A Rule-Based Grapheme-to-Phoneme Conversion System

Piotr Kłosowski

Department of Telecommunication and Teleinformatics, Faculty of Automatic Control, Electronics and Computer Science, Silesian University of Technology, Akademicka 16, 44-100 Gliwice, Poland; pklosowski@polsl.pl

Abstract: This article presents a rule-based grapheme-to-phoneme conversion method and algorithm for Polish. It should be noted that the fundamental grapheme-to-phoneme conversion rules have been developed by Maria Steffen-Batóg and presented in her set of monographs dedicated to the automatic grapheme-to-phoneme conversion of texts in Polish. The author used previously developed rules and independently developed the grapheme-to-phoneme conversion algorithm. The algorithm has been implemented as a software application called TransFon, which allows the user to convert any text in Polish orthography to corresponding strings of phonemes, in phonemic transcription. Using TransFon, a phonemic Polish language corpus was created out of an orthographic corpus. The phonemic language corpus allows statistical analysis of the Polish language, as well as the development of phoneme- and word-based language models for automatic speech recognition using statistical methods. The developed phonemic language corpus opens up further opportunities for research to improve automatic speech recognition in Polish. The development of statistical methods for speech recognition and language modelling requires access to large language corpora, including phonemic corpora. The method presented here enables the creation of such corpora.

Keywords: grapheme-to-phoneme conversion; speech recognition; language corpus; language modelling; language statistical analysis.

1. Introduction

Natural language processing often requires grapheme-to-phoneme (G2P) conversion of an orthographic text [1]. G2P converts strings of graphemes to corresponding sequences of phonetic transcription characters, directly from orthographic representations and it is crucial for many applications in various areas of speech and language processing [2]. The most frequent applications of grapheme-to-phoneme conversion occur in text-to-speech systems, which require high quality phonetic transcriptions to function well [3]. Tools for converting graphemes to phonemes are also used in theoretical and applied linguistics. Such tools are useful in many areas of linguistic research (e.g., phonetics, phonology, dialectology, and language acquisition), in order to obtain preliminary phonetic transcriptions of large language corpora [4].

The main goal of research on the conversion of graphemes to phonemes is improving speech recognition for the Polish language [5,6]. In addition, research studies have been conducted on the phonetic properties of Polish phonemes [7,8], speech recognition based on such analyses [9], speaker recognition [10–14] and new applications of speech and language processing (e.g., speech translation) [15]. Particularly good results in speech recognition are achieved through the use of statistical language models [16–19]. However, currently the field of language modelling is shifting from statistical methods to neural networks and deep learning methods [20,21]. The development of statistical and deep learning methods for speech recognition and language modelling requires access to large language corpora, including phonemic corpora [18–25]. The main motivation for undertaking this research on automatic grapheme-to-phoneme conversion and its application, was the development of effective methods of creating a phonemic language corpus for Polish, comprised of
phonemic transcriptions derived from an orthographic language corpus through grapheme-to-phoneme conversion.

2. Problem Formulation

The process of converting graphemes to phonemes in orthographic text involves converting a string of orthographic characters into a corresponding string of phonetic transcription characters (representing phonemes or allophones) [2]. A ‘grapheme’ is any of the units of any writing system for any language, a term coined by analogy with the ‘phoneme’ of a spoken language [26]. Graphemes include alphabetic letters, typographic ligatures, numerical digits, punctuation marks, and other individual symbols of writing systems. Since the orthographic text is the only source of pronunciation information in the process of converting graphemes into phonemes, this process must be based on appropriate formal rules, depicting the correct pronunciation of orthographic strings in a given language [27].

Phonemes are usually written in specially designed alphabets. The most widely used alphabet is the International Phonetic Alphabet (IPA) [28]. For the Polish language, as with other Slavic languages, a special transcription system, called the Slavistic Phonetic Alphabet (SPA), is most frequently used [29]. The other very commonly used phonetic alphabet is the Speech Assessment Methods Phonetic Alphabet (SAMPA) [30]. SAMPA is a machine-readable phonetic alphabet, using 7-bit printable ASCII characters, based on the IPA. Table 1 presents the phonemic inventory of Polish with examples, in the SPA, IPA, and SAMPA phonetic alphabets and corresponds to the set of phonemes used for the purpose of this study.

Automatic grapheme-to-phoneme conversion is not a new problem. The first linguist who noted it, and tried to provide a solution for a particular language (Czech), was H. Kučera [31]. Research on solutions to the automatic grapheme-to-phoneme conversion problem have also been initiated for other languages [32–34].

Table 1. The set of Polish phonemes with examples, written in the SPA, IPA, and SAMPA phonetic alphabets, which corresponds to the set of phonemes used for the purpose of this study.

| No. | Phonetic Alphabet Symbols | Example of Occurrence in Polish |
|-----|--------------------------|-------------------------------|
| SPA | IPA | [SAMPA] | |
| 1   | [e] | [e] | serce |
| 2   | [a] | [a] | baba |
| 3   | [o] | [o] | oko |
| 4   | [t] | [t] | trawa |
| 5   | [n] | [n] | noc |
| 6   | [y] | [i] | syty |
| 7   | [j] | [j] | jajo |
| 8   | [i] | [i] | wieci |
| 9   | [r] | [r] | rok |
| 10  | [s] | [s] | sok |
| 11  | [v] | [v] | wada |
| 12  | [p] | [p] | praca |
| 13  | [u] | [u] | bok |
| 14  | [m] | [m] | mama |
| 15  | [k] | [k] | kot |
| 16  | [n] | [n] | kosti |
| 17  | [d] | [d] | dudek |
| 18  | [l] | [l] | lato |
| 19  | [u] | [u] | lisy |
| 20  | [s] | [s] | szysza |
| 21  | [r] | [r] | fala |
| 22  | [z] | [z] | koza |
| 23  | [c] | [s] | całko |
| 24  | [b] | [b] | baba |
| 25  | [g] | [g] | godło |
Table 1. Cont.

| No. | Phonetic Alphabet Symbols | Example of Occurrence in Polish |
|-----|---------------------------|---------------------------------|
|     | [SPA] | [IPA] | [SAMPA] |
| 26  | [ś]  | [s]  | [s']   | siano |
| 27  | [ć]  | [k]  | [ks']  | ciasto |
| 28  | [ń]  | [n]  | [n']   | higiena |
| 29  | [ć]  | [k]  | [ks']  | czarno |
| 30  | [ź]  | [z]  | [z']   | klik |
| 31  | [ż]  | [ż]  | [ż']   | zarno |
| 32  | [ń]  | [n]  | [n']   | kin |
| 33  | [ń]  | [ń]  | [ń']   | dziadzic |
| 34  | [ź]  | [ź]  | [ź']   | nadzy |
| 35  | [ź]  | [ź]  | [ź']   | zarno |
| 36  | [ń]  | [ń]  | [ń']   | magiczny |
| 37  | [ń]  | [ń]  | [ń']   | drozdzie |

In Poland, the first linguist who wrote about the possibility of phonetic interpretation of text by machines was W. Doroszewski in 1969 [35]. The largest contributions to solving the problem of automatic grapheme-to-phoneme conversion for Polish, were the publications of Maria Steffen-Batóg [36,37]. The first implementation of a grapheme-to-phoneme conversion algorithm for Polish, designed for the machine ODRA 1204, was made in 1971 by M. Warmus [38]. Further attempts to implement automatic grapheme-to-phoneme conversion have also been reported in various publications from that time to the present [39–45].

3. Methodology

Systems for converting graphemes to phonemes can be implemented in many ways, often roughly classified as either dictionary-based or knowledge-based (rule-based) strategies, although there are many intermediate solutions. Data-driven (dictionary-based) solutions involve storing as much phonological knowledge as possible in a lexicon, while rule-based systems consist of rules based on inference or proposed by expert linguists. Both dictionary-based and rule-based methods for converting graphemes to phonemes have their advantages and limitations. Searching for a word in the lexicon is relatively computationally inexpensive, whereas most rule-based system algorithms consume significantly more computational resources to produce a phoneme sequence. In addition, the dictionary method requires a large phonetic dictionary and complex morphophonemic rules, while the rule-based method is unable to model full morphophonemic constraints on its own. Very often, a rule-based system will incorporate a dictionary as an exception list. This solution was used in the grapheme-to-phoneme conversion application presented here.

3.1. Conversion Rules

The first set of rules for the conversion of graphemes into phonemes for the Polish language was developed by Maria Steffen-Batóg and presented in her set of monographs dedicated to automatic grapheme-to-phoneme conversion of texts in Polish [36,37]. Knowledge-based grapheme-to-phoneme conversion, unlike data-driven approaches, exploits rules derived by humans and/or from linguistic studies [46]. The rules for rule-based grapheme-to-phoneme are typically formulated in the framework of finite state automata [47]. The primary advantage of such rule-based approaches is that they can provide complete coverage. To achieve this objective it is crucial to use the algebra of sets and, for this purpose, it is necessary to define the basic sets of graphemes employed. Examples of these basic sets for the Polish language are presented below [37]. One of the basic sets is $Z$, $E$, and $S$. $Z$ is a set of alphabetic characters for Polish, $E$ is a set of three graphemes that are non-alphabetic for Polish, and $S$ is a set of special characters [37]:
\[ Z = \{ a, a, b, c, \check{c}, d, e, \check{e}, f, g, h, i, j, k, l, m, n, \check{n}, o, \check{o}, p, r, s, \check{s}, t, u, w, y, z, \check{z} \} \]  
\[ E = \{ q, v, x \} \]  
\[ S = \{ ., ?, !, ,, :, ;, - , (, ) , #, / \} \]  

The union of sets \( Z \cup S \) will be indicated by \( Z + S \) and defined as follows:

\[ Z + S = \{ \alpha : \alpha \in Z \lor \alpha \in S \} \]

Similarly, subtraction of sets \( Z \setminus S \) will be indicated by \( Z - S \). Auxiliary subsets \( O \subset S \) and \( P \subset S \) are also defined as follows:

\[ O = S - \{ / \} = \{ ., ?, !, ,, :, ;, - , (, ) , # \} \]
\[ P = O - \{ # \} = \{ ., ?, !, ,, :, ;, - , (, ) \} \]

The character “#” is a pause character between the words and the character “/” is a sign to separate characters within the word. The set \( X \) is a collection of alphabetic characters in Polish, including special characters:

\[ X = Z + E + S \]
\[ X = \{ a, a, b, c, \check{c}, d, e, \check{e}, f, g, h, i, j, k, l, m, n, \check{n}, o, \check{o}, p, r, s, \check{s}, t, u, w, y, z, \check{z}, \check{q}, v, x, ., ?, !, ,, :, ;, - , (, ) , #, / \} \]

The set \( X \) does not contain digits because all numbers are always converted to alphabetic representations. Let \( \Gamma \) and \( \Delta \) be arbitrary sets of words in the \( X \) alphabet. The concatenation of the sets \( \Gamma \) and \( \Delta \), written as \( \Gamma \cdot \Delta \) or \( \Gamma \Delta \) is an operation defined as follows:

\[ \Gamma \cdot \Delta = \Gamma \Delta = \{ \alpha \beta : \alpha \in \Gamma \land \beta \in \Delta \} \]

For any words \( \alpha, \beta, \gamma, \delta \) we receive:

\[ \{ \alpha, \beta \} \cdot \{ \gamma, \delta \} = \{ \alpha \gamma, \alpha \delta, \beta \gamma, \beta \delta \} \]

Here is an example to illustrate this type of relationship:

\[ \{ s, c \} \cdot A = \{ s, c \} A = \{ sa, sa, se, se, si, si, so, so, \check{s}o, su, sy, ca, ca, ce, ce, ci, co, \check{c}o, cu, cy \} \]

where \( A \) is the set of vowels in Polish:

\[ A = \{ a, a, e, \check{e}, i, o, \check{o}, u, y \} \]

It is also necessary to define empty words, written as 1, which do not contain any letters of the \( X \) alphabet. In this case, for any word \( \alpha \), equality is fulfilled:

\[ \alpha \cdot 1 = 1 \cdot \alpha = \alpha \]

The process of automatically converting graphemes to phonemes can be described as a function \( F \) defined by the following formula:

\[ F(\alpha) = \beta \]
where:

\[ a = a_1 \ldots a_k \ldots a_n \land a_k \in X \land (1 \leq k \leq n) \]  \hspace{1cm} (15)

\[ \beta = \beta_1 \ldots \beta_k \ldots \beta_m \land \beta_k \in Y \land (1 \leq k \leq m) \]  \hspace{1cm} (16)

and where \( a \) is a sequence of \( a \) alphabetic characters, \( \beta \) is a sequence of \( b \) phonemic characters, \( b \) is determined by \( a \), but \( a \) need not be equal to \( b \) at all. The set \( Y \) is the entire set of phonemic characters for Polish described by the SPA alphabet:

\[ Y = \{ i, y, e, a, o, u, \dot{a}, \dot{u}, r, l, m, n, \ddot{n}, \dddot{n}, f, v, s, z, \ddot{s}, \dddot{s}, \ddot{c}, \dddot{c}, \dddot{d}, b, t, d, k, g, \dot{k}, \dot{g}, c, \ddot{c}, \dddot{c}, \ddot{z}, \dddot{z} \} \]  \hspace{1cm} (17)

The grapheme-to-phoneme conversion of correctly written orthographic texts in Polish is the transformation of sequences written in the \( X \) alphabet, to a correct form in the phonemic alphabet \( Y \). The grapheme-to-phoneme conversion \( F \) function can be described by a set of formal grapheme-to-phoneme conversion rules defining how each sequence \( a \) representing a word in the \( X \) alphabet, can be transformed into a correct sequence \( \beta \) in the phonemic alphabet defined by the set \( Y \). The conversion rules are usually numerous with varying degrees of complexity. The size and complexity of a set of grapheme-to-phoneme conversion rules depends on the number of letters in the orthographic alphabet and the degree to which each letter can be pronounced differently in various contexts.

The first set of grapheme-to-phoneme conversion rules for Polish was developed by Maria Steffen-Batóg and presented in her set of monographs dedicated to automatic grapheme-to-phoneme conversion of texts in Polish [36,37]. The knowledge contained in these publications was the basis for developing the grapheme-to-phoneme conversion algorithm for Polish implemented by the author in the Python programming language [48].

According to Maria Steffen-Batóg, all grapheme-to-phoneme conversion rules relating to each orthographic letter can be stored in a matrix called the grapheme-to-phoneme conversion rule-table for the letter in question. The general form of a grapheme-to-phoneme conversion rule-table is shown in Table 2 [37].

Table 2. The general form of a grapheme-to-phoneme conversion rule-table [37].

| \( a_k \) | \( \Delta_1 \) | \( \ldots \) | \( \Delta_j \) | \( \ldots \) | \( \Delta_{m-1} \) | \( \Delta_m \) |
|-----------|----------------|----------------|----------------|----------------|----------------|----------------|
| \( \Gamma_1 \) | \( \beta_{1,1} \) | \( \ldots \) | \( \beta_{1,j} \) | \( \ldots \) | \( \beta_{1,m-1} \) | \( \beta_{1,m} \) |
| \( \Gamma_2 \) | \( \beta_{2,1} \) | \( \ldots \) | \( \beta_{2,j} \) | \( \ldots \) | \( \beta_{2,m-1} \) | \( \beta_{2,m} \) |
| \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) |
| \( \Gamma_i \) | \( \beta_{i,1} \) | \( \ldots \) | \( \beta_{i,j} \) | \( \ldots \) | \( \beta_{i,m-1} \) | \( \beta_{i,m} \) |
| \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) |
| \( \Gamma_{n-1} \) | \( \beta_{n-1,1} \) | \( \ldots \) | \( \beta_{n-1,j} \) | \( \ldots \) | \( \beta_{n-1,m-1} \) | \( \beta_{n-1,m} \) |
| \( \Gamma_n \) | \( \beta_{n,1} \) | \( \ldots \) | \( \beta_{n,j} \) | \( \ldots \) | \( \beta_{n,m-1} \) | \( \beta_{n,m} \) |

Table 2 was constructed with the following assumptions [37]:

\[ a_k \in X \]  \hspace{1cm} (18)

\[ \forall 1 \leq i \leq n : \Gamma_i \subset X \]  \hspace{1cm} (19)

\[ \forall 1 \leq j \leq m : \Delta_j \subset X \]  \hspace{1cm} (20)

\[ \forall 1 \leq i \leq n \land 1 \leq j \leq m : \beta_{i,j} \in Y \]  \hspace{1cm} (21)

The \( a_k \) letter in the upper left corner of the table indicates that it is the table of grapheme-to-phoneme conversion rules for the orthographic letter \( a_k \). The headings of the columns in the table \( (\Delta_1, \Delta_2, \ldots, \Delta_j, \ldots, \Delta_{m-1}, \Delta_m) \) represent various ways of phonetically transcribing the letter \( a_k \), depending on the letters following \( a \) in context. Similarly, the headings of the rows \( (\Gamma_1, \Gamma_2, \ldots, \Gamma_i, \ldots, \Gamma_{n-1}, \Gamma_n) \) determine various ways of phonetically transcribing the letter \( a_k \), depending on the letters preceding \( a_k \) in context. At the intersec-
tions of the rows and columns, one finds the grapheme-to-phoneme conversions for the letter \( \alpha_k \) in each specific orthographic context. For example, the intersection of the \( i \)-th row and \( j \)-th column indicates the element \( \beta_{i,j} \), which implies the following conversion rule:

- For each orthographic string \( s = \gamma \alpha_k \delta \), where \( \gamma \in \Gamma_i \) and \( \delta \in \Delta_j \), the grapheme-to-phoneme conversion process is defined as the assignment \( \alpha_k \rightarrow \beta_{i,j} \).

The symbol \( \beta_{i,j} \) in a specific conversion rule could represent a single phonemic letter, sequence of phonemic letters, or empty letter, written as symbol “1”. It is also worth remembering that certain combinations of orthographic characters in a particular language may be absent from the table. Therefore, there may be some orthographic contexts of the letter \( \alpha_k \) for which phonetic transcription cannot be determined. In such cases, specific fields in the table are left empty. This principle often allows the simplification of grapheme-to-phoneme conversion rules tables.

One of the simplest grapheme-to-phoneme conversion rules for Polish is the rule for orthographic letter “a” presented in Table 3. According to the grapheme-to-phoneme conversion rule defined in Table 3, each orthographic letter “a”, in all orthographic contexts (before or after any sequence of letters), should be converted into phoneme \([a]\) [37].

**Table 3.** The grapheme-to-phoneme conversion rule-table for the letter “a” in Polish [37].

| \( a \) | \( X \) | \( X \) | \( a \) |
|--------|--------|--------|--------|

An example of a more complex table containing nine grapheme-to-phoneme conversion rules, is the rule-table for letter “å” presented in Table 4 [37].

**Table 4.** The grapheme-to-phoneme conversion rule-table for letter “å” in Polish [37].

| \( a \) | \( \Delta_1 \) | \( \Delta_2 \) | \( \Delta_3 \) | \( \Delta_4 \) | \( \Delta_5 \) | \( \Delta_6 \) | \( \Delta_7 \) | \( \Delta_8 \) | \( \Delta_9 \) |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| \( X \) | o      | om     | on     | on     | on     | on     | on     | on     | oÅ    |

The definition of grapheme-to-phoneme conversion rules for letter “å” requires additional sets \( \Delta_1, ..., \Delta_9 \) defined as follows [37]:

\[
\Delta_1 = \{ l, ł, m \} \\
\Delta_2 = \{ p, b \} \\
\Delta_3 = \{ t, k, g \} \\
\Delta_4 = c(X - \{ i, h \}) \\
\Delta_5 = d(X - \{ z, ž \}) \\
\Delta_6 = dz(X - i) \\
\Delta_7 = \{ č, ci, dž, dzi \} \\
\Delta_8 = \{ s, ž, f, w, s, z, ž, ch \} \\
\Delta_9 = O
\]

Another example of a more complex rule table is the table for orthographic letter “u”. It consists of five parts, presented in Tables 5–8. This table contains 88 grapheme-to-phoneme conversion rules [37].
Table 5. The grapheme-to-phoneme conversion rule-table for orthographic letter “u” in Polish—part 1 [37].

| u   | X - {a, e} | X - / | / |
|-----|------------|-------|---|
|     | u          | u     | u |

Table 6. The grapheme-to-phoneme conversion rule-table for letter “u” in Polish—part 2 [37].

| u   | (X - {e, o, #})za | X |
|-----|-------------------|---|
|     | u                 | u |

Table 7. The grapheme-to-phoneme conversion rule-table for letter “u” in Polish—part 3 [37].

| u   | k | w | l | cz | sz | [s, m]O | t |
|-----|---|---|---|----|----|--------|---|
|     | u | u | u | u  | u  | u      | u |

Table 8. The grapheme-to-phoneme conversion rule-table for letter “u” in Polish—part 4 [37].

| u   | X - {t, k, w, l, c, s, m} | c(X - z) | s(X - (O + z)) | m(X - O) |
|-----|----------------------------|----------|----------------|---------|
|     | u                         | u        | u              | u       |

The most complex are the tables of rules for converting graphemes to phonemes in Polish for the following orthographic letters: “z”—174 rules, “s”—157 rules, “r”—91 rules, “u”—88 rules, and “d”—rules.

As presented in the literature by Steffen-Batóg, the grapheme-to-phoneme conversion rules are clear and universal [37], but they are correct only for Polish. For any other language, it is necessary to define that language’s own grapheme-to-phoneme conversion rules.

On the basis of Batóg’s work [36,37], the author of this paper implemented the grapheme-to-phoneme conversion rules for Polish in the Python programming language [49]. The implementation includes 975 conversion rules covering all 35 letters and special characters [48] of Polish orthography. Although many words have multiple correct pronunciations, the rules implemented here in this way produce only one (the most basic) pronunciation for any
given word. The list of implemented grapheme-to-phoneme conversion rules for Polish is presented in Table 9, where: $P$ is a set of special characters excluding \{#, @, −, /\}, “#” is a pause character at the end of a word, “@” is a pause character at the beginning of a word, “/” is a delimiter character and “−” is a dash character.

### Table 9. The list of implemented grapheme-to-phoneme conversion rules for Polish.

| Table Number | Orthographic Letter | No. of Rows | No. of Columns | No. of Cells | No. of Rules |
|--------------|---------------------|-------------|----------------|--------------|--------------|
| 1            | z                   | 34          | 64             | 2079         | 174          |
| 2            | s                   | 17          | 62             | 976          | 157          |
| 3            | r                   | 11          | 22             | 210          | 91           |
| 4            | u                   | 15          | 15             | 196          | 88           |
| 5            | d                   | 12          | 46             | 495          | 83           |
| 6            | i                   | 14          | 13             | 156          | 80           |
| 7            | Ć                    | 12          | 19             | 198          | 40           |
| 8            | z                   | 13          | 16             | 180          | 36           |
| 9            | c                   | 11          | 28             | 270          | 36           |
| 10           | t                   | 6           | 26             | 125          | 31           |
| 11           | k                   | 4           | 17             | 48           | 18           |
| 12           | f                   | 6           | 13             | 60           | 16           |
| 13           | w                   | 2           | 15             | 14           | 14           |
| 14           | ż                   | 3           | 9              | 16           | 13           |
| 15           | ś                   | 6           | 10             | 45           | 12           |
| 16           | p                   | 2           | 12             | 11           | 11           |
| 17           | b                   | 2           | 11             | 10           | 10           |
| 18           | g                   | 2           | 10             | 9            | 9            |
| 19           | e                   | 2           | 10             | 9            | 9            |
| 20           | a                   | 2           | 10             | 9            | 9            |
| 21           | n                   | 3           | 9              | 16           | 8            |
| 22           | y                   | 4           | 3              | 6            | 6            |
| 23           | l                   | 3           | 3              | 4            | 4            |
| 24           | p                   | 3           | 3              | 4            | 4            |
| 25           | h                   | 2           | 2              | 1            | 1            |
| 26           | l                   | 2           | 2              | 1            | 1            |
| 27           | ň                   | 2           | 2              | 1            | 1            |
| 28           | m                   | 2           | 2              | 1            | 1            |
| 29           | j                   | 2           | 2              | 1            | 1            |
| 30           | ó                   | 2           | 2              | 1            | 1            |
| 31           | ô                   | 2           | 2              | 1            | 1            |
| 32           | e                   | 2           | 2              | 1            | 1            |
| 33           | a                   | 2           | 2              | 1            | 1            |
| 34           | q                   | 2           | 2              | 1            | 1            |
| 35           | v                   | 2           | 2              | 1            | 1            |
| 36           | x                   | 2           | 2              | 1            | 1            |
| 37           | #                   | 2           | 2              | 1            | 1            |
| 38           | @                   | 2           | 2              | 1            | 1            |
| 39           | −                   | 2           | 2              | 1            | 1            |
| 40           | /                   | 2           | 2              | 1            | 1            |
| **TOTAL**    |                     |             |                | **5162**     | **975**      |

The list presented in Table 9 shows that the number of grapheme-to-phoneme conversion rules implemented is 975. The number of cells (potential letter-context combinations) in grapheme-to-phoneme conversion tables is much larger, here 5162. This means that some of them (i.e., 4187 cells) are empty and have no function in the grapheme-to-phoneme conversion process. Empty cells represent letter-context combinations not allowed in Polish.

### 3.2. Conversion Algorithm

The grapheme-to-phoneme conversion rules are the foundation of the grapheme-to-phoneme conversion algorithm implemented here [48]. The conversion algorithm defines how the function $F(α) = β$ works and determines the method for obtaining sequences of phonemic $β_k$ letters on the basis of orthographic sequences $α_k$ and their contexts. The block diagram of the grapheme-to-phoneme conversion algorithm for a single orthographic letter is presented in Figure 1.
The grapheme-to-phoneme conversion algorithm, using the general form of grapheme-to-phoneme conversion rules shown in Table 2, is shown here in terms of 12 steps:

1. Read an input orthographically represented word \( a \), where \( a = a_1a_2 \ldots a_k \ldots a_n \) and \( a_1, \ldots, a_n \in X \);
2. At the beginning of the orthographically represented word \( a \), let \( k = 1 \);
3. Read a single letter \( a_k \) of the input word \( a \);
4. Check the context of the letter \( a_k \) in the word \( a \):
   - Read a string of letters \( \gamma = a_1 \ldots a_{k-1} \) preceding the orthographic \( a_k \) letter;
   - Read a string of letters \( \delta = a_{k+1}a_{k+2} \ldots a_n \) following the orthographic \( a_k \) letter;
5. Select the appropriate grapheme-to-phoneme conversion rule table for the orthographic letter \( a_k \);
6. Select the appropriate \( j \)-th table’s column for the orthographic letter \( a_k \), where \( \delta \in \Delta_j \);
7. Select the appropriate \( i \)-th table’s row for the orthographic letter \( a_k \), where \( \gamma \in \Gamma_i \);
8. Select the appropriate grapheme-to-phoneme conversion rule for the orthographic letter \( a_k \), as defined by the cell in the \( j \)-th column and \( i \)-th row of the table;
9. Use the appropriate grapheme-to-phoneme conversion rule for the orthographic letter \( a_k \), as defined by the cell in the \( j \)-th column and \( i \)-th row of the table:
   - In the cell with the table coordinates \([i,j]\), phonemic letter \( \beta_{i,j} \) letter is saved, corresponding to the orthographic letter \( a_k \) with the following context \( \delta \) and preceding context \( \gamma \);
   - The grapheme-to-phoneme conversion rule for orthographic letter \( a_k \) will be implemented by assignment \( a_k \rightarrow \beta_k = \beta_{i,j} \);
10. Increment \( k = k + 1 \) and read the next orthographic letter \( a_k \) of the input word \( a \);
11. Go to step number 3, unless \( k > n \), where \( n \) is the length of input word \( a \);
12. Output the result of the grapheme-to-phoneme conversion of orthographic word \( a = a_1 \ldots a_n \) where \( a_1, \ldots, a_n \in X \), is a new phonemic representation of the word \( \beta = \beta_1 \ldots \beta_n \), where \( \beta_1, \ldots, \beta_n \in Y \).

These 12 steps summarize the automatic grapheme-to-phoneme conversion algorithm. This algorithm can also be presented as “pseudocode”, as shown in the list called Algorithm 1.
Algorithm 1. The algorithm for grapheme-to-phoneme conversion (implementation of function $F(\alpha) = \beta$).

**Require:** $\alpha = \alpha_1 \ldots \alpha_n$ and $\alpha_1, \ldots, \alpha_n \in X$

1: $n \leftarrow \text{length}(\alpha)$
2: $k \leftarrow 1$
3: while ($k \leq n$) do
4:   $\gamma \leftarrow \alpha_1 \ldots \alpha_{k-1}$
5:   $\delta \leftarrow \alpha_k \alpha_{k+1} \ldots \alpha_n$
6:   $\text{table} \leftarrow \text{tables}[^{\alpha_k}]$ [% selection of appropriate table]
7:   $R \leftarrow \text{rows}(\text{table})$ [% number of table rows]
8:   $C \leftarrow \text{cols}(\text{table})$ [% number of table columns]
9:   for all ($2 \leq i \leq R$) do
10:      if ($\gamma \in \Gamma_i$) then
11:         $I \leftarrow i$ [% selection of appropriate table row]
12:      end if
13:   end for
14:   for all ($2 \leq j \leq C$) do
15:      if ($\delta \in \Delta_j$) then
16:         $J \leftarrow j$ [% selection of appropriate table column]
17:      end if
18:   end for
19:   $\beta_k \leftarrow \text{table}[I,J]$ [% reading phoneme letter from the table]
20:   $k \leftarrow k + 1$
21: end while
22: $\beta \leftarrow \beta_1 \ldots \beta_{k-1}$ [% string of phonemes]
23: return $\beta$

4. Results

This grapheme-to-phoneme conversion algorithm for Polish was implemented in the Python programming language as an independent application called TransFon [48]. The developed application allows the grapheme-to-phoneme conversion of any orthographic text file in Polish.

Evaluation of automatic grapheme-to-phoneme conversion implementations is crucial. It is necessary to determine the application’s degree of success (accuracy). The evaluating and testing procedure for this automatic grapheme-to-phoneme conversion implementation consisted of the following elements:

- Performing grapheme-to-phoneme conversion of an orthographic text corpus file containing the most frequently used words in Polish (those with different orthographic representations), obtained from resources in the National Corpus of Polish [50];
- Validation and verification of the conversion results for those words using the Polish language dictionary that specifies the correct pronunciation of words in Polish;
- Registering cases of incorrect conversion, errors, and other problems encountered;
- Attempts to solve the problems.

This grapheme-to-phoneme conversion application was implemented in such a way that the conversion algorithm stopped when a grapheme-to-phoneme conversion problem occurred (e.g., when there was no rule allowing for a correct phonetic transcription). This made it easier to improve and develop the conversion application. In addition, any doubts about correct pronunciations were resolved with help of the aforementioned dictionary.

In order to evaluate the implemented grapheme-to-phoneme conversion system, the word error rate (WER), and phoneme error rate (PER) were used, as the methods usually used to evaluate the performance of automatic speech recognition systems. The WER value for this grapheme-to-phoneme conversion system was calculated as the ratio of the number of incorrectly converted words to the total number of words converted from the language corpus. The PER parameter was similarly defined and calculated for phonemes.
The causes of problems and errors in the automatic grapheme-to-phoneme conversion operation were as follows:

- Errors in the implementation of the algorithm or conversion rules;
- Missing conversion rules in tables (rules not included in the tables) for some orthographic letters in contexts;
- Problems with conversion of foreign words, acronyms and words that are not in the Polish language dictionary.

These problems were solved in the following ways:

- Implementation errors in the conversion algorithm and rule tables were corrected by modifying the application source code;
- The problem of missing conversion rules in tables has been solved by adding rules to the tables. It should be noted that the added grapheme-to-phoneme conversion rules cooperate with the rules implemented earlier and known from the literature [36,37]. In order to complete the rule tables, new rules were added for selected letters e.g.: “i”, “n”, “d”, “z”, “ż”, “ć”, “f”, “s” in some contexts, in particular:
  - adding or correcting conversion rules for the letter “i” for correct conversion of words: “unii”, “bedzie”, “sobie”, “razie”, “diabel”, and similar contexts;
  - adding or correcting conversion rules for the letter “n” for correct conversion of words: “branzy” and similar contexts;
  - adding or correcting conversion rules for the letter “d” for correct conversion When the letter “d” occurs at the end of a word, e.g.: “od”, “pod”, “przed”, “nad”, “miliard”, “wyjazd”, “Witold”, “grand”, “hold”, “prawd”;  
  - adding or correcting conversion rules for the letter “z” for correct conversion of words: “trzeci”, “zamierza”, “poradzę”, “bezzwrotny”, and similar contexts;
  - adding or correcting conversion rules for the letter “ż” for correct conversion of words: “tożsamość”, “manadżer”, and similar contexts;
  - adding or correcting conversion rules for the letter “ć” for correct conversion of words: “ćwiczenia”, “dziećmi”, “zadośćuczniać”, “ćwiartki”, and similar contexts;
  - adding or correcting conversion rules for the letter “f” for correct conversion of words: “Afganistan”, “Hoffman”, and similar contexts;
  - adding or correcting conversion rules for the letter “s” for correct conversion of words: “przeprosin”, “sina”, “Helsinki”, and similar contexts;
- The problem of converting foreign words and acronyms was solved by using the developed dictionary in which the rules for converting foreign words and acronyms were defined. Thus, the rule-based grapheme-phoneme conversion was supplemented by the dictionary method;
- Examples of grapheme-to-phoneme conversion rule-table additions implemented by the author are shown in Tables 10–17.

Table 10. The grapheme-to-phoneme conversion rule-table addition for the letter “i” in Polish for words: “unii”, “bedzie”, “sobie”, “razie”, “diabel”, and similar contexts.

| i   | A-{i} | A  | e | e | ab |
|-----|-------|----|---|---|----|
| [c,n]| 1     |    |   |   |    |
| dz  |       | 1  |   |   |    |
| b   |       |    | j|   |    |
| z   |       |    | 1|   |    |
| d   |       |    |   | j|    |
Table 11. The grapheme-to-phoneme conversion rule-table addition for the letter “n” in Polish for words: “branży” and similar contexts.

| n   | ˙z |
|-----|----|
| X   | łj |

Table 12. The grapheme-to-phoneme conversion rule-table addition for the letter “d” in Polish for words: “od”, “pod”, “przęd”, “nad”, “miliard”, “wyjazd”, “Witold”, “grand”, “rajd”, “huld”, “prawd”, and similar contexts.

| d   | S  |
|-----|----|
| A   | t  |
| {t, z, l, n, ł, w} | t |

Table 13. The grapheme-to-phoneme conversion rule-table addition for the letter “z” in Polish for words: “trzeci”, “zamierza”, “poradzę”, “bezzwrotny”, and similar contexts.

| z   | X | a | {æ, æ} | z |
|-----|---|---|--------|---|
| tr  | 1 |   | 1      |   |
| r   | 1 |   |        |   |
| d   |   |   | 1      | s |
| e   |   |   |        |   |

Table 14. The grapheme-to-phoneme conversion rule-table addition for the letter “ż” in Polish for words “tożsamość”, “manadżer”, and similar contexts.

| ż   | sa | er |
|-----|----|----|
| A   | ż  |    |
| d   | 1  |    |

Table 15. The grapheme-to-phoneme conversion rule-table addition for the letter “ć” in Polish for words: “ćwiczenia”, “dziećmi”, “zadośćuczynić”, “ćwiartki”, and similar contexts.

| ć   | wic | m | u | wi · A |
|-----|-----|---|---|--------|
| X   | ć   |   |   | ć     |
| e   | ć   |   |   | ć     |
| ś   | ć   |   |   | ć     |

Table 16. The grapheme-to-phoneme conversion rule-table addition for the letter “f” in Polish for words: “Afganistan”, “Hofmann”, and similar contexts.

| f   | ga | f  |
|-----|----|----|
| X   | v  | f  |

Table 17. The grapheme-to-phoneme conversion rule-table addition for the letter “s” in Polish for words: “przeprosin”, “siną”, “Helsinki”, and similar contexts.

| s   | in | iną | in |
|-----|----|-----|----|
| A   | ś  |     |    |
| #   | ś  | ś   |    |
| l   |    |     | s  |

A number of improvements have made it possible to increase the effectiveness of the grapheme-to-phoneme conversion process.

The results obtained were compared to the results available in the literature [51–55]. Table 18 presents WER and PER values of the developed grapheme-to-phoneme conversion system before improvements. Table 19 presents the WER and PER values of the developed
grapheme-to-phoneme conversion system after improvements. The summary of evaluation results for the system developed, before and after improvements, is presented in Table 20. The results presented in Table 20 indicate that conversion performance improved significantly after the implementation of the improvement.

Table 18. The WER and PER values of the developed grapheme-to-phoneme conversion system before improvements.

| No. | Parameter                                      | Value  |
|-----|-----------------------------------------------|--------|
| 1   | Number of unique words checked                | 1,943,458 |
| 2   | Number of G2P conversion errors for unique words | 33,638 |
| 3   | The WER value for unique words                | 1.731% |
| 4   | Number of words in the corpus                 | 230,300,300 |
| 5   | Number of G2P conversion errors for words in the corpus | 3,707,890 |
| 6   | The WER value for the corpus                  | 1.610% |
| 7   | Number of checked unique words phonemes       | 16,293,828 |
| 8   | Number of G2P conversion errors for phonemes | 34,324 |
| 9   | The PER value for unique words                | 0.211% |
| 10  | Number of phonemes in the corpus              | 1,263,992,460 |
| 11  | Number of G2P conversion errors for phonemes in the corpus | 3,713,206 |
| 12  | The PER value for the corpus                  | 0.294% |

Table 19. The WER and PER values of the developed grapheme-to-phoneme conversion system after improvements.

| No. | Parameter                                      | Value  |
|-----|-----------------------------------------------|--------|
| 1   | Number of unique words checked                | 1,943,458 |
| 2   | Number of G2P conversion errors for unique words | 7525  |
| 3   | The WER value for unique words                | 0.387% |
| 4   | Number of words in the corpus                 | 230,300,300 |
| 5   | Number of G2P conversion errors for words in the corpus | 69,802 |
| 6   | The WER value for the corpus                  | 0.030% |
| 7   | Number of checked unique words phonemes       | 16,282,255 |
| 8   | Number of G2P conversion errors for phonemes | 8063  |
| 9   | The PER value for unique words                | 0.050% |
| 10  | Number of phonemes in the corpus              | 1,263,415,734 |
| 11  | Number of G2P conversion errors for phonemes in the corpus | 73,786 |
| 12  | The PER value for the corpus                  | 0.006% |

Table 20. The summary evaluation results of developed grapheme-to-phoneme conversion system, before and after improvement.

| No. | Parameter                                      | Value before Improvements | Value after Improvements |
|-----|-----------------------------------------------|---------------------------|--------------------------|
| 1   | The WER value for unique words                | 1.731%                    | 0.387%                   |
| 2   | The WER value for the corpus                  | 1.610%                    | 0.030%                   |
| 3   | The PER value for unique words                | 0.211%                    | 0.050%                   |
| 4   | The PER value for the corpus                  | 0.294%                    | 0.006%                   |
As the final result of improvements to the G2P implementation, the software application called TransFon was developed, which allows automatic grapheme-to-phoneme conversion of any orthographic text file in Polish [48]. Numerous tests of this software application on a large orthographic text corpus [56], in a variety of operating environments, confirmed the accuracy of the application’s operation.

To demonstrate the operation of TransFon application, selected works of Polish literature were subjected to automatic grapheme-to-phoneme conversion. A sample fragment of Adam Mickiewicz’s epic poem “Pan Tadeusz” and a fragment of the output file obtained after automatic grapheme-to-phoneme conversion are presented with the printouts attached to this paper. An example fragment of an input text file containing an orthographic text from the epic poem “Pan Tadeusz” by Adam Mickiewicz is shown in Listing 1 [57].

Listing 1. A sample fragment of the input text file containing orthographic text from Adam Mickiewicz’s epic poem “Pan Tadeusz”.

```
Litwo! Ojczyzno moja! ty jesteś jak zdrowie.
Ile cię trzeba cenić, ten tylko się dowie,
Kto cię stracił. Dziś piękność twoją w całej ozdobie
Widzę i opisuję, bo tęsknię po~tobie.

Panno Święta, cojasnej bronzisz Częstochowy
I w Ostrej świeczis Branisz Ty, co gród zanikowy
Novogrodzki ochraniasz z jego wiernym ludem!
Jak mnie dziecko do zdrowia powróciłaś cudem
(Gdy od płaczącej matki pod Twoją opiekę
Ofiarowany, martwą podniesioś Twoją
I zaraz mogłem pieszcz do Twych świątyń progu
Iść za wrocłowe życie podziękować Bogu).

Tymczasem przenoś moją duszę utęsknioną
Do tych pagórków leśnych, do tych łąk zielonych,
Szeroko nad błękitnym Niemnem rozciągaj
Do tych pól malowanych zbożem rozmaitem,
Wyzłaczanych pszenicą, posrebrzanych żytem;
Gdy bursztynowy świerzop, gryka jak śnieg biała,
Gdy starożytnym rumieńcem dzięcielina pała,
A wszystko przepasane, jakby wstęgą, miedzą
Zieloną, na niej z rzadka ciche grusze siedzą.
```

An example fragment of the output file containing phoneme transcriptions of Adam Mickiewicz’s the epic poem “Pan Tadeusz” is shown in Listing 2.

Listing 2. A fragment of the output file containing phonemic notation as a result of automatic grapheme-to-phoneme conversion of Adam Mickiewicz’s epic poem “Pan Tadeusz”.

```
[litvo] [ojtS1rzno] [moja] [ti] [jestes'] [jak] [zdrojw]
[lito] [ts'e] [tSeba] [tsen'ste] [ten] [ts] [s'e] [dovye]
[st] [strats'iw] [dz'si'] [pjenkno's'ts'] [tvoe'] [t] [tsawej] [ozdobje]
[tsaw] [i] [opisuje] [bo] [ts'akn'e] [po] [tobye]

[panno] [s'vjenta] [tsol] [jasnej] [bron'si] [tSe'toxovi]
[i] [t] [ostrej] [s'vjet'si] [bram'e] [ti] [tsol] [grut] [zankovl]
[novogurts'j] [oxran'ad] [s] [jego] [sjernln] [ludeng]
[jak] [mun'e] [dz'etsko] [do] [zdrojwa] [povrt'siuse] [tsudem]
[gt] [ot] [punktonteje] [matk'i] [pot] [tvoyo] [opjeke]
[ojaroian] [martvov] [podm'oswem] [povjeke]
[ts] [zaraz] [magwen] [psjeno] [do] [tv] [s'vjonln'] [prugw]
[ts] [sa] [vrutson] [zits'we] [podz'enkovats'] [bogu]
[tak] [nas] [povrts'i] [tsudem] [sa] [ojtS1rzln] [vono]
[tsntSenen] [plenos'] [noje] [mube] [tu's'krono]
[do] [tix] [pargurk] [les'nix] [do] [tix] [konk] [z'eloni]
[Seroko] [nat] [bvenk'tlnln] [n'emmen] [rots'tongn'onix]
[do] [tix] [pol] [malovani] [zhoZen] [rozmaiten]
[vrznatamln] [pemen'tsoe] [pozrehznit] [ziten]
[gtze] [burStinoiv] [s'vjeZop] [grikal] [jak] [s'n'ek] [bjava]
[gtze] [pan'en's'tim] [runjen'utm] [dz'em'ts'elina] [pava]
[ts] [filoto] [popepana] [yagbl] [rstengo'] [njezdoe]
[s'elono] [na] [n'ej] [z] [zatka] [tis'ixe] [groue] [s'edzoe]
```


One measure of the performance of an automatic conversion of graphemes into phonemes algorithm is word processing speed (WPS). The WPS value for the application TransFon was estimated by measuring the time required to convert the text, Adam Mickiewicz’s epic poem “Pan Tadeusz”, containing 10,972 lines of text and 70,291 words [57]. Automatic grapheme-to-phoneme conversion of the whole text file took about 38 s. It is possible to calculate that the TransFon application can process text files at an average speed of 1849 words per second, which is 110,940 words per minute:

$$WPS = \frac{70291 \text{ [words]}}{38 \text{ [seconds]}} \approx 1849 \text{ [words/second]} = 110940 \text{ [words/minute]}$$ (31)

The measurement of grapheme-to-phoneme conversion speed was performed on a PC workstation with CPU Intel Core i7 4770K @ 3.50 GHz. The computational power of this CPU is around 130,000 MIPS.

Another application of the TransFon software can be assistance in language corpora processing. Development of statistical methods for speech recognition requires access to large language corpora [58]. One of the largest corpora available for the Polish language is the National Corpus of Polish (NCP) [56]. The NCP corpus is available to the scientific community, offers great flexibility, and has great scientific value. This corpus provides the scientific community, particularly linguists, as well as computer scientists interested in natural language processing, materials showing the contemporary state of the Polish language, meeting all the scientific requirements. Word-frequency lists obtained using the National Corpus of Polish seem particularly valuable. Automatic grapheme-to-phoneme conversion allows creating large phonemic language corpora from orthographic language corpora. A phonemic language corpus for Polish was developed by the author using automatic grapheme-to-phoneme conversion of an orthographic language corpus, in order to be able to perform statistical phonological analysis of the Polish language, and to develop phoneme-based statistical language models for Polish to improve automatic speech recognition [17–19].

A sample fragment of the frequency list for the phonemic corpus of Polish developed is presented in Listing 3. The phonetic frequency list file contains 19,43,458 Polish words written orthographically, their phonetic transcriptions in the SAMPA phonemic alphabet, and additionally the number of each word occurring in the NCP source corpus. The total number of word-tokens in the NCP corpus is 230,300,300. This describes a plain text version of the NCP corpus file containing 230 million words in Polish. It should also be noted that, the standard SAMPA transcriptions for Polish include several sequences of phonetic transcription labels that may cause ambiguity unless separated by spaces or other characters. To avoid this problem, the individual phonemes were separated by square brackets.

Making the developed rule-based implementation available could be very useful to the community—both for linguistic research and for more in-depth performance comparisons between the current rule-based solution and others. Therefore, it is planned to make TransFon application available after making necessary corrections in the implemented algorithm and determining the conditions and license of making the application available for other users.
Listing 3. A sample fragment of the frequency list file of the phonemic language corpus developed for Polish.

```
7692997 w [f]
5333210 i [i]
4235003 na [na]
4158902 z [a]
3981525 się [s'e]
3601719 nie [n'e]
2904114 do [do]
2205896 ze [Ze]
2171877 to [to]
1731304 o [o]
1728527 jest [jest]
1662839 a [a]
1003027 jak [jak]
983396 po [po]
912660 od [ot]
877522 ale [ale]
847373 za [za]
775006 przez [pSes]
764024 co [tso]
663771 dla [dia]
645573 czy [tSI]
610035 tym [tIm]
607673 już [juS]
544429 się [s'e]
544343 tak [tak]
534509 tylko [t1ko]
500801 ma [m]
475172 noże [noZe]
461226 tego [tego]
445705 ze [ze]
426201 jego [jego]
413855 oraz [oras]
392445 bo [bo]
380741 które [kture]
386863 będzie [ben'dz'e]
376600 ich [ix]
367282 tej [tej]
359528 było [bIwo]
358750 też [teS]
357026 który [kturI]
348641 jeszcze [jedStSe]
348423 był [bIt]
346507 być [bIts '
346596 jednak [jednak]
345989 przy [pSI]
...```

5. Statistical Approach

Various grapheme-to-phoneme conversion methods can be used, but three types are most important and most often used: rule-based methods, dictionary-based methods, and statistical methods [51].

Rule-based grapheme-to-phoneme conversion methods may be complemented by dictionary-based and statistical methods. The rule-based grapheme-to-phoneme conversion system discussed in this paper was complemented by the dictionary-based method. Statistical methods can also improve grapheme-to-phoneme conversion for Polish in the future. For this purpose, the author performed statistical analysis of the grapheme-to-phoneme conversion rules in Polish.

For better analysis of the obtained results, a special naming scheme for conversion rules was adopted. The naming conventions of the grapheme-to-phoneme conversion rules were as follows:

\[ R_i = L_R_C_P \] (32)

where: \( R_i \) is the name of the rule, \( L \) is an orthographic letter to which the rule applies, \( R \) is a row number in the grapheme-to-phoneme conversion table, \( C \) is a column number in the grapheme-to-phoneme conversion table, and \( P \) is a phonemic letter, to which the rule applies. A sample list of grapheme-to-phoneme conversion rules for letter “ą” using these naming conventions is presented in Table 21.
The frequency of each $R_i$ grapheme-to-phoneme conversion rule was calculated during automatic grapheme-to-phoneme conversion of an orthographic text corpus file in Polish containing 1,843,069,533 letters, obtained from the National Corpus of Polish. A sample list of the most frequently used grapheme-to-phoneme conversion rules in Polish is presented in Table 22.

The frequencies of grapheme-to-phoneme conversion rules used in Polish are presented in Figure 2.

Table 21. A sample list of grapheme-to-phoneme conversion rules for orthographic letter “ą” [37].

| No. | Rule Name $R_i$ | Orthographic Letter | Row Number $L$ | Column Number $R$ | Phoneme Letters $P$ |
|-----|-----------------|---------------------|----------------|-------------------|-------------------|
| 1   | a_1_1_o         | a                   | 1              | 1                 | [o]               |
| 2   | a_1_2_om        | a                   | 1              | 2                 | [om]              |
| 3   | a_1_3_on        | a                   | 1              | 3                 | [on]              |
| 4   | a_1_4_on        | a                   | 1              | 4                 | [on]              |
| 5   | a_1_5_on        | a                   | 1              | 5                 | [on]              |
| 6   | a_1_6_on        | a                   | 1              | 6                 | [on]              |
| 7   | a_1_7_on’       | a                   | 1              | 7                 | [on’]             |
| 8   | a_1_8_o˜        | a                   | 1              | 8                 | [o˜ ]             |
| 9   | a_1_9_o˜        | a                   | 1              | 9                 | [o˜ ]             |

Table 22. A sample list of the most frequently used grapheme-to-phoneme conversion rules in Polish.

| No. | Number of Occurr. of $i$ | Frequency of Occurr. in % $f(R_i) \cdot 100$ | Rule Name $R_i$ |
|-----|--------------------------|-----------------------------------------------|-----------------|
| 1   | 230609384                | 12.512                                       | a_1_1_          |
| 2   | 230606397                | 12.512                                       | a_1_1_          |
| 3   | 120713048                | 6.580                                        | a_1_1_a         |
| 4   | 107031055                | 5.807                                        | e_1_1_e         |
| 5   | 102221066                | 5.546                                        | o_1_1_o         |
| 6   | 52025013                 | 2.823                                        | y_3_2_          |
| 7   | 50499227                 | 2.740                                        | i_12_11_i       |
| 8   | 47433333                 | 2.574                                        | r_1_1_r         |
| 9   | 44069532                 | 2.391                                        | t_1_1_t         |
| 10  | 43352871                 | 2.352                                        | n_1_1_n         |
| 11  | 38974957                 | 2.115                                        | n_1_1_m         |
| 12  | 38920643                 | 2.112                                        | w_1_1_v         |
| 13  | 34811970                 | 1.889                                        | P_1_1_           |
| 14  | 31876002                 | 1.730                                        | j_1_1_j         |
| 15  | 30407964                 | 1.650                                        | u_1_1_u         |
| 16  | 28149906                 | 1.527                                        | n_1_7_n’        |
| 17  | 26241250                 | 1.424                                        | p_1_1_p         |
| 18  | 24158723                 | 1.311                                        | z_2_23_         |
| 19  | 23770872                 | 1.290                                        | l_1_1_w         |
| 20  | 21373108                 | 1.160                                        | k_1_2_k         |
| 21  | 21286163                 | 1.155                                        | l_1_1_l         |
| 22  | 20689142                 | 1.123                                        | d_1_1_d         |
| 23  | 16884069                 | 0.916                                        | s_1_2_s         |
| 24  | 16046642                 | 0.871                                        | i_3_2_          |
| 25  | 15310336                 | 0.831                                        | c_1_3_ts        |
The values presented in Table 22 depend on the frequencies of letters in Polish orthography, which were also calculated for the orthographic language corpus used. The frequencies of the letters of Polish orthography are presented in Table 23 and Figure 3.
Table 23. The frequencies of letters in Polish orthography.

| No. | Number of Occurr. of 1345943574 | Frequency of Occurr. in % | Letter |
|-----|---------------------------------|---------------------------|--------|
| 1   | 120646654                       | 8.96245                   | a      |
| 2   | 111589675                       | 8.28964                   | i      |
| 3   | 106665716                       | 7.92385                   | e      |
| 4   | 102135976                       | 7.58735                   | o      |
| 5   | 76298546                        | 5.66797                   | z      |
| 6   | 75403146                        | 5.60146                   | n      |
| 7   | 61445747                        | 4.56461                   | u      |
| 8   | 61016675                        | 4.53273                   | r      |
| 9   | 57237441                        | 4.25198                   | s      |
| 10  | 53549355                        | 3.97801                   | c      |
| 11  | 53399610                        | 3.96688                   | t      |
| 12  | 52240953                        | 3.88081                   | y      |
| 13  | 45634859                        | 3.39007                   | k      |
| 14  | 44605635                        | 3.31361                   | d      |
| 15  | 41733129                        | 3.10022                   | p      |
| 16  | 38973075                        | 2.89518                   | m      |
| 17  | 31851311                        | 2.36613                   | j      |
| 18  | 31481368                        | 2.33865                   | u      |
| 19  | 28274720                        | 2.10044                   | l      |
| 20  | 23763998                        | 1.76535                   | ì      |
| 21  | 19865437                        | 1.47574                   | b      |
| 22  | 18405166                        | 1.36726                   | g      |
| 23  | 15475239                        | 1.14961                   | q      |
| 24  | 14222227                        | 1.05652                   | h      |
| 25  | 13906034                        | 1.03303                   | q      |
| 26  | 12130116                        | 0.90111                   | ż      |
| 27  | 11204643                        | 0.83236                   | ó      |
| 28  | 9280005                         | 0.68938                   | å      |
| 29  | 6119384                         | 0.45459                   | ć      |
| 30  | 4087022                         | 0.30361                   | f      |
| 31  | 2474196                         | 0.18380                   | ñ      |
| 32  | 826516                          | 0.06140                   | ź      |
| 33  | 114138                          | 0.00848                   | v      |
| 34  | 65450                           | 0.00486                   | x      |
| 35  | 11434                           | 0.00085                   | q      |

However, Zipf’s law does not well describe the distributions of orthographic letters and phonemes in representations of words. The examination of such frequencies for 95 languages, as found in the literature [60], shows that phonemic and orthographic letter frequencies are best described by an equation first developed by Yule, which also describes the distribution of DNA codons [61].

The frequencies of letters in a language’s orthography are well described by Yule’s equation [60]:

$$ Y_r = \frac{a}{r^b} \cdot c^r $$

(34)

where $Y_r$ is the frequency of the letter, $r$ is the rank of the letter, if frequencies are ranked from highest ($r = 1$) to lowest ($r = n$), and $a$, $b$, and $c$ are parameters estimated from the statistical data. The fit of Yule’s equation to the ranked frequency distribution of Polish orthographic letters is presented in Figure 4.
Figure 3. The frequencies of letters in Polish orthography.

Figure 4. The fit of Yule’s equation to the ranked frequency distribution of the Polish orthographic letters.
The average fit of Yule’s equation to the ranked frequency distribution of the Polish letters was measured by the coefficient of determination $R^2$. The coefficient of determination for the fit of Yule’s equation, presented in Formula (34), correctly yields the ranked frequency distribution of Polish letters:

$$R^2 = 0.97509$$

(35)

Additionally, the root mean square error $RMSE$ value was calculated for this case:

$$RMSE = 2.9071 \cdot 10^{-3}$$

(36)

The $R^2$ value indicates how well statistical data fit into a statistical model. The $R^2$ value here is equal to 0.97509 and indicates that Yule’s equation fits very well the obtained statistical data on orthographic letter frequencies in Polish.

Analogous regularity was observed for the frequency distribution of the grapheme-to-phoneme conversion rules for Polish presented in Table 22. Figure 5 presents the fit of Yule’s equation to the ranked frequency distribution of grapheme-to-phoneme conversion rules used for Polish.

The summary of evaluation results for the fit of Yule’s equation to the ranked frequency distribution of orthographic letters (1) and grapheme-to-phoneme conversion rules in Polish (2) is presented in Table 24.

Table 24. The evaluation results of the fit of Yule’s equation to the ranked frequency distribution of orthographic letters (1) and grapheme-to-phoneme conversion rules (2) in Polish.

| No. | Yule’s Equation | $R^2$  | $RMSE$   |
|-----|----------------|--------|----------|
| 1   | $Y_r = \frac{0.066783}{r} \cdot 0.91^r$ | 0.97509 | $2.9071 \cdot 10^{-3}$ |
| 2   | $Y_r = \frac{0.12512}{r^{0.954}}$ | 0.95107 | $1.8022 \cdot 10^{-3}$ |
The values of $R^2$ presented in Table 24 indicate that Yule’s equation fits the obtained statistical data for the frequencies of orthographic letters and the grapheme-to-phoneme conversion rules in Polish. On this basis, it can be concluded that the data obtained from statistical analysis of grapheme-to-phoneme conversion rules used in Polish, based on an orthographic language corpus, are reliable.

The results presented, from the statistical analysis of grapheme-to-phoneme conversion rules in Polish, represent the basis for a statistical approach to Polish G2P in a future implementation.

6. Conclusions

The results of the grapheme-to-phoneme conversion research presented in this paper were compared to other results published in the literature [3,27,39,41–43,45,51,53,55,62–73]. On the basis of this comparison, the following conclusions can be drawn:

- Automatic conversion of graphemes into phonemes in orthographic texts is not only a technical issue, consisting in developing appropriate algorithms for converting graphemes into phonemes, but also a serious linguistic problem. Only specialists in linguistics and phonetics of a given language are able to formulate appropriate rules for converting graphemes into phonemes for speech [51];
- An additional complication is that automatic conversion of graphemes to phonemes is a language-specific problem with different spelling and pronunciation conventions within the same language [55,68–70];
- Effective solutions for automatic grapheme-to-phoneme conversion in one language may not help solve the same problems for a different language. There is not only one language and technical problem of automatic conversion of graphemes to phonemes to be solved, but many different problems with different levels of difficulty that should be solved for each language separately [51];
• Automatic grapheme-to-phoneme conversion is widely used not only in speech synthesis, but also in speech recognition [3,53];
• A separate, but very important problem is the evaluation of grapheme-to-phoneme conversion processes [53,71]. Evaluation and validation of grapheme-to-phoneme conversion implementations is a laborious and time-consuming process. All problems registered for the G2P implementation discussed in this paper were positively resolved;
• The G2P implementation developed for this research is not the only one for Polish [27,39,41,43,45], however only one of the others is available for free use [41];
• The author of the paper analysed for comparison the only available application for the Polish language, named Transcriber [41]. The application was implemented in the C++ programming language. The implemented method uses a dictionary of 5018 words and 767 defined conversion rules. For comparison, the software presented by the author in this paper was implemented in Python programming language, 975 conversion rules were implemented and the dictionary is very limited and plays only a supporting role. This means that TransFon has implemented 208 more transcription rules, which is over 27% more. The application failed to compile due to the lack of inclusion in the source code of the appropriate libraries that were used by the programmer to create the application. This made it impossible to evaluate the correctness of the application and seriously hindered the comparison with the software created by the author of the paper; However, based on the analysis of the application’s source code, you can see that the principle of the application is also rule-based, but the author of the Transcriber application tried to refine and improve the application’s performance by adding new words to the dictionary (exceptions). The author of the TransFon application, on the other hand, tried to add and supplement transcription rules in a similar way as is known in the literature. This is evidenced by the dictionary size used in both applications;
• The G2P system presented here could be used for Polish corpus development;
• The G2P implementation presented here did not exploit any similar pre-existing tools [48];
• It is worth noting that the solutions presented here for the development of language and speech corpora in Polish are not the only ones and publications on this subject are available [72,73];
• Of particular interest are the results presented in publications by Grażyna Demenko et al. [39,62–67].

This paper presents a rule-based method for grapheme-to-phoneme conversion and an implementation for Polish. The major original author’s achievements presented in this paper are as follows:
• Implementation of the known from the linguistic literature rules of converting graphemes into phonemes for the Polish language in the Python programming language [36,37,40,44];
• Developing an algorithm for automatic conversion of graphemes into phonemes for the Polish language and implementing it in the Python programming language with numerous improvements;
• Development of a software for automatic conversion of graphemes into phonemes called TransFon, which enables automatic conversion of graphemes into phonemes of any orthographic text files in the Polish language;
• Application of the developed methods to create phoneme-based language corpora using the automatic conversion of graphemes to phonemes;
• Statistical analysis of the occurrence frequency of particular grapheme-to-phoneme conversion rules in Polish;
• Comparison of the results obtained with those published in the literature and discussion.

It should be noted that the research presented in this article used basic principles and the fundamental grapheme-to-phoneme conversion rules developed by Maria Steffen-Batóg and presented in her set of monographs dedicated to the automatic grapheme-to-phoneme conversion of texts in Polish [36,37]. The author used previously developed rules and developed independently the grapheme-to-phoneme conversion algorithm and software
application. The application allows to convert any text in Polish orthography to the corresponding strings of phonemes as well as creating large phonemic language corpora based on orthographic language corpora.

The system for rule-based grapheme-to-phoneme conversion implemented here is complemented by dictionary-based methods, and was used to obtain statistics for the use of grapheme-to-phoneme conversion rules in Polish, potentially enabling the improvement of grapheme-to-phoneme conversion for Polish in the future.

The grapheme-to-phoneme conversion system developed and its ability to create phonemic language corpora for Polish open up further opportunities for research on improving automatic speech recognition in Polish. The plan for further research towards achieving this goal, using the phonemic language corpus developed, includes:

• Performing a better and more detailed statistical analysis of the Polish language based on the phonemic language corpus developed [17,19];
• Developing more efficient word-based and phoneme-based statistical language models for speech recognition applications in Polish [18,19];
• Application of deep learning methods to language modelling and speech recognition [20,21].

The main problem in the development of phoneme-based statistical language models for Polish is the difficulty in obtaining sufficiently large phonemic language corpora. The phonemic language corpus development method presented in this paper, based on automatic grapheme-to-phoneme conversion, can significantly remedy this problem.

Funding: This work was supported by the Polish Ministry of Science and Higher Education funding for statutory activities on Silesian University of Technology.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The author declares no conflict of interest.

References
1. Hirschberg, J.; Manning, C.D. Advances in natural language processing. Science 2015, 349, 261–266.
2. Lee, F. Automatic grapheme-to-phone translation of english. J. Acoust. Soc. Am. 1967, 41, 1594A.
3. Bagshaw, P. Phonemic transcription by analogy in text-to-speech synthesis: Novel word pronunciation and lexicon compression. Comput. Speech Lang. 1998, 12, 119–142.
4. Kawaguchi, Y.; Takagaki, T.; Tomimori, N.; Tsuruga, Y. Corpus-Based Perspectives in Linguistics. In Usage-Based Linguistic Informatics; John Benjamins Publishing Company: Amsterdam, The Netherlands, 2007.
5. Kłosowski, P. Speech Processing Application Based on Phonetics and Phonology of the Polish Language. In Communications in Computer and Information Science, Proceedings of the 17th International Conference Computer Networks, Ustron, Poland, 15–19 June 2010; Kwiecien, A., Gaj, P., Stera, P., Eds.; Computer Networks; Springer: Berlin, Germany, 2010; Volume 79, pp. 236–244.
6. Kłosowski, P. Improving speech processing based on phonetics and phonology of Polish language. Prz. Elektrotech. 2013, 89, 303–307.
7. Izydorczyk, J.; Kłosowski, P. Acoustic properties of Polish vowels. Bull. Polish Acad. Sci. Tech. Sci. 1999, 47, 29–37.
8. Izydorczyk, J.; Kłosowski, P. Base acoustic properties of Polish speech. In Proceedings of the International Conference Programable Devices and Systems PDS2001 IFAC Workshop (IFAC 2001), Giwice, Poland, 22–23 November 2001; pp. 61–66.
9. Kłosowski, P.; Dustor, A.; Izydorczyk, J.; Kotas, J.; Slimok, J. Speech Recognition Based on Open Source Speech Processing Software. In Communications in Computer and Information Science, Proceedings of the 21st International Science Conference on Computer Networks (CN), Brunow, Poland, 23–27 June 2014; Kwiecien, A., Gaj, P., Stera, P., Eds.; Computer Networks, CN; Springer: Berlin, Germany, 2014; Volume 431, pp. 308–317.
10. Dustor, A.; Kłosowski, P. Biometric Voice Identification Based on Fuzzy Kernel Classifier. In Communications in Computer and Information Science, Proceedings of the 20th International Conference on Computer Networks (CN), Lwouek Slaski, Poland, 17–21 Jun 2013; Kwiecien, A., Gaj, P., Stera, P., Eds.; Computer Networks, CN; Springer: Berlin, Germany, 2013; Volume 370, pp. 456–465.
11. Dustor, A.; Kłosowski, P.; Izydorczyk, J. Influence of Feature Dimensionality and Model Complexity on Speaker Verification Performance. In Communications in Computer and Information Science, Proceedings of the 21st International Science Conference on Computer Networks (CN), Brunow, Poland, 23–27 June 2014; Kwiecien, A., Gaj, P., Stera, P., Eds.; Computer Networks, CN; Springer: Berlin, Germany, 2014; Volume 431, pp. 177–186.
64. Demenko, G.; Bachan, J.; Wagner, A.; Wyroślak, P. Speech corpus creation for automatic analysis of phonetic convergence. In Studientexte zur Sprachkommunikation, Proceedings of 27th Conference on Electronic Speech Signal Processing (ESSV), Leipzig, Germany, 2–4 March 2016; Oliver, J., Ed.; Hochschule für Telekommunikation Leipzig (HTTL): Leipzig, Germany, 2016; pp. 183–190.
65. Demenko, G.; Grocholewski, S.; Klessa, K.; Rau, Z. Polish language resources for speech technology: Jurisdict Language corpora. In Human Language Technologies as a Challenge for Computer Science and Linguistics, Proceedings of the 4th Language & Technology Conference, Poznań, Poland, 6–8 November 2009; Zygmunt, V., Ed.; Adam Mickiewicz University, Poznań, 2009; pp. 165–169.
66. Demenko, G.; Klessa, K.; Szymański, M.; Breuer, S.; Hess, W. Polish unit selection speech synthesis with boss: Extensions and speech corpora. Int. J. Speech Technol. 2010, 13, 85–99.
67. Demenko, G.; Szymański, M.; Cecko, R.; Lange, M.; Klessa, K.; Owsianny, M. Development of large vocabulary continuous speech recognition using phonetically structured speech corpus. In Proceedings of the 17th International Congress of Phonetic Sciences (ICPhS XVII), Hong Kong, China, 17–21 August 2011; pp. 568–571.
68. Kosaner, O.; Birant, C.C.; Aktas, O. Improving Turkish language training materials: Grapheme-to-phoneme conversion for adding phonemic transcription into dictionary entries and course books. In Procedia Social and Behavioral Sciences, Proceedings of the 13th International Educational Technology Conference, Lisbon, Portugal, 30 October–1 November 2014; Isman, A., Siraj, S., Kiyici, M., Eds.; Volume 103, pp. 473–484.
69. Lee, J.; Kim, B.; Lee, G.G. Hybrid Approach to Grapheme to Phoneme Conversion for Korean. In Proceedings of the InterSpeech 2009: 10th Annual Conference of the International Speech Communication Association 2009, Brighton, UK, 6–10 September 2009; Volume 1–5. pp. 1299–1302.
70. de Jesus Aguiar Pontes, J.; Furui, S. Predicting the phonetic realizations of word-final consonants in context—A challenge for French grapheme-to-phoneme converters. Speech Commun. 2010, 52, 847–862.
71. Schraagen, M.; Bloothooft, G. A qualitative evaluation of phoneme-to-phoneme technology. In Proceedings of the 12th Annual Conference of the International-Speech-Communication-Assocation 2011 (Interspeech 2011), Florence, Italy, 27–31 August 2011; Volume 1–5. pp. 2332–2335.
72. Żelasko, P.; Ziółko, B.; Jadczyk, T.; Skurzok, D. AGH corpus of Polish speech. Lang. Resour. Eval. 2016, 50, 585–601.
73. Ziółko, B.; Jadczyk, T.; Skurzok, D.; Żelasko, P.; Gałka, J.; Pędzimaz, T.; Gawlik, I.; Pałka, S. SARMATA 2.0 automatic Polish language speech recognition system. In Proceedings of the Sixteenth Annual Conference of the International Speech Communication Association, Dresden, Germany, 6–10 September 2015; Interspeech: Dresden, Germany, 2015.