Rule-based Intelligent System for Dictating Mathematical Notation in Polish

Agnieszka Bier, Zdzisław Sroczyński*

* zdzislaw.sroczynski@polsl.pl
Silesian University of Technology
Faculty of Applied Mathematics

13th International Conference
on Advances in Computer-Human Interactions
ACHI 2020

Authors

- ## Agnieszka Bier

  is an assistant professor at the Faculty of Applied Mathematics of Silesian University of Technology, Gliwice, Poland. She received Ph.D. degree in mathematics from University of Silesia, Katowice, Poland in 2010 and MSc Eng in computer science from Silesian University of Technology in 2008. Her research interests include group theory, cryptography and computer assisted learning.

- ## Zdzisław Sroczyński

  is a graduate at the Faculty of Automatic Control, Electronics and Computer Science. He received Ph.D. degree in computer science in 2011 from Silesian University of Technology, Gliwice, Poland. Currently an assistant professor at the Faculty of Applied Mathematics of Silesian University of Technology. His scientific interest include image and natural language processing, human-computer interaction, multimedia systems, assistive technologies, e-learning, programming of mobile devices.

Outlook

- Motivation
- Math description languages
- Dictation of mathematical content and transcription
- Parsing to math description languages
- Experimental results and discussion

Motivation

- Editing mathematical content requires specialized graphical tools or math coding languages and hence is difficult or impossible for people with visual impairment
- As alternative, we propose a system:
  - capturing dictated mathematical content
  - parsing the content to one of the math description languages
- Potential areas of usage: education and science (in Polish)

Motivation

Alternative methods of presentation and editing mathematical content:

- automatic recognition of printed expressions
- recognition of handwritten equations
- editing with the visual editors (WYSIWYG)
- editing with the use of document description languages (LaTeX, MathML)
- editing with the use of specialized notations and software (for example Braille dot language)
- verbalization (translation to/from the spoken, natural language version, not available in Polish)

Math description languages

There are two main description languages, used to encode two-dimensional mathematical notation into linear form:

- LaTeX – de-facto standard for scientific and educational documents
- MathML, compatible with XML specification, used in Internet applications.

In our framework and "Equation wizard" editor, we use dedicated internal language, the so-called EQED format, which helps to preserve context of the equation parts.

Math verbalization

Problems:

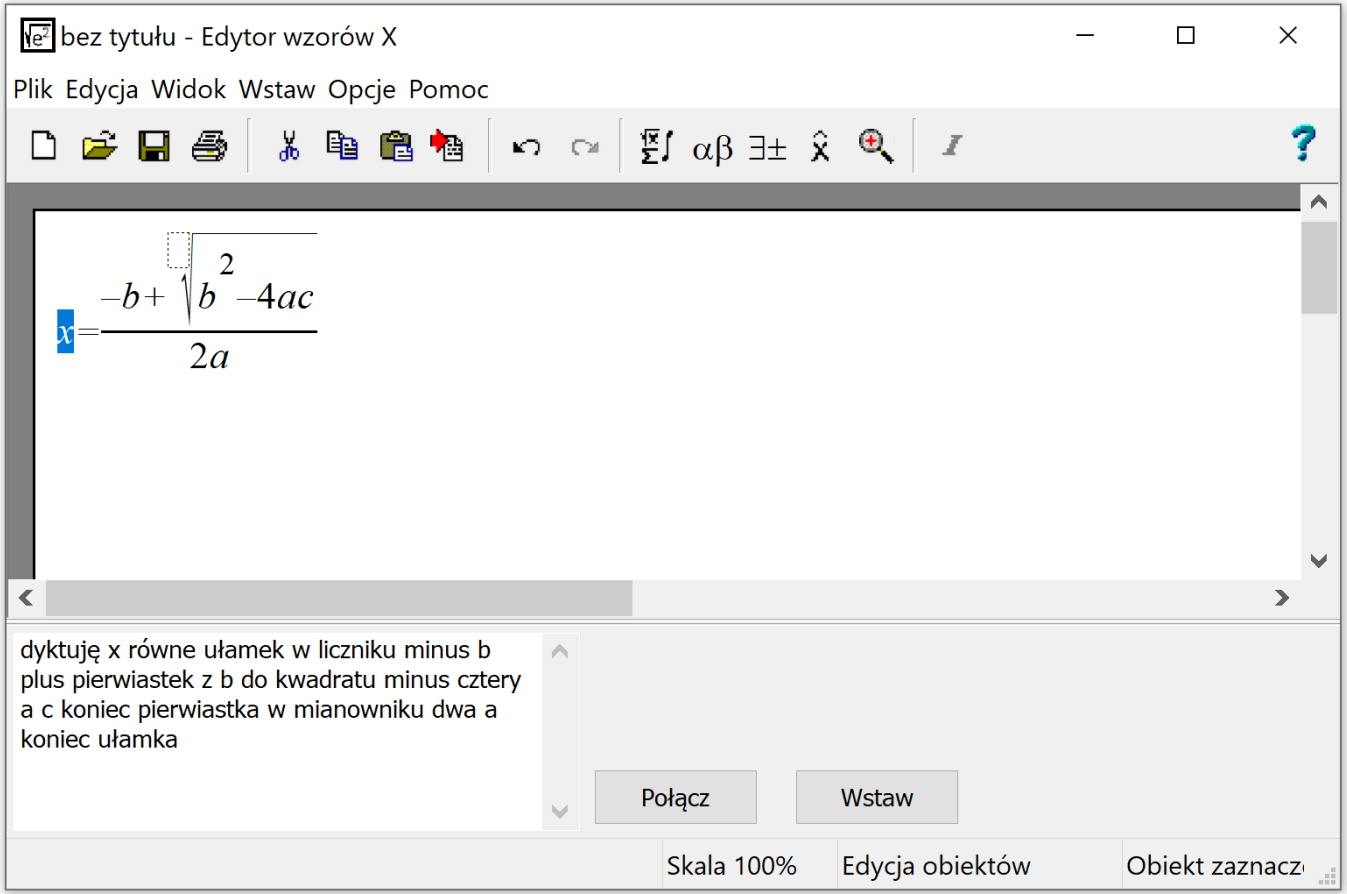
- no formal rules for the verbalization of the mathematical content
- spelling mathematical expressions is usually supported with visual presentation of the formulae
- for visually impaired people math verbalization should have very precise syntax
- terminating signs for distinguishing subexpressions are needed

Dictation of mathematical content and transcription

Dictation engine:

- encapsulated in the "Equation wizard" editor (a classic desktop app for MS Windows) for testing
- natural language description of the mathematical notation appears in the multi-line text widget
- supplementary multi-platform application for mobile operating systems (iOS and Android) for Polish language voice recognition
- includes parser designed for automatic recognition of the structure of the mathematical expression given in spoken form

Dictation of mathematical content and transcription



"Equation wizard" editor: the user interface of dictation module.

Parsing verbal form of mathematical expressions

Our editor uses tree-like data structure to represent the expression:

- the leaves of the tree contain specialized objects representing particular parts of the equation (simple letters or numbers) or two-dimensional templates
- recursive traversal through entire tree gives the possibility to translate, edit, move or alter parts of the expression

Parsing verbal form of mathematical expressions

The parser environment processes the input from the natural language into the graphical, visual representation in the following steps:

- the acquisition of the input description string in the natural language
- transfer of the input description string into the recognition module
- the actual translation with the set of predefined replacement rules
- backward scanning for the corrections of the keywords without explicit termination mark
- the transfer of the result string encoded with EQED commands for visualization
- the final visual appearance of the dictated equation is returned

Parsing verbal form of mathematical expressions

The translation rules have the following format:

`natural_language_phrase~EQED_internal_format,`

where ~ is a separator. Left side is the part to search for, right side is the corresponding notation in EQED format.

```
pierwiastek stopnia~\EQEDroot{
pierwiastek z~\EQEDroot{\EQEDplain{}}{
pod pierwiastkiem~}{
koniec pierwiastka~}
do kwadratu~#}{\EQEDnplain{2}}
do potegi~#}{\EQEDnplain{
otworz nawias~$EQEDbrackets{\left({
zamknij nawias~}{\right)@
```

Example rules for translation Polish verbalized mathematical notation into structural form.

Parsing verbal form of mathematical expressions

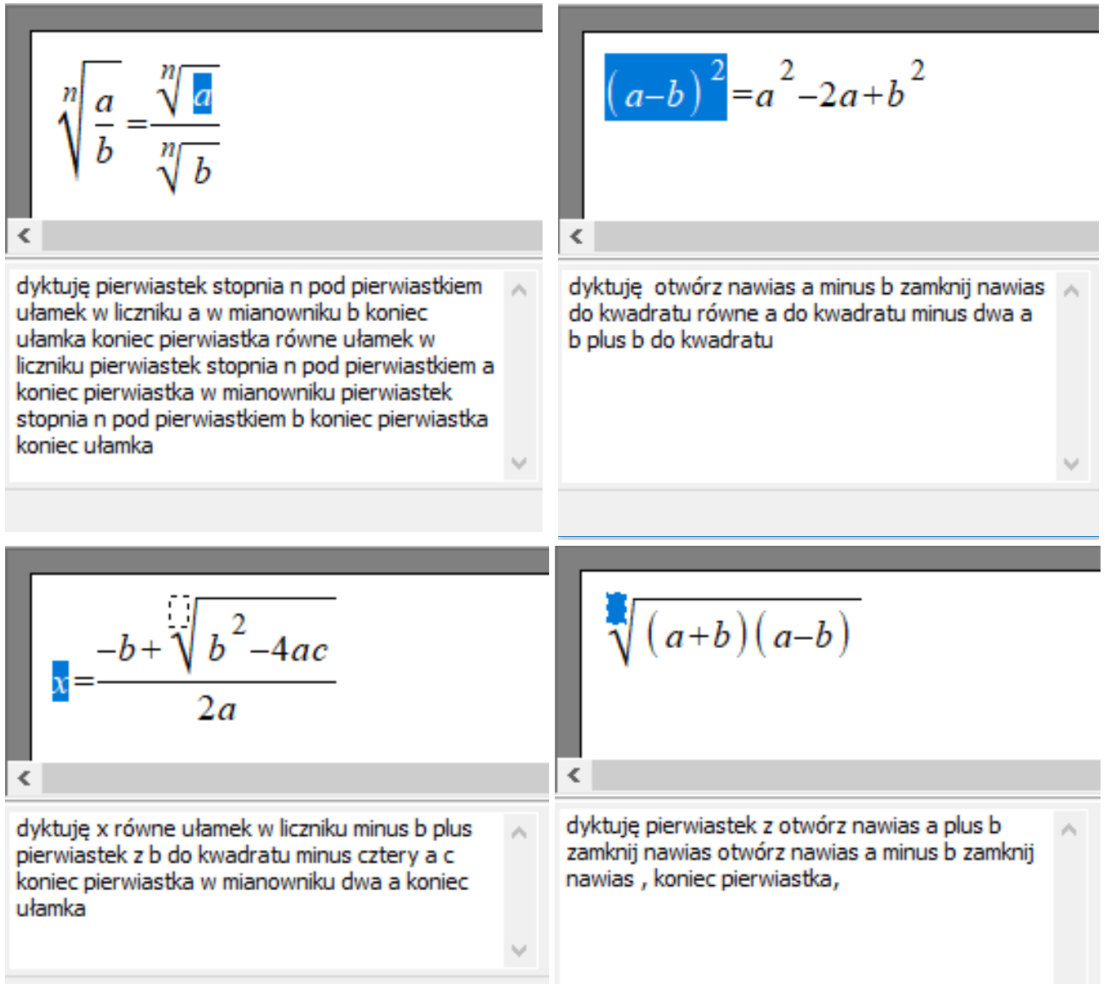
- Symbols #, @ and $ were introduced to maintain subexpressions which do not have explicit termination command or require pairing of possibly nested symbols as brackets. These symbols are processed at the final stage of parsing.
- The current version of the parser supports different variants of the spoken notation, making the dictation as natural as possible.
- There is the requirement to provide the termination command for the majority of subexpressions.

Experimental results

We have performed a series of experiments involving four experts, with the use of different operating systems for assistant mobile application (Android and iOS).

- The experts were familiar with the "Equation wizard" editor user interface and editing rules
- The experts knew the rules of verbalization of the mathematical notation and could check it live before dictation experiments
- Multiple dictation attempts of different equations were performed by every expert

Experimental results



Examples of successfully dictated equations with the visualization in "Formula wizard" editor.

Experimental results and conclusions

- The usage of the existing engine and graphical equation editor increased the effectiveness of the translation
- The system works smoothly for the current, limited list of translation rules
- Translation rules can be customized according to specific features of speech recognition modules from the mobile operating systems

Thank you!