AUTOMATIC TRANSLATION INTO FRENCH GRADE 2 BRAILLE

by

Monique Truquet

Abstract

I would like to speak on the following subjects. How to translate texts into Braille with only a typist who doesn't know Braille rules.

Difficulties encountered: codes of letters with accents and Braille signs. They have been studied when we worked for Braille Grade I.

To obtain Braille Grade 2 we use a syntactical analyser, a "hashing" table and a "contraction" program.

It seems that this last program can be used for other languages and it is this one that we shall study.

Introduction and Summary

I shall first present the difficulties I have encountered before and during translation into Braille Grade I. Then I shall present the automatic translation into Braille Grade 2.

I should like to speak about Mr. Espitallier (teacher for the blind) who died last year and gave me some helpful hints on Braille.

At this moment I am being helped with the Braille by Mrs. Fresquet (teacher for the blind) and I should like to thank her.

I should like to say that I have been working on the automatic translation into Braille since December 1966 and that Mr. Lorho (IRIA engineer) gave me some hints for the choice of a syntactic analyser.

French Braille Grade I

At first we have carried out an automatic translation from inkprint texts to Braille Grade I. We considered that this first step was necessary to find solutions to the following problems:

- how to introduce letters with accents or Braille signs
- how to build tables
- how to make up a Braille page correctly

with all the special signs (capitals, numerical signs, italic signs . . .), or dividing the words when necessary or suppressing a space when possible: before punctuation or when there are two spaces instead of one.

We have encountered some difficulties while we were punching data: We are working on an IBM 7044; its cards are punched with an IBM 026. The keyboard of this IBM does not have enough characters, so we coded those which do not exist by a letter followed by a digit.

Accentuated letters are represented by a vowel followed:

by 1 if it is a grave accent (ˋ)
by 2 if it is an acute (ˊ)
by 3 if it is a circumflex (^)
by 4 if it is a diaeresis (¨)

For punctuation and Braille signs the same code is used, so that a semi-colon is coded P1. We have put a "1" because a semi-colon is used more than a question mark P2, or an exclamation mark P3.

A capital is coded M1.

We now present all the codes used:

A1 for à
A3 for â
A4 for ä
C5 for ç
D1 for beginning of a title (Début de titre)
D2 for beginning of a sub-title (Début de sous-titre)
D3 for beginning of a hyphen (Début de tiret)
D4 for beginning of an italic (Début d'italique)
D5 for :
E1 for è
E2 for é
E3 for ê
E4 for ë
F1 for end of a title (Fin de titre)
F2 for end of a sub-title (Fin de sous-titre)
F3 for end of a hyphen (Fin de tiret)
F4 for end of an italic (Fin d'italique)
G1 for « opening brackets (ouverture Guillemets)
G2 for » closing brackets (fermeture Guillemets)
I3 for î
I4 for ï
M1 for capital (Majuscule)
O3 for ô
O6 for oe
L1 for new paragraph (saut de ligne)
L2 for new page (saut de page)
P1 for ;
P2 for ?
P3 for !
P4 for mark before a name or after initial
} Punctuation
T1 for hyphen when the interlocutor is changing
U1 for ù
U3 for û
U4 for ü
V1 for end of verse (Fin de vers)

French Braille Grade 2

1. How to Recognize Words or Phrases

To recognize words or phrases, we have chosen a syntactical analyser with a list structure and a "hashing" table.

We have used the grammar which follows:

TEXT:: = |TERM ΣTERM| followed by |punctuation Σpunctuation|
TERM:: = SPECIAL SIGN/PHRASE/WORD/NUMBER
WORD:: = LETTER ΣLETTER
NUMBER:: = DIGIT ΣDIGIT
SPECIAL SIGNS are in a table
PHRASES are in a table: these are expressions like: "c'est-à-dire, la plupart . . ."
LETTERS are in a table
DIGITS are in a table

We have chosen an analysis from top to bottom its axiom being TEXT.

2. How to find "PHRASES"

Supposing we have to find the phrase "à travers" or when it is coded "A1 TRAVERS". We have to survey the following "tree":

$$\begin{tikzcd}
 & & & & & & & U \arrow[r] & S \\
 & & \text{-} \arrow[r] & D \arrow[r] & E \arrow[r] & S \arrow[r] & S \arrow[r] \arrow[ur] & O \arrow[r] & U \arrow[r] & S \\
 & & b \arrow[r] & C \arrow[r] & O \arrow[r] & N \arrow[r] & T \arrow[r] & R \arrow[r] & A \arrow[r] & I \arrow[r] & R \arrow[r] & E \\
A \arrow[r] \arrow[dr] & U \arrow[r] \arrow[ur] \arrow[uur] & J \arrow[r] & O \arrow[r] & U \arrow[r] & R \arrow[r] & D \arrow[r] & \text{'} \arrow[r] & H \arrow[r] & U \arrow[r] & I \\
 & 1 \arrow[r] & b \arrow[r] \arrow[dr] \arrow[ddr] \arrow[dddddr] & C \arrow[r] & A \arrow[r] & U \arrow[r] & S \arrow[r] & E \\
 & & & M \arrow[r] & E \arrow[r] & S \arrow[r] & U \arrow[r] & R \arrow[r] & E \\
 & & & P \arrow[r] \arrow[ddr] & E \arrow[r] \arrow[dr] & I \arrow[r] & N \arrow[r] & E \\
 & & & & & U \arrow[r] & b \arrow[r] & P \arrow[r] & R \arrow[r] & E \arrow[r] & 1 \arrow[r] & S \\
 & & & & R \arrow[r] & E \arrow[r] & 2 \arrow[r] & S \arrow[r] & E \arrow[r] & N \arrow[r] & T \\
 & & & T \arrow[r] & R \arrow[r] & A \arrow[r] & V \arrow[r] & E \arrow[r] & R \arrow[r] & S
\end{tikzcd}$$

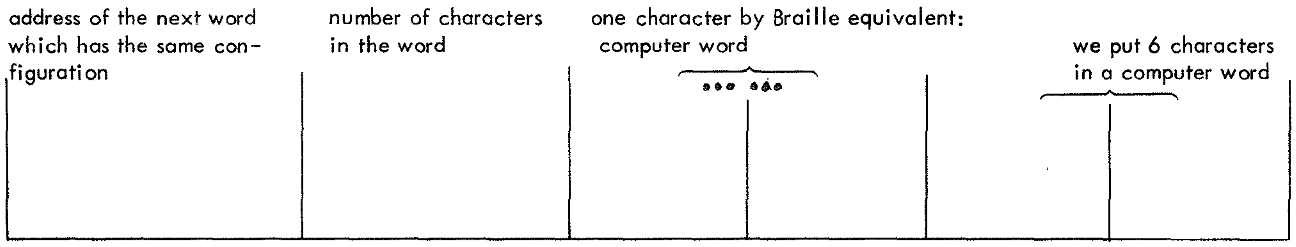
Suppose that we have recognized A1 and the space (b) and nothing else, then A1 and the space are translated and the tests continue.

3. "Hashing" Table

If the word is not a phrase we have to find it in the "hashing" table. We have put in this table all the words which do not obey the rules.

We have said that a word is formed by a sum of letters, so we do the sum of letters and we keep only the six light weight "bits". The "hashing" table is built with the help of this sum; it is a table made up of half words, half characters.

We have the following configuration:


address of the next word which has the same configuration
number of characters in the word
one character by Braille equivalent:
computer word
••• •••
we put 6 characters in a computer word

If the word is not in the "hashing" table we call upon the contraction program.

4. French Contractions - Contraction Program

4.1 Contractions

The use of "contractions" depends on many rules; some may not be used at the beginning of a word like "ER", "EUR", "ION", others like "RECOM", "REDIS" may be used at the beginning of a word, and in general, they may be used before a consonant.

As we detach words from the text, the beginning of a word is recognized by the fact that it is the first character of a word, and the end the last character (since we know the number of characters).

To recognize "contractions" we need: two counters, the number of characters in the word and the number of characters in the contractions.

In French the longest contraction has twelve characters, for example "DISPOSITIONS". We see that "DISPOSITIONS" may be a word, so we begin by testing if the whole word has twelve characters or less.

If the word has more than twelve characters, we keep the twelve characters on the right and we make tests from left to right until a group of letters or a letter is recognized. Then we put all the characters not yet translated on the right and we test again from left to right.

We have to speak about the formation of the "contraction" table. It is built with the help of the number of characters in the contraction. This information is followed by two numbers of two digits which represent the rules that we must test. If the first rule is not verified, the second rule is taken and if no rule is verified the tests continue. When the rule is verified, we output the Braille equivalent.

4.2 Examples

4.2.1 INTERPELLATION

This word has fourteen characters, therefore we keep only twelve characters on the right and we test to determine if it is a contraction. TERPELLATION is not a "contraction" so we continue to test. We find "ATION", a contraction that may be used at the end of the word, therefore it is translated.

Nine characters remain: INTERPELL

"IN" is a "contraction" and because that group of letters is before a consonant, "IN" is translated.

Seven characters remain: TERPELL

"LL" is a "contraction" and because that group of letters is between two vowels, "LL" is translated.

Five letters remain: TERPE

In this case "T" only is translated.

Four letters remain: ERPE

Now "E" on the right is translated.

Three letters remain: ERP

"ER" is a contraction and because it is before a consonant, "ER" is translated. Finally P is translated.

Thus INTERPELLATION is translated: (IN) - (T) - (ER) - (P) - (E) - (LL) - (ATION)

4.2.2 ILLOGIQUEMENT

Sometimes in a word it is possible to contract two different groupings of letters; the priority rule is governed by the fact that we may choose the longer contraction.

We take the following example:

ILLOGIQUEMENT

Firstly, we test "ILLOGIQUEMENT". Since it is not a contraction, we examine the word minus the first letter "I" that is, "LLOGIQUEMENT", then we examine "LOGIQUEMENT", which is a contraction. The Braille equivalent is put into a buffer while we examine "IL". "IL" is not a "contraction" and we examine "I", "I" is translated, lastly "L".

"ILLOGIQUEMENT" becomes: (I) - (L) - (LOGIQUEMENT)

If we tested the word from the first character from the left to right, "LL" would be first encountered, it would be difficult to find "LOGIQUEMENT"; this is the reason why we have not used this method.

5. Presentation of a Braille Grade 2 Page

We must respect all the rules, and naturally the fact that a Braille line does not have two consecutive spaces between two words or a space before punctuation. The program corrects that and also adds a space after a punctuation if it does not exist.

We try to respect the inkprint text and we do what is mentioned there.

We also respect the following rules:

- the surname must be preceeded by the Braille capital sign and the name must be translated character-by-character
- a foreign name must be preceeded by the integral Braille sign and must also be translated character-by-character
- for example, the name of a school must be preceeded by the Braille capital sign and must be translated into Braille Grade 2

CONCLUSION

We have no problem with the contraction method.

We use 170 contractions and about 2,000 words in the "hashing" table; statistics have shown that we test 25 words at the most.

We are working at present to obtain a good presentation of a Braille page. We think that we shall obtain a Braille text without error, if the text is properly punched, of course.

Our problem is to obtain raised dots perfectly, but since we have no money specifically for this research we use our own resources and hence the Braille character is not the standard size.

Our programs are written in FORTRAN and M.A.P. Later we want to rewrite M.A.P. programs into PL/I or other universal languages.

We also have to do another task: The translation into mathematical Braille. We shall use the same method but we shall have to create other codes. As we have seen, those codes permit us to drive the main program towards subprograms or to obtain the Braille equivalent. This method is interesting because it is very tractable.