Analysis of Braille translation software

4 authors, including:

Rostyslav Zatserkovnyi
Ukrainian State Academy of Printing Technologies
2 PUBLICATIONS **0** CITATIONS

Roksolyana Zatserkovna
Ukrainian State Academy of Printing Technologies
10 PUBLICATIONS **0** CITATIONS

ANALYSIS OF BRAILLE TRANSLATION SOFTWARE

R. H. Zatserkovnyi, V. Z. Mayik, R. S. Zatserkovna, L. Ya. Mayik

Ukrainian Academy of Printing,
19, Pid Holoskom St., Lviv, 79020, Ukraine

Assistive computer programs are one of the most invaluable technologies for people who are blind or visually impaired – this category of inclusive software can be effectively utilized for both educational purposes and everyday use, enabling its users to work with standard windowed graphical user interfaces. In this article, research publications related to software designed to assist people with visual impairments are reviewed and analyzed. Particular note is given to software that converts plaintext or rich text documents into Braille files that can be used by tactile printers or displays; otherwise known as 'Braille translators'. Additionally, the feature sets both commercial and non-commercial or open-source Braille translators are analyzed with an emphasis on their support of a Ukrainian Braille alphabet, ability to modify translation tables or internal translation logic and other functional or non-functional requirements.

Keywords: *Braille, visual impairments, Duxbury, Liblouis, Brailleblaster.*

Background of the study. One of the priority areas of inclusivity – the process of including people with disabilities into taking active part in society – is technological innovation for improving the quality of life of people with disabilities. An important part of this field is the creation of software to assist people with visual impairments: both full and partial blindness as well as other visual disturbances. This software can be divided into several different categories:

– Screen readers – software that allows people with visual impairments to effectively use computers with graphical user interfaces. This is accomplished by using text synthesizers or other non-visual means to convey textual information which would normally be displayed on a computer screen. Screen readers are capable of reading individual words, graphics, or entire documents – in the latter case the user is also informed about formatting changes. Examples of screen readers include JAWS (Job Access With Speech) and NVDA (Non-Visual Desktop Access).

– Screen magnifiers – programs that enlarge all (or some) of the content on a computer's graphical output so that people with visual impairments can better see words or graphics. Unlike screen readers, this software is generally used by people who still retain some degree of functional vision. Built-in screen magnifiers can be found in most operating systems – these include Microsoft Windows, OS X, and the GNOME or KDE graphical environments for Linux.

– Braille translators – software that converts, or 'translates', regular text documents into Braille files while preserving the original document's formatting and more

complex elements such as tables, figures or mathematical formulas. Generally, the output of this software is intended to be used by a Braille printer or Braille display (also known as an 'electronic notetaker'). Some examples of software belonging to this category are Duxbury Braille Translator, Brailleblaster, RTFC Braille-Converter, and the open-source Liblouis library.

– Optical Braille recognition software – software that converts pictures of printed Braille text to plaintext documents according to user-defined settings. These programs are useful for the needs of people who are unable to read Braille, and for the purpose of preserving or archiving printed Braille literature.

This paper analyzes Braille translators, software that converts plaintext documents into Braille files. Braille translators both support a wide variety of input text formats – plaintext, Microsoft Word, PDF, HTML, XML, etc. – and allow the user to work with many models of Braille printers or electronic notetakers. They are used for printing both fictional books and nonfiction or textbooks; in particular, Braille translators that support mathematical formulas in formats like LaTeX or MathML can be used by people with visual impairments to study mathematics, physics, computer science or other engineering disciplines.

Analysis of recent research and publications. Overall, research concerning the use of Braille in inclusive education is quite prominent. Some of the recent studies in the field concern the development of standards to formally digitize complex elements in plaintext documents, such as mathematical formulas [1] or physical diagrams [2]. Several inclusive programs are designed to target mobile devices – these include the Android Braille application for learning the tactile writing system [3] [4], an authentication method for people with visual impairments [5], or an app for reading Braille text using a mobile phone's camera [6]. Other studies address the enhancement of optical Braille recognition by using neural networks to recognize Braille characters in an image [7].

On the other hand, the analysis and development of Braille translators has received less attention in recent studies. Papers regarding this category of inclusive software algorithms include methods for translating Braille script into plaintext [8] and vice versa [9], as well as implementation details of specific Braille translators. However, the main drawback of many algorithms is that they target a single language's Braille alphabet (usually English) and often cannot be successfully used to translate Ukrainian text.

Goal. The main goal of the paper is to analyze software for converting text documents into Braille which supports a Ukrainian braille alphabet, and compare the capabilities of different Braille translators.

Main research material. Braille is a tactile writing system designed for people with visual impairments. It was designed by Louis Braille in the early 19th century as a modernization of an existing "night alphabet" system that depicted 36 all sounds of the French language with sets of 12 dots [10].

Braille is based on a combination of six raised dots arranged in rectangular blocks called cells (each containing two columns of three dots). The number and arrangement of dots in each cell encodes information about the letter, digit, symbol or word represented by that cell.

In the traditional 6-point system, there are only 64 (2^6) possible combinations of characters that can be encoded in a single cell, including a blank cell without convex dots, and therefore the same Braille character can be mapped to multiple values depending on the context. For example, in Cyrillic alphabets, the number 6 and the Latin letter f are represented by the same symbol. This problem was partially solved by an eight-point system that uses eight dots in two columns and four rows and is capable of definitively encoding all ASCII characters (0 to 255). An 8-dot system is also used by the Unicode standard.

The mappings of plaintext characters to Braille cells, which are also known as 'braille alphabets', differ from one language to another, and a single language can often use several different standards. In some languages, Braille alphabets are also divided into several grades depending on their complexity:

- A Grade 1 alphabet maps cells to letters, digits and punctuation marks, with one cell always corresponding to a single character. While the alphabet is simple to learn due to its lack of ambiguity, its main disadvantage is that books printed in it tend to be cumbersome compared to their plaintext equivalents, which is why it is used predominantly by Braille learners. Many children with visual impairments move on to more complex levels as early as elementary school [11].
- A Grade 2 alphabet maps Braille cells to both single characters and contractions used to save space. Similar to shorthand writing methods that were used by journalists or secretaries in the 19th century, these contractions can represent syllables, morphemes, or even whole words.
- A Grade 3 alphabet is a much less verbose type of shorthand that maps entire words or phrases to a few characters and reduces spacing between words or sentences to compress the final document. Generally, Grade 3 alphabets are not formalized – this category is used to refer to contractions used by people with visual impairments, which are heavily personalized and rarely used in published works.

Words and abreviations

a	but	can	do	every	from	go	have	just	knowledge	like	more	not
people	quite	rather	so	that	us	very	will	it	you	as	and	for
of	the	with	child/ch	gh	shall/sh	this/th	which/wh	ed	er	out/ou	ow	bb
cc	dd	en	gg; were	in	st	ing	ar					

Fig. 1. Examples of contractions in Grade 2 English Braille

Most Braille translators use a variation of a translation table-based algorithm [8]. It works by traversing a plaintext document from the start to the end and replacing one or

more plaintext character with the longest Braille contraction that can properly represent them (thus taking into account Grade 2 or Grade 3 contractions). Text-to-Braille mappings are contained in a translation table, which can also specify more complex or context-specific rules. Each language uses its own set of translation tables for Braille alphabets of different grades.

The main advantage of this approach is that a single Braille translation algorithm can work with hundreds of languages – internationalization is achieved by using different translation tables. However, the output of a translator based on this algorithm requires manual verification, as many Grade 2 or Grade 3 contractions are context-specific. For instance, in a Grade 2 alphabet, the letters 'wh' can be replaced with a single shorthand cell, but only if they belong to a single word – the contraction will therefore be used when translating '**wh**ile', but not 'ra**w h**ide'. While this rule is relatively simple, others are far more complex and may not be properly represented by a translation table. Some rules like these have even been removed from Braille standards in order to simplify Braille translation algorithms [12].

It should be noted that standardized shorthand-like abbreviations needed to design a Grade 2 alphabet have not yet been developed for the Ukrainian alphabet. Moreover, there are no formal standards for Ukrainian Braille – even basis aspects such as character mappings, punctuation or line spacing specifics. Ukrainian translation tables are often copied from different Soviet or Russian standards with non-standardized custom tweaks. Therefore, it is important for Ukrainian end users of Braille translators to tailor translation tables used by the software to their own needs. Often, Ukrainian Braille tables are borrowed from Russian standards with some changes. Therefore, one of the most important aspects of Braille translators in the context of the Ukrainian end user is the ability to tailor the conversion tables of the program to their own needs.

One of the most popular Braille translators is Duxbury Braille Translator – a program for Microsoft Windows and OS X operating systems, the first version of which was released by Duxbury Systems Inc. in 1995. The Duxbury Braille Translator (hereinafter referred to as DBT) includes tools that can automate the process of converting plain text to Braille script (and vice versa) and a visual editor for working directly with Braille documents.

DBT supports over 170 languages, including Ukrainian, and can translate documents to both Grade 1 and Grade 2 alphabets in languages where these alphabets are standardized. (Using the translator's own terminology, these alphabets are referred to as 'uncontracted' and 'contracted' respectively.) The Ukrainian alphabet in DBT is 'uncontracted' – a single character almost always corresponds to a single braille cell, although more than one cell is used to display certain punctuation, capital letters and digits. However, the translation table does not contain any abbreviations or abbreviations. This particular alphabet is based a Soviet standard for the Ukrainian alphabet, developed in 1990 [13].

DBT works with two internal file formats, .dxp and .dxb, which contain plain text and Braille script respectively. One of these formats can be converted to the other using the software's translation functionality. In addition to its own formats, DBT also accepts files in .brf (Braille Ready Format). Each page of a BRF file contains up to 25 lines with

39 braille characters – the format supports only braille characters, spaces and indentation. E-books adapted for the visually impaired are usually distributed in this format.

The translator can also import a number of files created by word processors – these include Microsoft Office, OpenOffice, HTML, XML files as well as TeX and LaTeX formats. However, the program does not support PDF files – these need to be converted to one of the supported formats via third-party software.

In addition to plaintext, DBT can also translate mathematical formulas as long as their location in the document is manually marked by the user. The Ukrainian alphabet supports the Soviet standard for the designation of mathematical, physical, chemical and astronomical symbols [14]; DBT reuses this standard for most countries of Eastern Europe and the former USSR, as well as the Simplified Chinese alphabet.

One of the major drawbacks of DBT from a Ukrainian end user's point of view is the inability to alter translation tables – a deliberate decision made to ensure that documents translated by the software comply with each language's formal Braille standards. However, as there is no set standard for a Ukrainian Braille alphabet, DBT's translation table may not match the conventions adopted by the end user or their organization.
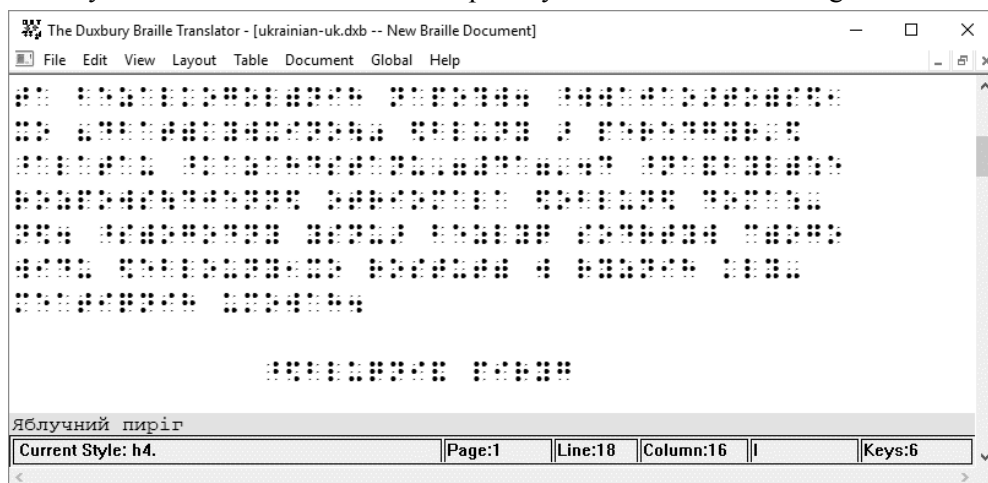

Рис. 2. The user interface of DBT

DBT does provide access to translation table files, namely:
– ukracb.btb (the file responsible for setting the Braille font)
– ukracp.btb (file containing Braille settings)

However, the format of these table files or the process of modifying them are not documented, and their contents may be altered by software updates.

One alternative to commercial Braille translators is Liblouis, an open-source Braille translation library that has been in development since 2002. Like DBT, this library supports text translation and formatting in Grade 1 and Grade 2 alphabets for several hundred languages. The Ukrainian translation table at Liblouis was developed by the 'Lemur' Citizens Association and is a copy of the (standardized) Russian alphabet with the added symbols Е, І, Ї and Ґ – the table also supports standard mathematical symbols [15].

LibLouis's intuitive translation table system allows the user to easily add new tables to a program or modify existing ones. Translation tables used by the library consist of a sequence of translation rules, which in turn contain three parts - a translation command, a sequence of text characters, and one or more Braille cells. If the translator encounters a sequence of plaintext characters in a document and the condition is specified by the translation command is met, then LibLouis converts these characters to the Braille cells specified in the rule. For example, a translation rule

always world 456-2456

will always translate the word 'world' into two Braille characters – 456 (i.e. the 4th, 5th and 6th dots in the cell will be raised) and 2456, or the letter 'w'. Other translation commands are triggered only under certain conditions - for example, the 'lowword' command is activated when and only if a character specified in the translation rule is preceded and followed by a space.

Liblouis also includes built-in tools for debugging and testing translation tables, and a user creating their translation table can run both manual and automated tests to verify that they work correctly. The easiest way to do so is to write a YAML file that serves as an automated test, specifying a series of words and phrases to be translated mapped to their expected translation. The lou_checkyaml subroutine can then verify a table's correctness based on this series of tests [16]. Other subroutines such as lou_debug, lou_trace, and lou_checktable can be used for manual testing and finetuning a translation table.

The main disadvantage of Liblouis is its lack of support for formatted documents (the library only supports plaintext files) and its non-visual user interface. Each of the library's parameterized subroutines – lou_tranlateString, lou_backTranslate, lou_compileString, lou_hyphenate etc. – corresponds to a function in the library's C-based backend, and must be called as a console application.

Since Liblouis is an open-source library, several full-fledged Braille translators with a visual interface have been developed based on its core capabilities. One example of a complete Liblouis-based translator is BrailleBlaster. This program, designed by the American Printing House for the Blind, is mainly intended for printing Braille textbooks to English standards, but is open to the public. (Unlike Liblouis itself, the software is not open source.) It supports both Microsoft Windows and OS X and Linux operating systems, and includes drivers for many models of Braille printers. In addition to plain text, BrailleBlaster supports the translation of tables and mathematical formulas, and allows the user to replace images with text descriptions, while maintaining the original formatting of the translated document. The program is adapted for users with visual impairments and works well with screen readers and magnifiers.

However, BrailleBlaster is a US-oriented program. To date, it only supports the English alphabets UEB and EBAE, as well as Spanish and Cherokee. The user cannot utilize their own translation tables to work with other languages, including Ukrainian.

Conclusions. This paper examines several Braille translators and compares their functionality. While commercial solutions can be effectively used to translate a rich text document, their translation tables' lack of flexibility can be a hindrance for Ukrainian users due to the lack of a standardized translation table. Open-source libraries such as

Liblouis can thus become the basis of a new software product designed to meet the needs of a Ukrainian user.

REFERENCES

1. Nazemi A., Fernando C., Murray I., McMeekin David. A. Access to all components of scanned mathematical documents by vision-impaired students. Assistive Technology. 2018. Vol. 30. № 2. Pp. 59–65.

2. Dickman A. G., Martins A. O., Ferreira A. C., Andrade L. M. Adapting diagrams from physics textbooks: a way to improve the autonomy of blind students. Physics Education. 2014. Vol. 49. № 5. Pp. 526–531.

3. Nahar L., Jaafar A., Ahamed E., Kaish A. B M A. Design of a Braille Learning Application for Visually Impaired Students in Bangladesh. Assistive Technology. 2015. Vol. 27. № 3. Pp. 172–182.

4. Nahar L., Sulaiman R., Jaafar A. «Bangla Braille learning application» in smartphones for visually impaired students in Bangladesh. Interactive Learning Environments. 2019. Pp. 1–14.

5. Alnfiai M., Sampalli S. BraillePassword: accessible web authentication technique on touchscreen devices. Journal of Ambient Intelligence and Humanized Computing. 2019. Vol. 10. № 6. Pp. 2375–2391.

6. Shreekanth T., Udayashankara V. A Review on Software Algorithms for Optical Recognition of Embossed Braille Characters. International Journal of Computer Applications. 2013. Vol. 81. № 3. Pp. 25–35.

7. Hassan M., Ahmed G. M. Conversion of English Characters into Braille Using Neural Network. Iraqi Journal of Computer, Communication, Control and Systems Engineering. 2011. Vol. 11. Pp. 28–35.

8. Blenkhorn P. A system for converting braille into print. IEEE Transactions on Rehabilitation Engineering. 1995. Vol. 3. № 2. Pp. 215–221.

9. Blenkhorn P. A system for converting print into braille. IEEE Transactions on Rehabilitation Engineering. 1997. Vol. 5. № 2. Pp. 121–129.

10. Jiménez J., Olea J., Torres J., Alonso I., Harder D., Fischer K. Biography of Louis Braille and Invention of the Braille Alphabet. Survey of Ophthalmology. 2009. Vol. 54. № 1. Pp. 142–149.

11. DePountis V. M., Pogrund R. L., Griffin-Shirley N., Lan W. Y. Technologies Used in the Study of Advanced Mathematics by Students who Are Visually Impaired in Classrooms: Teachers' Perspectives. Journal of Visual Impairment & Blindness. 2015. Vol. 109. № 4. Pp. 265–278.

12. Sullivan J. Braille Becomes Electric: The Trials and Triumphs of Braille Translation Software. Journal of Visual Impairment & Blindness. 2009. Vol. 103. № 7. Pp. 389–391.

13. World Braille Usage, Third Edition. National Library Service for the Blind and Physically Handicapped, Library of Congress. 2013, 210 p.

14. Система обозначений по математике, физике, химии и астрономии : учеб. пособ. / состав.: Быков А. Г., Егоров М. И., Голубчиков А. Ф., Морозова Г. Б., Проскуряков И. В. ; под общ. ред. А. Г. Быкова. 2-е изд., перераб. и доп. Москва : ВОС, 1982.

15. Liblouis. URL: https://github.com/liblouis/liblouis/blob/master/tables/uk.utb.

16. Liblouis – An open-source translator and back-translator. URL: http://liblouis.org/documentation/liblouis.html.

REFERENCES

1. Nazemi, A., Fernando, C., Murray, I., & McMeekin, David. (2018). A. Access to all components of scanned mathematical documents by vision-impaired students: Assistive Technology, 30, 2, 59–65 (in English).

2. Dickman, A. G., Martins, A. O., Ferreira, A. C., & Andrade, L. M. (2014). Adapting diagrams from physics textbooks: a way to improve the autonomy of blind students: Physics Education, 49, 5, 526–531 (in English).

3. Nahar, L., Jaafar, A., Ahamed, E., & Kaish, A. B M A. (2015). Design of a Braille Learning Application for Visually Impaired Students in Bangladesh: Assistive Technology, 27, 3, 172–182 (in English).

4. Nahar, L., Sulaiman, R., & Jaafar, A. (2019). «Bangla Braille learning application» in smartphones for visually impaired students in Bangladesh: Interactive Learning Environments, 1–14 (in English).

5. Alnfiai, M., & Sampalli, S. (2019). BraillePassword: accessible web authentication technique on touchscreen devices: Journal of Ambient Intelligence and Humanized Computing, 10, 6, 2375–2391 (in English).

6. Shreekanth, T., & Udayashankara, V. (2013). A Review on Software Algorithms for Optical Recognition of Embossed Braille Characters: International Journal of Computer Applications, 81, 3, 25–35 (in English).

7. Hassan, M., & Ahmed, G. M. (2011). Conversion of English Characters into Braille Using Neural Network: Iraqi Journal of Computer, Communication, Control and Systems Engineering, 11, 28–35 (in English).

8. Blenkhorn, P. (1995). A system for converting braille into print: IEEE Transactions on Rehabilitation Engineering, 3, 2, 215–221 (in English).

9. Blenkhorn, P. (1997). A system for converting print into braille: IEEE Transactions on Rehabilitation Engineering, 5, 2, 121–129 (in English).

10. Jiménez, J., Olea, J., Torres, J., Alonso, I., Harder, D., & Fischer, K. (2009). Biography of Louis Braille and Invention of the Braille Alphabet: Survey of Ophthalmology, 54, 1, 142–149 (in English).

11. DePountis, V. M., Pogrund, R. L., Griffin-Shirley, N., & Lan, W. Y. (2015). Technologies Used in the Study of Advanced Mathematics by Students who Are Visually Impaired in Classrooms: Teachers' Perspectives: Journal of Visual Impairment & Blindness, 109, 4, 265–278 (in English).

12. Sullivan, J. (2009). Braille Becomes Electric: The Trials and Triumphs of Braille Translation Software: Journal of Visual Impairment & Blindness, 103, 7, 389–391 (in English).

13. World Braille Usage, Third Edition. (2013). National Library Service for the Blind and Physically Handicapped, Library of Congress (in English).

14. Bykov, A. G., Egorov, M. I., Golubchikov, A. F., Morozova, G. B., & Proskuriakov, I. V. (1982). Sistema oboznachenii po matematike, fizike, khimii i astronomii. 2-e izd., pererab. i dop. Moskva : VOS (in Russian).

15. Liblouis. Retrieved from https://github.com/liblouis/liblouis/blob/master/tables/uk.utb (in English).

16. Liblouis – An open-source translator and back-translator. Retrieved from http://liblouis.org/documentation/liblouis.html (in English).

АНАЛІЗ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ПЕРЕВЕДЕННЯ ДОКУМЕНТІВ У ШРИФТ БРАЙЛЯ

Р. Г. Зацерковний, В. З. Маїк, Р. С. Зацерковна, Л. Я. Маїк

Українська академія друкарства,
вул. Під Голоском, 19, Львів, 79020, Україна
vol_maik@meta.ua

Асистивне програмне забезпечення відноситься до однієї з найцінніших сучасних технологій для людей, які є сліпими або слабозорими. Програми, що належать до цієї категорії інклюзивного програмного забезпечення, можуть бути ефективно використані як в навчальних цілях, так і для повсякденного використання: таке програмне забезпечення дозволяє користувачам з вадами зору ефективно працювати зі звичними віконними графічними інтерфейсами, навіть якщо такі інтерфейси не адаптовані для користувачів з особливими потребами. У цій статті розглядаються та аналізуються дослідницькі публікації, що стосуються асистивного програмного забезпечення, цільовою аудиторією якого є люди з вадами зору; в результаті аналізу виділяються пріоритетні напрямки досліджень в галузі асистивного програмного забезпечення. В процесі аналізу особлива увага приділяється програмному забезпеченню, що здатне перетворювати текстові документи – як прості текстові файли, так і повноцінні розмічені текстові документи формату .doc, .rtf, .odt і т. ін. – у спеціальні файли з текстом у форматі шрифта Брайля для подальшої обробки таких документів брайлівськими принтерами та тактильними дисплеями; при цьому зберігаючи початкову розмітку та форматування документів, а також видаляючи або адаптуючи зображення, таблиці, формули та інші складні елементи. Такі програми також відомі як "Брайлівські перекладачі". У статті аналізуються функціональні перекладачі як комерційних, так і некомерційних або відкритих Брайлівських перекладачів, в тому числі Duxbury Braille Translator, Liblouis і BrailleBlaster. Особлива увага у аналізі звертається на можливість використання українського алфавіту Брайля, здатність власноруч редагувати брайлівські таблиці інтернаціоналізації, задавати власні правила роботи алгоритму перекладу або змінювати інші користувацькі налаштування з врахуванням потреб конкретного кінцевого користувача.

Ключові слова: *шрифт Брайля, люди з вадами зору, Duxbury, Liblouis, Brailleblaster.*