An Approach to Natural Language Equation Reading in Digital Talking Books

No Author Given

No Institute Given

Abstract. Mathematic equations are, of necessity, a must in any mathematic textbooks but also in physics, communications and, in general, in any technology related texts. Furthermore, their usage in Digital Talking Books (DTB) can be eased if its corresponding counterpart in both text and/or spoken forms can be automatically generated. Therefore, an automatic system to translate or convert them into text and latter to speech is needed to broaden the scope of the DTBs. In this paper we address the implementation of a "translation" system that converts mathematical equations into text in such a way that it resembles as much as possible the "natural" reading of those entities. The system we implemented is more than just a translator from some form of mathematical notation into text, since reading heuristics were included. The goal is to mimic as much as possible the "usual" way someone reads equations. The reading heuristics can, and sometimes do, lead to ambiguity in the meaning or in the generated text. But, the naturalness of the results can be a better option when the reader is, at least, vaguely familiar with the topics. Evaluation results, although preliminary, are presented and surely validate our approach.

Key words: MathML, Digital Talking Books, Speech Alignment

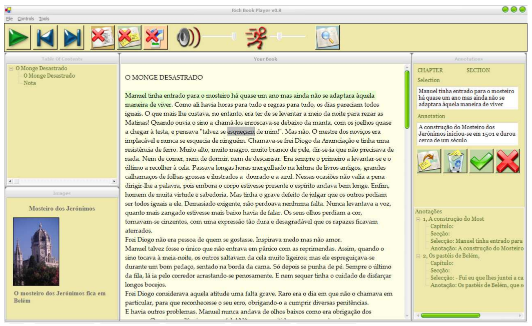
1 Introduction

Digital Talking Books (DTBs) are complex entities that comprise several types of data: text, images, sounds (voice, music, etc) and other metadata in a structured framework. This framework, not only allows the visualization and navigation of the data, but also provides search and indexing capabilities within the digital book through, at least, its text. In previous projects we developed a framework [1] where text and audio could be time-aligned in such a way that the user can, for instance, locate some word(s) or sentence(s) in the text and listen immediately to their corresponding recorded audio. Since there is no standard way to read tables, graphs and figures, we restricted the scope of those DTBs to novel, poetry, fiction, children's stories and didactic text books. However, there is a broad consensus that DTBs are an important learning tool and, therefore, mathematics and other exact or applied science textbooks can not be excluded. So, this work describes our efforts to complement DTBs with an automatic system to convert mathematical equations to text or voice. As mentioned, the audio file in a DTB is a recorded version of the text. Therefore, if the book contains mathematical equations, the speaker must, in the recording session, visually interpret the equation and only after that he/she shall begin reading it. This procedure is feasible for speakers that are confident with mathematics. Therefore, we devoloped a system to automatically convert equations into text, alleviating the technical requirements of the speaker. The "translation" system was developed for Portuguese but can be easily used with other languages, since operators, variables and other functions are specified in configuration files.

This paper is organized as follows: section 1 is devoted to the Introduction, where a brief description of the DTBs (in the next subsections) is included; in section 2, we present a brief description of the Mathematical Markup Language (MathML); in section 3 the translation system is described and its evaluation is presented in section 4 and Conclusions are presented in section 5.

1.1 Digital Talking Books

DTBs are based on different types of data, structured according to some standard. They also require a player or browser that allows users to navigate, to index and to retrieve information (text, sound, images, etc.). The player was developed using a model based framework for adaptive multi-modal environments [2]. Besides supporting the features described in the DTB standard[1], the player introduces features complementing the synchronized presentation of text and audio, such as: addition of content related images; variable synchronization units, ranging from word to paragraph; annotation controlled navigation; definition of new reading paths; adaptation of the visual elements; behavioral adaptation reflecting user interaction, amongst others. Some of these features are visible in Figure 1.



Mosteiro dos Jerónimos

O mosteiro dos Jerónimos fica em Belém

O MONGE DESASTRADO

Manuel tinha entrado para o mosteiro há quase um ano mas ainda não se adaptara àquela maneira de viver. Como ali havia horas para tudo e regras para tudo, os dias pareciam todos iguais. O que mais lhe custava, no entanto, era ter de se levantar a meio da noite para rezar as Matinas! Quando ouvia o sino a chamá-los enroscava-se debaixo da manta, com os joelhos quase a chegar à testa, e pensava "talvez se esqueçam de mim!". Mas não. O mestre dos noviços era implacável e nunca se esquecia de ninguém. Chamava-se frei Diogo da Anunciação e tinha uma resistência de ferro. Muito alto, muito magro, muito branco de pele, dir-se-ia que não precisava de nada. Nem de comer, nem de dormir, nem de descansar. Era sempre o primeiro a levantar-se e o último a recolher à cela. Passava longas horas mergulhado na leitura de livros antigos, grandes calhamaços de folhas grossas e ilustrados a dourado e a azul. Nessas ocasiões não valia a pena dirigir-lhe a palavra, pois embora o corpo estivesse presente o espírito andava bem longe. Enfim, homem de muita virtude e sabedoria. Mas tinha o grave defeito de julgar que os outros podiam ser todos iguais a ele. Demasiado exigente, não perdoava nenhuma falta. Nunca levantava a voz, quanto mais zangado estivesse mais baixo havia de falar. Os seus olhos perdiam a cor, tornavam-se cinzentos, com uma expressão tão dura e desagradável que os rapazes ficavam aterrados.
Frei Diogo não era pessoa de quem se gostasse. Inspirava medo mas não amor.
Manuel talvez fosse o único que não entrava em pânico com as reprimendas. Assim, quando o sino tocava à meia-noite, os outros saltavam da cela muito ligeiros; mas ele espreguiçava-se durante um bom pedaço, sentado na borda da cama. Só depois se punha de pé. Sempre o último da fila, lá ia pelo corredor arrastando-se penosamente. E nem sequer tinha o cuidado de disfarçar longos bocejos.
Frei Diogo considerava aquela atitude uma falta grave. Raro era o dia em que não o chamava em particular, para que reconhecesse o seu erro, obrigando-o a cumprir diversas penitências.
E havia outros problemas. Manuel nunca andava de olhos baixos como era obrigação dos

Selection: Manuel tinha entrado para o mosteiro há quase um ano mas ainda não se adaptara àquela maneira de viver

Annotation: A construção do Mosteiro dos Jerónimos iniciou-se em 1501 e durou cerca de um século

Anotações
1, A construção do Most
Capítulo:
Secção:
Selecção: Manuel tinha entrado para
Anotação: A construção do Mosteiro
2, Os pastéis de Belém,
Capítulo:
Secção:
Selecção: - Fui eu que lhes juntei a ca
Anotação: Os pastéis de Belém, que s

Fig. 1. The DTB player interface

[1] www.niso.org/standards/resources/Z39-86-2002.html

1.2 Alignment of DTBs

The main drawback of DTBs is their construction process, specially if they are derived from non-structured data. In either case, the audio files must be recorded in good acoustic conditions and, preferably, by professional speakers. Those speakers are provided with the written form (text) of the DTBs that they must read in a clear and natural way. Then, the audio files undergo manual editing to remove reading errors and extraneous noises (e.g. lip noises, breathing). Figure 2 depicts the production procedure of a DTB. Our decoder is based on *WFSTs* [3] in the sense that its search space is defined by a distribution-to-word transducer that is built outside the decoder. That search space is usually constructed as $H \circ L \circ G$, where H is the *HMM* or phone topology, L is the lexicon and G is the language model. For alignment, G is just the sequence of words that constitute the orthographic transcription of the utterance. The output of

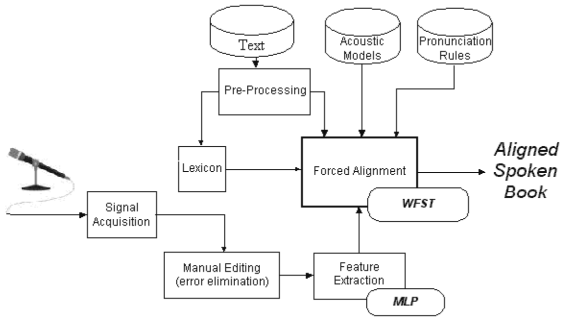
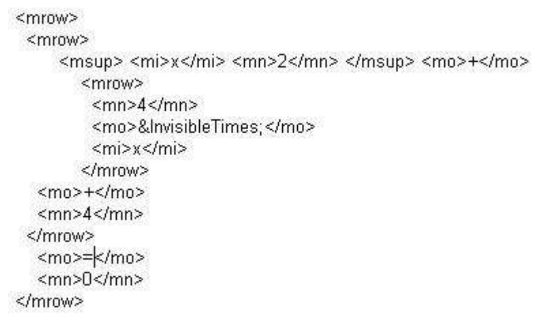


Fig. 2. Alignment of DTBs

the alignment system is a text file comprising the time instances corresponding to the alignment units (phonemes, words or groups of words) from the alignment process. This alignment system proved very robust even for aligning very long recordings: a 2-hour long book could be aligned in much less than real-time.

2 MathML

The Mathematical Markup Language (MathML) was chosen in this work over other existing alternatives, such as LaTeX and Microsoft Word that, besides being able to display mathematical formulas in a correct way, are also widely used. LaTeX is the *de facto* standard for the communication and publication of scientific documents [4]. It uses a high-quality typesetting system, but it is not made for Web presentation. The great advantage of LaTeX and his typesetting system is mainly the quick and easy production of documents. LaTeX was not meant for integration (other than with documents of the same type), or to be parsed by exterior applications (except for its own compilers). Microsoft Word is also a *de facto* standard for documents. But MSWord has also several problems, the first of them is being a proprietary binary file format and, as a consequence, it is not easy to retrieve information from it. Until recently (Feb 2008) [5], the only way to extract information was by reverse-engineering MSWord files. Another problem is that MSWord is made for "What You See Is What You Get" editors, so it does not focus on the information that it contains but rather in the way it displays that information. The MathML [6] is is a open standard and is a XML derived format, and is easily integrated in applications. Furthermore, it also can be displayed inside browsers and others applications that can display XML documents. It is easily read from external applications, since a XML parser will parse it correctly. The information is well structured and, therefore, easy to be read. As with XML and other derived formats, MathML files can grow in size faster than the data or the information it contains, since it requires multiple tags [7]. MathML can be used to both display the mathematical content or to represent the content of the mathematical formulas. This is the reason why the standard defines the *display* and *content* methods for the creation of MathML. In this work, we opted for the *display* method, since is used for displaying formulas in a browser, while the *content* method is to be used inside applications that need the description of how a formula is made and how functions inside the formula apply to each other. MathML is composed by a set of tags that are defined in the standard. These tags are used to define the type of the content inside the tags (numbers, variables, operators, etc...), or some functions to define the organization and structure of the formula. As an example, consider the simple formula, $x^2 + 4x + 4 = 0$ and its corresponding MathML description:



```
<mrow>
 <mrow>
      <msup> <mi>x</mi> <mn>2</mn> </msup> <mo>+</mo>
         <mrow>
          <mn>4</mn>
          <mo>&InvisibleTimes;</mo>
          <mi>x</mi>
         </mrow>
   <mo>+</mo>
   <mn>4</mn>
  </mrow>
   <mo>=</mo>
   <mn>0</mn>
</mrow>
```

The easiest way to implement a translator is to convert the MathML tags directly into their mathematical counterparts. And, in fact, this simple equation is, in most currently available "translators", converted into text as: "*eks* to the power of begin exponent two end exponent plus four times *eks* plus four equals zero". But, thankfully, no one speaks or reads such simple equations this way. Easier forms, such as: "the square of *eks* plus four *eks* plus four equals zero" are commonly used either by teachers or students. However, this sentence holds different meanings which leads to a conflict of notation and, therefore, mathematical meaning. To avoid ambiguity, punctuation marks should be included: "the square of *eks*, plus four {times} *eks*, plus four, equals {to} zero" to explicitly delineate the scope of mathematical operators (curl brackets represent optional terms.) We will address this topic latter in this work.

3 Math Translation

Although MathML is a standard, it evolves from time to time and, as a consequence, our translation system must be easily modified to cope with those updates. This requirement can be addressed more easily by a modular design of the translation system. Therefore, the program is divided in 3 blocks, figure 3, the Input Parser, the Math Dictionary and the Output Generator. The first two blocks comprises the Dictionary module. The Input Parser extracts the in-

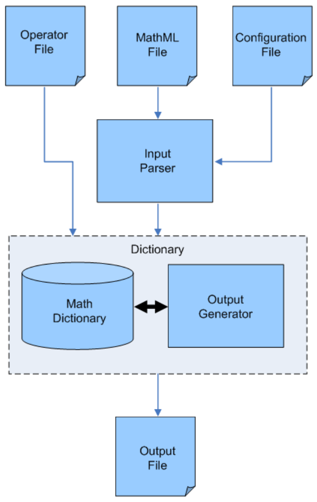


Fig. 3. Translation System diagram

formation from the input MathML file and also from the configuration file and, creates the correspondent data structures to the Dictionary module. This module process the data from the data structures and applies heuristics to improve the readability of the output. This operation requires some cautions, namely all data structures must be built around the MathML tree structure already created and, as such, can be processed recursively. However, since the same operators can be either nodes or leaves in the tree structure, there is no unique way of assert operator precedence. Therefore, to overcome such problems, special reading heuristics were developed. The Math Dictionary block signals the output generator where heuristics should be placed in the output text. In this way, we try to guarantee the translation of the mathematical equation is as similar to the natural language as possible, without impairing its comprehension. The output generator also adds punctuation marks and normalizes any digits to the corresponding numerals.

3.1 Reading Heuristics

Most of the mathematical operators are directly translated into written text. However, some of the most used ones need further processing, namely power, fraction, derivative and matrices. This processing avoids the direct translation of the operators, since no one reads a *simple* fraction like "fraction, *begin numerator ... end numerator* divided by *begin denominator ... end denominator* end of fraction" but rather like "*numerator* over *denominator*". Of course this may lead, in some cases, to ambiguity but, the gained naturalness may compensate this problem. In the following, some reading heuristics that were implemented to increase the naturalness of the conversion will be presented.

Fractions pose the problem of identifying what is the numerator and what is the denominator. To address this problem, fractions were divided in two categories, *short* or *simple* and *long* fractions. *Short* fractions are defined by having a reduced number of symbols in the denominator and/or in the numerator (a configurable threshold of three was chosen); fractions that are not *short* by this criteria are considered as *long*. For those, both numerator and denominator are explicitly written to avoid ambiguity. For example, consider the fractions:

$\frac{5}{100}$ that is converted into "5 over 100" instead of "fraction begin of numerator 5 end of numerator begin of denominator 100 end of denominator", and

$\frac{1}{y+bx+c}$ that is converted into "fraction, numerator 1, denominator y plus bx plus c". instead of "fraction begin of numerator one end of numerator begin of denominator y plus b times x plus c end of denominator".

Powers have a powerful impact on the reading of an equation. For example, x^2 should be converted to "eks square" instead of "eks to the power of begin exponent two end exponent". To implement this heuristic, just a simple check on the number of digits on the exponent will not work. In fact, tree processing has to be completely halted after the detection of "squares", "cubic", etc., beacuse an out-of-order in the parsing of the equation occurs in this conversion.

Derivatives such as $\frac{df(x)}{dx}$ must be identified as such and clearly separated from fractions. Otherwise, it would be converted as "fraction begin numerator derivative of $f(x)$ end numerator, begin denominator derivative of x, end denominator" instead of "derivative of $f(x)$ in order to x".

The implementation of this simple heuristic was the most problematic one, since, contrary to the other operators that have heuristics, this one is represented as a leaf and not a node in MathML. In this case, the *fraction* operator precedes the *derivative* operator and the fractions heuristic is already under way. To avoid this situation, before the fraction heuristic starts, a verification is done to check the existence of derivatives. If this test is positive, the the derivative heuristic is activated prior to the fraction heuristic.

Matrices can be small or very large, full of content, sparse, etc. So, creating a heuristic to process all possibilities would result in an disproportionate effort. Informal experiences with people reading matrices showed that people always say the size of the matrix in the first place. After that, no common reading methodology was found and, as a consequence, we decided to implement the same procedure for reading matrices. The heuristic starts to evaluate the size of the matrix and writes it in the beginning of the conversion text. After that, all the lines of the matrices are read one after the other, after being numbered. Although this procedure generates lots of text for small matrices, readability and comprehension are kept.

3.2 Translation Examples

Although the implemented translation system was developed for Portuguese, it is easily modified to to English. As a consequence, the following translation examples are in English, although the test results that will be presented in session 4 only comprises the Portuguese version.

$\mathbf{1 + 2 = 3}$

1 plus 2 equals to 3

$\boldsymbol{\frac{1}{3} * 3 = 1}$

1 over 3 times 3 equals to 1

$\boldsymbol{\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}}$

fraction numerator minus b plus or minus square root of b square minus 4 a c denominator 2 a.

$$\begin{bmatrix} a^2 - 2 & 4 & b^3 - 1 \\ b + 8 & 0 & ab^2 \\ 1 & -2b & -ab \end{bmatrix}$$

Matrix size 3 by 3: row 1, a square minus 2, 4, b cube minus 1, row 2, b plus 8, 0, a b square, row 3, 1, minus 2b, minus ab.

4 Experimental results

Two sets of experiments were undertaken, since the initial informal tests showed that ambiguity could arise due to the implemented heuristics. Since it is our intent to include technical books in DTBs, any ambiguity issues can be overcome by a mere visual inspection of the equations or the formulas. However, visually impaired users may not have the capability to visualize them. So, we decided to check the translation system without any visual support of the original equation. Furthermore, DTB users can have different technical backgrounds, ranging from elementary mathematics to more advanced calculus so, the translation tests comprised an "easy test set" (ETS) and a "Difficult Test Set", (DTS). Therefore, the former test set was given to 11 persons while the latter was solved by 15 persons, according to their skills. The ETS had 11 questions ranging from simple formulas $A + B = C$ to more elaborate ones, like $x = \frac{\sqrt{b^2-4\times a\times c}}{2\times a}$. Examinees were only given the output of the translator, without any clue of what the original equation could be and they had to write it down. No blank answers were allowed, so they had to opt for an answer even if they were unsure of its correctness. The same procedure was followed for the DTS but, in this case, a total of 13 questions were given. In this test, the questions were much more elaborate to encompass matrices, partial derivatives, summations, integrals, etc. For example, the Fourier transform was included, $(X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt)$. This equation was translated (in Portuguese) as *xis abre parêntesis éfe fecha parêntesis igual a integral de menos infinito até infinito, xis abre parêntesis tê fecha parêntesis exponencial de menos dois pi éfe tê derivada tê.* This translation presents two problems, the ambiguity of the power's exponent and the *derivada* that should have been *diferencial* or even *dê* as is usually read. To summarize the results, 316 of questions were collected and we got a total number of 38 erroneous. Although it would be tempting to say that a comprehensibility[2] of 88% was achieved, one should bear in mind that it would be very easy to design a test with 100% of right answers. However, we knew beforehand that some of the implemented heuristics would rise ambiguity issues, namely in fractions / divisions or even in exponents. From the answers, two different situations are evident: if the examined can identify from the text some well-known formula, he/she writes down the correct answer, no matter the ambiguity! By the opposite, if the text is not recognized, answers are not correct. In fact, one of the formulas was the volume of the sphere, $V = \frac{4}{3}\pi r^3$ that was correctly transcribed and the other was the kinetic energy, $E = \frac{1}{2}mc^2$ that most people wrongly transcribed. Both have the same translation in terms of their inner structure and, therefore, should lead to the same erroneous answers, for example, $E = \frac{1}{2mc^2}$. This problem is directly related with the heuristic used for translating fractions. To make the translation as natural as possible, we opted for avoiding the usual conversion to *Vê igual a fracção numerador 4 denominador 3 fim de fracção vezes pi vezes érre potência expoente três* for *Vê igual quatro sobre três pi érre ao cubo*. Informal discussions with the users that answered wrongly to this ques-

[2] herein defined as the ratio of correct answers versus total number of answers.

tions confirmed that they did not recognize the kynetic energy formula but did remember the volume of the sphere. They complained that there was ambiguity in the text, as expected. However, when asked to read that equation, surprisingly they said precisely what was written. Similar problems were detected with powers/exponents because we opted for suppressing statements like *begin of ... end of ...*. These declarations are quite usefull to impose sctrict boundaries to the scope of some operands but they contribute to a difficult and tiresome reading.

5 Conclusions

This paper reports the implemented translation system for mathematical equations or formulas to be included in DTBs. The aim of this system is to produce an output text that any reader can use as a tool for studying mathematics or physics. So, the most important requirement was the naturalness of the result, even if the translation could present some problems due to ambiguity. Our expectation was that this ambiguity issue could be lessened by visual inspection of the equation. Therefore, two sets of experiments involving users with different technical backgrounds were done in order to assert this issues. In both cases, users had to write down the formulas from the given text. One test comprised only simple formulas while the other contained more elaborate ones, such as integrals, etc..

Overall, test results were considered very good although some errors were reported. These errors were identified as a result of ambiguity in the output text as previously expected. In either case, results showed that although ambiguity was present in some situations, if the reader could identify the equations content, he/she could immediately overcame that problem and produce the correct answers. Some of the heuristics, namely the derivatives and matrices heuristics, need improvements to cope with a broader set of formulas.

References

1.
2. Duarte, C. and Carriço, L.: Users and Usage Driven Adaptation of Digital Talking Books. Proc. 11th International Conference on Human-Computer Interaction (HCII 2005), Las Vegas, Nevada, jul 2005.
3. Mohri, M., Riley, M., Hindle,D., Ljolje, A. and Pereira, F.: Full Expansion of Context-Dependent Networks in Large Vocabulary Speech Recognition. In Proc. ICASSP 98, Seattle, Washington, 1998.
4. http://www.latex-project.org/
5. http://www.microsoft.com/interop/docs/OfficeBinaryFormats.mspx
6. W3C, MathML Standart, http://www.w3.org/TR/2003/REC-MathML2-20031021
7. David Megginson, "Imperfect XML: Rants, Raves, Tips, and Tricks ... from an Insider". Addison Wesley Professional, Dec 8, 2004.