier to make good software decisions if an overall plan for computer use has been developed and needs priorities have been established. Pertinent questions to ask of each software, in line with an overall plan for computer use, include the following: does it fit into the curriculum? will it meet either the general needs of many students or the special needs of a smaller group of students? will the users (teacher and students) likely be comfortable with it? An excellent courseware, for example, can be a waste of money if it has minimal application to general and specific instructional objectives or if it does not accord with the taste of the users.

Hardware suitability

When selecting a courseware for preview, check whether it is compatible with computers that are available in the institution for which the software is being purchased. If it does not, there is no point in proceeding with the preview. Often a programme written for one computer model will not run on another model. However, some publishers have developed multiple versions for different systems. If this is so, there is a need to check if the institution's computers have all the necessary accessories and enough memory to run the programme. Request for a preview copy only after ascertaining compatibility. If a preview copy is not available, a demonstration of the courseware can be arranged with the vendors. Another preview technique is to visit computer fairs and company demonstrations of courseware. In schools where hardware selection has yet to be made, such demonstration/preview sessions are especially useful in providing additional data upon which to base the decision to purchase the hardware.

Software content and design

When teachers select books for educational use, they judge the quality and accuracy of the information presented. In the same way, the quality of the content of the computer/multimedia courseware is an important criterion for evaluation. The content of the software must be accurate, correct, up-to-date, complete and appropriate for the target group. Occasionally courseware contains spelling mistakes or grammatical errors due to poor editing or hasty production. Such courseware should not be used by students. Content should also be presented in a logical, clear and objective manner, and be free from bias and stereotyping.

The content of a software, however, is not the only criterion for determining whether it is worth buying or not. The content may be excellent but if there are no interactive features for learners to respond to, the software would be little different from a printed book except that it is presented in a digital form. Therefore the process involved in evaluating a computer/multimedia software is much more complicated than that of evaluating a textbook or any printed material. The following are three criteria for evaluating content and design:

1. Instructional Design
2. Programme Design
3. Documentation

Instructional Design

Is the package well designed from the educational viewpoint of what constitutes learning? What are the objectives? What is the procedure used to achieve these objectives? Is there any assessment instrument to test understanding? Is it suitable for your purpose and your students?

Instructional design refers to the approach on which the design of learning materials are based in order to optimise learning under certain prescribed conditions. It covers the purpose of the software, the target population, the learning approach, the development and the sequencing of the content. Each software is designed for a specific purpose. Therefore a software designed for self-learning is not suitable for a teacher-directed lesson; a software developed for remedial learning is not suitable for use in regular classrooms, and a software developed for an advanced level is not suitable for the elementary level.

On the other hand, there may not be any software in the market which is designed for a specific course. In this case, the teacher needs to check how well a software matches the objectives of his/her course. Some software are designed to teach specific skills; others are designed to teach knowledge or both. That is to say, if a course emphasises knowledge, a knowledge-based software should be

chosen. If a course emphasises the learning of skills, for example, communication skills in English or mapping skills in social studies, a skill-based software can be suitable. In this case, it is important to examine the scope and sequence of the skills presented in the software. Does it meet the standard prescribed for effective learning? Sometimes the teacher needs a software which provides a balance of skill and knowledge components. In this case, he/she should check that the materials are presented in small, well-sequenced units and each unit contains useful examples. There should be tasks for students to practise the skills taught. There should also be immediate feedback for each response. If the response is incorrect, additional tasks and feedback should be available. Explanation should be given for each incorrect response.

While useful or good lesson designs may vary, the user should check that the programme contains the basic educational steps, with sufficient supporting materials to teach the intended content and/or skills. It is also important that all instructional materials are well-written. They should present a model of good writing to students in terms of clarity, conciseness, syntax and structure. Poorly written software should not be considered.

Programme Design

A useful educational software should have good instructional design as well as good programme design. Cockerton and Shimel (1997) in their experiment to evaluate a hypermedia document as a learning tool, discovered that information presented in the form of computer texts, graphics and images, including video clips are much more preferred by children than what is available in textbooks. Children also showed a higher level of motivation in learning from the hypermedia material than from the textbook because of the programme design - the capability of the hypermedia material to interact with the user, an important feature which is lacking in printed material. In any case, a software with a good programme design should be easy to use, be instructionally sound and should take full advantage of the unique capabilities of the computer - interaction, branching, graphics, and sound - and be free from error or bugs. When evaluating the programme design of any software, at least consider the following aspects:

i. Instructions for using the software

There should be clear instructions on how to use the software. The instructions should suit the level of the intended user and should be self-explanatory. The ideal programme design should allow the user to skip this page if he/she wants to.

ii. Help screen

A good programme should have an adequate amount of information in the help screen and should be user friendly. One should be able to gain access to the appropriate help screen at any point in the programme with ease. A good help system does not interrupt the lesson flow. One should be able to return to where one left off without difficulty.

iii. Flow of the programme

Some programmes crash because of a faulty input. For example, instead of entering '2', a user enters 'two'. This is not acceptable, thereby causing the programme to crash. A good programme should contain 'traps' to catch any wrong entry. In this case, a message should appear requesting a re-entry with only numbers like: 'Please type in your answer. Use numbers only.' Even though traps are ideal for error-checking, many programmes do not provide this feature. In some cases, it is not apparent

that an error has been made until the user progresses further into the programme and get caught in it. This type of software should be avoided if at all possible.

iv. Communicating with the programme

A programme can be command-driven or menu-driven. Most new software are menu-driven. It has the advantage of being easier to use. But there can be problems too. One common problem is that the user is not given the choice to leave the programme at the point he/she wants to do so. Another problem is that the user must start from the beginning with the initial instructions each time the programme is used. A good design should allow users to quit and to enter at several points of the programme, and to move along easily. For younger children, a menu-driven programme is preferred because they do not have to remember the command as they move along the programme.

v. Screen display

Screen display refers to the layout of the text on the screen. Good programmes should provide a neat, well-spaced screen that facilitates reading. Moving from one screen to the next should be easily controlled by the user. The characteristics of a poor screen display include a full screen of text with single spacing that scrolls off the screen rapidly, or text containing too many flashing words. A good screen display includes proper centering and spacing of materials, along with a procedure for self-pacing such as 'press return to continue.'

vi. Student Interaction

The most valuable aspect of computer instruction is its ability to interact with the student. In a well-designed programme, student input determines the next computer output. Different student input should lead to different output. Entering and exiting the programme should be quick and efficient. Students who have used the programme before should be able to move quickly to the point at which they ended the previous session. If they prefer, they should be able to skip the pages which they do not wish to read and arrive at the relevant page without delay and difficulty. Another point to consider is that the system response time is short and the user feels comfortable interacting with the programme.

vii. Branching

Branching is the ability of the programme to allow the user to follow different tracks based on the students' ability level and choice. It also aids the user in locating the required information with ease. A good educational software should provide branching capability.

viii. Graphics and sound

Graphics and sound can be used to facilitate learning. Graphics should be used to demonstrate a concept, with the right colour and animation added to motivate the learner. Sound is an attractive feature to reinforce, focus or prompt students. Nevertheless, it can act as a distraction as well. Therefore a programme containing a sound option must allow the user to raise or lower the volume or to switch it off completely at any point in the programme.

The overall programme design is an important aspect of the software, just like the lesson design is an important aspect of the lesson. A software with good content material will not be a better learning tool than the printed textbook unless it takes advantage of the computer capabilities in its design to facilitate learning.

Documentation

Documentation refers to the guides, manuals, and other written material that tell the user how to use the software to facilitate learning. An effectively written documentation for educational software should include three parts. The first part should describe hardware and software installation and start-up instructions to begin the programme. It should cover the required hardware, DOS, programming language and any unusual hardware or software requirements. The second part should describe the precise educational goals of the programme and how to use the programme to achieve the desired goals and objectives. The third part should give step-by-step instructions necessary for the instructor and student to get the most benefit out of the programme. It should also include accurate background and supplemental information. Some documentation for educational software includes lists of additional resources, suggested activities and complementary materials for both the teacher and the student. Software that comes with a student manual is very helpful. If not, the teacher should extract the main points from the main manual for the students. Some manuals contain extra information about the instructional materials such as the age or grade level the materials are suitable for and the entry skills and knowledge required to use the programme effectively. They also outline the units of learning in the programme with statements about their goals, instructional objectives and the scope covered. The purpose of the unit should be clearly stated, whether it is designed for drill and practice, for tutorials, for instruction, as a demonstration to introduce a new area, or as an avenue to assess achievement. These manuals are very useful to the user - the students and the teacher. For any manual, easy access to needed information is vital. Therefore materials should be well organised for easy referencing.

Single subject or interdisciplinary

Educational software publishers normally categorise software under different subjects in their catalogues. It is vital for a teacher not to just look under his/her subject area. This is because there is no standard way of cataloguing. Different publishers use different subject headings. For instance, Mentorom Multimedia' s catalogue uses the *Science and Nature* category to house the geography and social studies software. Many software packages are made to contain sufficient materials to teach across the disciplines. Some educational software packages have excellent content for geography but because of the programming and instructional designs, they can also be used for language teaching. For instance, *Dwindling Resources and The environment: Conservation* are two specifically geography software. However, their text, photographs, video, illustrations and audio effects make them good resources for language teaching as well. These software can be shared across the disciplines. When there are limited funds, sharing software across the curriculum is a way to maximise the use. When there are unlimited funds, sharing software is fun and is a means to help students to build a link across subjects. Therefore interdisciplinary software have more value for money than strictly subject-based software.

Conclusion

In the current climate of enhancing the IT presence in schools and using multimedia software as a tool for teaching and learning, educators are faced with the task of selecting appropriate software, with a view to maximising learning and putting monetary resources to their best use. To many educators, selecting software is a new and daunting task. Complicating this task is the large number of software available in the market, each one competing for attention and sales. The situation, as it stands, calls for the design of a clear, comprehensive model to which users can refer when called upon to select software for use in schools. It is with this objective in mind that the 'x model,' discussed in this paper, has been designed. It provides a simple, useful and practical way to identify educational software which would add value to learning and maximise the potential that IT has to offer in pedagogy.

References

COCKERTON T. & R. SHIMELL (1997) 'Evaluation of a hypermedia document as a learning tool', *Journal of Computer Assisted Learning,* 13, pp. 133-144.

HARPER, J.A. & EWING, N.J. (1986) 'A comparison of the effectiveness of microcomputer and workbook instruction on reading comprehension performance of high incidence handicapped children', *Educational Technology,* 26(5), pp. 40-45.

LATHROP A. & GOODSON, B. (1983) *Courseware in the Classroom.* USA: Addison-Wesley.

MATTHEW K. (1997) 'A comparison of the influence of interactive CD-ROM storybooks and traditional print storybooks on reading comprehension', *Journal of Research on Computing in Education,* 29(3), pp. 263-275.

MILLER, L., BLACKSTOCK, J., & MILLER, R.(1994) 'An exploratory study into the use of CD-ROM storybooks', *Computers in Education,* 22(1/2), pp. 187-204.

REINKING, D. & CHANLIN, L.J. (1994) 'Graphic aids in electronic text', *Research and Instruction,* 33(3), pp. 207-232.