Preferred line options based on the PhLADiPreLiO using Haskell

Oleksandr Zhabenko

June 1, 2023

Author and software developer: Oleksandr Zhabenko
License: MIT

Abstract

The paper proposes an original approach to the task of constructing texts with given phonetic properties, which allows you to work efficiently and quickly on the problem, offering interesting solutions in both content and engineering. Elements of the theory are constructed, which is illustrated by the use of Haskell programs to significantly speed up the necessary calculations.

Keywords

Phonetic languages; prosodic languages; phonetics; prosody; syllables; rhythm; metre; verse; text; coherent states; syllable-as-a-whole (SaaW); phonetic-phenomena-as-a-whole (PhoPaaW); Haskell; phonetic languges approach to discovering preferred line options – PhLADiPreLiO – phladiprelio – the pronunciation is phlah - deeh - preh - ’leeh -oh, the emphasis (dynamic stress) on the second ’i’).

Introduction

Formulation of the problem

The object of research is various existing and potentially new human languages, as well as music.

The subject of research is methods of creating (changing) texts and sequences, which allow to obtain those that better meet the criteria that are based on sound, in particular, more rhythmic, sweet-sounding, more complex or easier to pronounce, more phonetically expressive, etc.

The main research method is mathematical modeling and computer programming using the proposed models.

The basis of the research was carried out for the Ukrainian language, while generalizations for other languages were proposed. Musical sequences have not been studied practically, but the obtained results can be transferred to them, taking into account the differences.

The relevance of the research is determined by the need to create various texts that have certain properties in terms of form, phonetic and prosodic sounding, for example, more or less rhythmic, better suited for music, poetry, speech therapy and logotherapy, etc.

Basic idea

There are different languages. They have different structure and rules. It is possible to create and use (based on one of existing widely used and well-spoken languages, in particular Ukrainian in this work) a 'phonetic' language that is better suited for poetry and music. It is even possible to create different versions of phonetic language. This paper proposes to create several different phonetic languages based on Ukrainian.

What does this mean?

If someone builds a phrase in a language that violates the rules of grammar or semantics, then this error is visible to a skilled speaker at once, it is identified as such almost instantly. Instead, if the sound of a phrase has some phonetic features, not counting accents, for example, the complexity of pronunciation or vice versa lightness, smoothness or abruptness, etc., then it is possible to identify it as an error or something significant not immediately or with special attention. One can imagine this as giving preference to the language semantics (meaning) and grammar, but less weight to phonetics. Phonetic language is that one built specifically to enhance the meaning and importance of the phonetic component itself.

Phonetic or prosodic language?

An interesting question is whether to call the approach "phonetic" or "prosodic" languages (Gross, n.d.). But I must say that we study the actual phonetic features, what is associated with the sound of speech. Among them is that which concerns certain phonetic phenomena in the general case, in particular phonemes or even palatalization. These questions are generally not the subject of the study of prosody as a science, as a certain component of phonetics, but are the subject of a broader study of phonetics. Moreover, there are no restrictions and bindings of the proposed approach to the actual syllables, which is more typical for the subject of the study of prosody. Generalizations in the package phladiprelio-general-simple can be made for more general cases.

However, at this stage of development, the vast majority of information here relates to or is directly related to syllables and prosody. Therefore, I leave the name ”phonetic languages”, given that prosody is a more specific branch of phonetics.

Ethical component

The proposed approach is similar to the approach of music theory. Thus, in music, among all sounds, musical ones stand out, later consonances and dissonances are studied, later notes, intervals, chords, melodies, composition, etc. There are recommendations, but they do not bind the creators, but help. Similarly, the proposed approach is designed to provide such assistance. Its strangeness at first glance cannot be a reason to deny it.

For Christians, to whom the author himself belongs, the words of Moses are important: ”And Moses went out, and spake unto the people the word of the LORD, and gathered together seventy men of the elders of the people, and set them near the tabernacle. And the LORD came down in the cloud, and spake unto him, Two of the men remained in the state, one named Eldad and the other named Modad, but the Spirit also rested on them. And the young man ran, and told Moses, and said, Eldad and Modad are prophesying in the camp: And Joshua the son of Nun, the servant of Moses, one of his chosen ones, answered and said, My lord Moses. But Moses said unto him, Hast thou not jealous of me? O that all the people of the LORD should be prophets, if the LORD would send His Spirit upon them”. (Numbers 11:24-29).

It is good that everybody can well write and speak.

First idea

Imagine that you can understand the information in the text regardless of the order of the words and preserve only the most necessary grammar (for example, the rule does not separate the preposition and the next word is preserved). Understand just like reading a text (after some learning and training, perhaps), in which only the first and last letters are preserved in words in their positions, and all the others are mutually mixed with each other. So imagine that you can understand (and express your thoughts, feelings, motives, etc.) the message of the text without adherence to strict word order. In this case, you can organize the words (keeping the most necessary grammar to reduce or eliminate possible ambiguity due to grammar, or rather a decrease in its volume), placing them so that they provide a more interesting phonetic sounding. You can try to create poetic (or at least a little more rhythmic and expressive) text or music.

It can also be an inspiring developmental exercise in itself. But how could you quickly find out which combinations are more or less fit? Also, can the complexity of the algorithms be reduced?

These are just some of the interesting questions. This work does not currently provide a complete answer to them, but is experimental one and a research, and any result of it is valuable.

Ukrainian is a language without strict requirements for word order in a sentence (although there are some established preferred options) and has a pleasant sounding. So, it can be a good example and instance. In addition, it is a native language for the author of the programs. Even if you don't want to create and use “phonetic” languages where phonetics is more important than grammar, then you can assess the phonetic potential of the words used in the text to produce specially sounded texts. It can also be valuable and helpful in writing poetry and possible other related fields.[49]

Sound Representations Durations as the Basis of the Approach

There is the fact as the basis of the approach that the language sounds have different durations, which depends on the many factors e. g. mean of the phoneme producing (the different one for every one of them), other factors that can be more or less controlled but usually the full control is not required and is not achieved. This leads to the fact that chaining of the phonemes and phonetic phenomenae sequentially among which there is also their syllables groupping introduces some rhythmic painting (picture, scheme). A human can (that is also trainable and can be developed) recognize the traits of the picture, compare them one with another, come to some phonetic-rhythmic generalizations and conclusions.

The question of determining the duration of speech sounds is not easy, but the exact result as already mentioned is not required. In this implementation of the approach of phonetic languages, certain statistical characteristics of sounds are used, in particular, possible durations are determined. If we compare the method of determining durations, which is proposed and used in the program of the r-glpk-phonetic-languages-ukrainian-durations package, the analogy will be the packaging of bulky objects. For the observer, the packaging will be an imaginary model of the process of obtaining sound durations. The pldUkr program (its generalization pldPL from the phonetic-languages-phonetics-basics package also follows this path, but it does not have normalization, because for different languages there may not be such a phenomenon as palatalization) uses linear programming to find the minimum convex hull (not in a strict mathematical sense), which can 'contain' the sounds of speech. This convex hull has an analogy of packaging, while the sounds of speech have an analogy of objects of variable volume inside the package. The same sound can be used in different situations, in different words with different durations, but the program tries to choose such durations that would 'cover' (similar to the envelope curve 'covers' a family of curves) all these variations for all sounds, with an additional normalization of the duration of the phonetic phenomenon of palatalization (softening) of the consonant, which is least controlled by man, and therefore it is expected that this duration is the most resistant to possible random or systematic fluctuations. For the Ukrainian language the possible duration which does not change strongly sounding is defined experimentally with use of the computer program mm1.

Finally, normalization is not mandatory, it is important that all durations are proportional to each other, i.e. it is not the durations themselves (which are numerically expressed as real positive numbers) that are important, but their mutual ratios (it is allowed to multiply these durations simultaneously by one and the same positive number that does not affect the results of the approach).

Polyrhythm as a Multi-Ordered Sequence Pattern

Let us have some sequence organized in the following way. Let us implement (generally speaking a conditional one) division of the sequence into compact single-connected subgroups with the same number of elements each in the subgroup, which actually means that we split the sequence into a sequence of subsequences with the same number of elements in each. Consider the internal ordering of each subsequence from the perspective of the placement of the values of its elements and repeatability of the some patterns of the placement of the elements. We assume that the elements can be compared in relation of order, that is, they are the elements of the data type that has an implemented instance of the class Ord.

Considering that the elements of the subsequences may be pairwise different (or in some cases equal), we will compare the positions on which the subgroups of elements that have a higher degree of relatedness ("closeness", "similarity") in value and order are located. Denote such subgroups by indices that have in the module code mostly a letter designation.

Then each subsequence will consist of the same number of elements of one nature (in particular, numbers of the type Double), in each subsequence there will be selected several subgroups of "similar" elements in value (and order, if the subsequences are sorted by the value), each of which will have its own index as a symbol (most often in the code – the characters). Subgroups must have (actually approximately) the same number of elements (in the code it is not strictly used for simplification of the former one, but it is so in the vast majority of cases because of the excessive "accuracy" of numbers of type Double that are used). Consider the question of positions in the subsequences of the corresponding subgroups in case of they have been belonging to different subsequences.

To assess this, we introduce certain numerical functions that have regular behavior and allow us to determine whether the subsequences actually have elements that belong to the relevant corresponding subgroups in the same places, or on different ones. It can be shown that the situation ”on different ones” corresponds to the presence of several rhythmimc patterns - for each subgroup will be their own, which do not mutually match, at the same time the ideal situation ”completely in the same places” corresponds to the case when these rhythms are consistent with each other, as is the case of coherence in quantum physics, in particular spatial and temporal coherence, which is important in particular for understanding of lasers and masers. Polyrhythms consisting of such rhythms, which cohere with each other, form a more noticeable overall rhythm, as well as the presence of coherence in the radiation leads to a more structured latter one (Zhabenko, n.d.b).

As an illustration for the ideas of the section the following data.

An example of the rhythmic sequence (almost ideal case).

```
Prelude Phladiprelio.Rhythmicity.PolyRhythm Numeric> let f x = putStrLn . showFFloat (Just 4)
(sin (2*pi*x)) $ ”” in mapM_ f [0,0.2..4]
0.0000
0.9511
0.5878
-0.5878
-0.9511
-0.0000
0.9511
0.5878
-0.5878
-0.9511
-0.0000
0.9511
0.5878
-0.5878
-0.9511
-0.0000
0.9511
0.5878
-0.5878
-0.9511
-0.0000

Prelude Phladiprelio.Rhythmicity.PolyRhythm Numeric> getPolyChRhData ’a’ 5 (PolyCh
[True,True,True,False] 5) (PolyRhythm [1,1,1,1,1]) . map (sin . (*pi) . (*2)) $
[0,0.2..4]
```

```
[[RP P c,RP P a,RP P b,RP P e,RP P d],[RP P c,RP P a,RP P b,RP P e,RP P d],
[RP P c,RP P a, RP P b,RP P e,RP P d],[RP P c,RP P a,RP P b,RP P e,RP P d]]
```

Here is the example of the sequence with not stable rhythm or with the rhythm that is less evident.

```
Prelude Phladiprelio.Rhythmicity.PolyRhythm Numeric> let f x = putStrLn .
showFFloat (Just 4) (sin (27182.81828459045*pi*x)) $ "" in mapM_ f [0,0.01..0.24]
0.0000
-0.5139
-0.8817
-0.9988
-0.8319
-0.4284
0.0969
0.5947
0.9233
0.9894
0.7742
0.3388
-0.1930
-0.6698
-0.9562
-0.9707
-0.7092
-0.2460
0.2872
0.7386
0.9801
0.9428
0.6375
0.1509
-0.3787

Prelude Phladiprelio.Rhythmicity.PolyRhythm Numeric> getPolyChRhData 'a' 5 (PolyCh
[True,True,True,False] 5) (PolyRhythm [1,1,1,1,1]) . map (sin . (*27182.81828459045) .
 (*pi)) $  [0,0.01..0.24]
[[RP P a,RP P b,RP P e,RP P d,RP P c],[RP P d,RP P e,RP P c,RP P b,RP P a],[RP P a,
```

```
RP P b, RP P c,RP P e,RP P d],[RP P d,RP P e,RP P c,RP P b,RP P a],[RP P a,RP P b,
RP P c,RP P e,RP P d]]
```

Coherent States of Polyrhythmicity as One of the Essential Sources of Rhythmicity

The described pattern of rhythmicity is one of the significant possible options for the formation of rhythmicity in particular in lyrics or music, but not the only one. It should be noted that the described mechanism of rhythm formation, as is noticed in the statistical experiments with texts using this code (the code of the library and its dependent packages on the Hackage site) may not be the only possible option, but in many cases it is crucial and influences the course of the rhythmization process (formation, change or disappearance of the rhythm). It is also known that the presence of the statistical relationship does not mean the existence of deeper connections between phenomena, in particular – the causality. "Correlation does not mean causality." A deeper connection implies the presence of other than the statistical ones to confirm it.

Rap Music Consequences

The code of the library allows in practice to obtain rhythmic patterns that are often close to the lyrics in rap style. Therefore, this can be attributed to one of the direct applications of the library.[51]

A Child Learns to Read, or Somebody New to the Language

When a child just begins to read words in the language (or, there can be just somebody new to the language) he or she starts with phonemes pronunciation for every meaningful written (and, hence, read) symbol. Afterwards, after some practice, he / she starts to read smoothly. Nevertheless, if the text is actually a poetic piece, e. g. some poem, it is OFTEN (may be, usually, or sometimes, or occasionally, etc.) just evident that the text being read in such a manner has some rhythmicity properties, despite the fact that the phonemes are read and pronounced in a manner of irregular and to some extent irrelevant to the normal speech mode lengths (durations). We can distinguish (often) the poetic text from the non-poetic one just by some arrangement of the elements.

The same situation occurs when a person with an accent (probably, strong, or rather uncommon) reads a poetic text. Or in other situations. The library design works just as in these situations. It assumes predefined durations, but having several reasonable (sensible) ones we can evaluate (approximately, of course) the rhythmicity properties and some other ones, just as the algorithms provided here.

This, to the mind of the author, is a ground for using the library and its functionality in such cases.[52]

Problem of choosing the best function and related issues

Consider the following question: suppose we have obtained the best version (in our subjective opinion or on the basis of some criteria, it is irrelevant here) of the line in one way or another (here the method does not really matter). Is there a function that makes this particular variant of the string optimal, i. e. for which such a variant of the string gives the maximum of all possible permutations? Yes, there is. This is easy to prove. The proof is reminiscent of the principle of equalizers.

Let $n \in N$ be the number of syllables in such a line. Arrange the durations of the syllables in ascending order (standard procedure for descriptive statistics). We will find the smallest nonzero difference between adjacent values, divide it by 5. Denote this value by δ. Now consider a number of syllable durations for our best string. Number each syllable from the beginning, counting from 1. Denote by $Y = \{y_i, \quad i \in N, \quad i = 1, 2, \ldots, n\}$ the set of all values of durations in the order of sequencing in the best line. Denote by $X = \{0 = x_1, x_2, \dots, x_i, x_{i+1}, \dots, x_{n+1}, \quad i \in N, \quad i = 1, 2, \dots, n\}$ the set of coordinates of the points of the ends of time conventional intervals, into which our best line divides the time line (the left edge is 0, because the countdown starts with 0). Denote by $M = \{z_i = \dfrac{x_i + x_{i+1}}{2}, \quad i \in N, \quad i = 1, 2, \dots, n-1\}$ a set of midpoints of the segments into which the time line is divided by the conditional intervals ends. Denote by $L_1[a, b]$ the class of Lebesgue-integrable functions on the interval $[a, b]$. Let's mark $I(y_i, z_i, \delta)[z_i - \delta, z_i + \delta]$ class of all bounded functions with $L_1[z_i - \delta, z_i + \delta]$, the maximum and minimum values of each of which lie on the segment $[y_i - \delta, y_i + \delta]$. We denote each function of class I by g.

Consider the class of functions F (a kind of finite analogues of the known delta functions of Dirac), defined as follows:

$$f(x, i) = \begin{cases} g \in I(y_i, z_i, \delta)[z_i - \delta, z_i + \delta], & \text{if } y_i \text{ is a unique value in the set } Y,\ x \in [z_i - \delta, z_i + \delta] \\ y_i, & \text{if } y_i \text{ has equal value with some other number from the set } Y,\ x \in [z_i - \delta, z_i + \delta] \\ 0, & \text{otherwise} \end{cases}$$

It is easy to see that

$$\sum_{i=1}^{n} \int_{-\infty}^{\infty} f(x, i)\, \mathrm{d}x,$$

where integration is carried out according to Lebesgue, is the desired function (because only the syllables of the best string in their places, taken into account with their indices, give a positive contribution to its value, and for all other variants of the function at least some of the syllables give 0 contribution), and it is not unique due to the fact that in the first line of definition $f(x, i)$ the function can have an arbitrary value from a closed non-empty interval. Therefore, there is at least one class of functions that is described by such a formula, for each of which this particular variant of the string will be optimal.

Let's ask the following question: if we consider not a line, but their combination, for example, a poem. Does a function that makes each line optimal (i. e. which will describe the whole work, each line in it) exist for the whole work (for this whole set of lines)?

In this case, the previous method of constructing the function does not give the desired result, because for two lines it may be that what is best for one of them is not the best for the other. In the case of increasing the number of lines, this general suboptimality only intensifies. However, the existence of such a function for different particular cases is a fundamentally possible situation, however, the probability of its existence should decrease both with increasing number of rows and with the appearance of different features of rows, which increase the differences between them. In general, the search for just one such function may be virtually impractical.

Nevertheless, if we consider the whole work as one line, considering it optimal, then the method just described to construct the corresponding function (class of functions) again gives the result. Yes, the number of syllables in it (in the generalized "line"-work) increases, but the procedure itself gives similar results (if we neglect the possible identical durations and repetitions of syllables in a larger text, which lead to the fact that some words can be rearranged and that makes the text only approximately optimal). It is possible to suggest further improvement of this procedure (for example, introduction of the factors which depend on values of the next durations of syllables) that allows to reduce suboptimality of the text.

But still, if you go from one text to another, the resulting function is unlikely to be optimal.

We conclude that for the whole set of expediently organized texts the existence of a universal function is seen as a certain hypothesis (with almost certainly proposes a negative answer).

Connection with fractals

We will pay attention to this fundamental ”break”, the fundamental dissimilarity of the texts.

But first note the following. The function considered above makes one line variant optimal at the same time you can consider not one option, but several, from which to choose the best. Then the task is reduced to search optimal group of options with the minimum possible number of components.

If we consider again the boundary case of a group with the number of elements equal to the factorial of the number of words in a line, it is easy to see for combinatorial reasons that such a group contains (consists of, is) all variants of a string of these words. But for 5 words such a group contains $5! = 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 = 120$ options for 5 words, a total of 600 words to read which usually takes a few minutes.

If you reduce the number of words in the optimal group, then among them may not be the right option, but the reading time decreases accordingly. So, if we have some way of at least approximate ordering, it will be expedient to find the optimal ratio ”group size - the degree of accuracy of approximate ordering”.

This leads to the idea of using different functions that are easy to calculate, have a certain characteristic behavior and allow you to roughly organize the sets of all permutations according to the search task, and the search is no longer a single option, but the optimal group of options.

At the same time, the search stages are fundamental, when the program does not give the desired result completely. Then you can change something in the data (without changing the words themselves), or change the words to change the options that the program works with, because it changes the structure of the set of all options and allows you to get other still somewhat ”optimal” options from the author’s point of view and in terms of the program. This gave impetus to the so-called recursive mode of operation, when the change of data is the merging of words (and hence the exclusion from consideration of options where they are not in a row), as well as the mode of several options.

An interesting observation is also that in recursive mode you can get lines in which more noticeable are the ”breaks” of rhythm, which can often be included in the rhythm through the introduction of additional pauses. Poems with pauses (for example, so-called caesuras) are also known in theory, for which this may be useful.

Also the ability to form lines from lines, and each line from certain components, resembles fractals. However, the relationship with fractals requires more detailed and in-depth study.

An interested reader can turn to the literature.[6, 31, 29, 25, 46, 17, 7, 35, 47, 10, 2, 28, 43, 37, 1, 5, 27, 40, 22, 3, 58, 63]

Analogy with the extremity principle

In mechanics and optics, the principle of extremity of action is known (a physical quantity of action is introduced, which for real processes in these areas takes among all possible values along the trajectories of the minimum (most often), or maximum value, or the constant one with respect to the trajectory, ie in one word the extreme value). In thermodynamics there is a law of increasing entropy of closed systems. In the case of mechanics and optics, the search for real trajectories is reduced to finding those of the possible trajectories for which the value of the action functional is extreme, which allows you to use the apparatus of higher mathematics to find these extreme values.[45]

It was suggested that the analogy with this principle, that is, the comparative mode of operation in combination with the mode of several properties, can give the best result. However, a wide range of good rhyming lines (and not only them) do not necessarily follow this rhyming pattern. Therefore, in general, the principle serves as a kind of search intuition, but not as a regularity.

Ability to use your own durations of representations of sounds or phonetic phenomena

The programs offer four different sets of phonetic representations by default but starting with version 0.13.0.0 it is possible to set your own durations. To do this, specify them as numbers of type 'Double' in the file in the order defined as follows:

UZ	'A'	D	дз	(твердий)	8
UZ	'A'	K	дз	(м'який)	9
UZ	'B'	D	ж	(твердий)	10
UZ	'B'	K	ж	(пом'якшений)	11
UZ	'C'	S	й		27
UZ	'D'	N	сь		54
UZ	'E'	L	ч	(твердий)	39
UZ	'E'	M	ч	(пом'якшений)	40
UZ	'F'	L	ш	(твердий)	41
UZ	'F'	M	ш	(пом'якшений)	42
	G				55
	H		ю		56
	I		я		57
	J		є		58
	K		ї		59
	L		'		60
	M		'		61
	N		нт		62
	O		ст		63
	P		ть		64
	Q		дзь		12
	R		зь		13
	S		нь		65
	T		дь		14
UZ	'a'	W	а		1
UZ	'b'	D	б	(твердий)	15
UZ	'x'	L	х	(твердий)	52
UZ	'x'	M	х	(пом'якшений)	53
UZ	'y'	W	и		5
UZ	'z'	D	з	(твердий)	25
UZ	'z'	K	з	(м'який)	26

UZ	'b'	K	б	(пом'якшений)	16
UZ	'c'	D	ц	(твердий)	38
UZ	'd'	D	д	(твердий)	17
UZ	'd'	K	д	(м'який)	18
UZ	'e'	W	е		2
UZ	'f'	L	ф	(твердий)	43
UZ	'f'	M	ф	(пом'якшений)	44
UZ	'g'	D	ґ	(твердий)	19
UZ	'g'	K	ґ	(пом'якшений)	20
UZ	'h'	D	г	(твердий)	21
UZ	'h'	K	г	(пом'якшений)	22
UZ	'i'	W	і		6
UZ	'j'	D	дж	(твердий)	23
UZ	'j'	K	дж	(м'який)	24
UZ	'k'	L	к	(твердий)	45
UZ	'k'	M	к	(пом'якшений)	46
UZ	'l'	S	л	(твердий)	28
UZ	'l'	O	л	(м'який)	29
UZ	'm'	S	м	(твердий)	30
UZ	'm'	O	м	(пом'якшений)	31
UZ	'n'	S	н	(твердий)	32
UZ	'n'	O	н	(м'який)	33
UZ	'o'	W	о		3
UZ	'p'	L	п	(твердий)	47
UZ	'p'	M	п	(пом'якшений)	48
UZ	'q'	E	ь		7
UZ	'r'	S	р	(твердий)	34
UZ	'r'	O	р	(м'який)	35
UZ	's'	L	с	(твердий)	49
UZ	't'	L	т	(твердий)	50
UZ	't'	M	т	(м'який)	51
UZ	'u'	W	у		4
UZ	'v'	S	в	(твердий)	36
UZ	'v'	O	в	(пом'якшений)	37
UZ	'w'	N	ць		66

where the specified values in the list refer to the phonetic representations of the data type UZPP2 (from the module Phladiprelio.Ukrainian.Syllable). The last column is 8-bit integers (GHC.Int.Int8), which represent these sounds in the new modules.

If you want to specify several such sets (up to 9 inclusive), you can specify '*' or several such characters from a new line, and then from the next line there will be a new set of values.

Each set should be in the following order: [1,2,3,4,5,6,7,8,9,10,11,15,16,17,18,19,20,21,22,23,24,25,26, 27,28,29,30,31, 32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,

where the number corresponds to the last column in the above diagram. 101 (prior to the version 0.20.0.0 here there was -1 instead, and it was at the beginning of the list, not at the end) corresponds to a pause between words (does not affect the search results of the line). Every new value must be written in the file from the new line.

Then when executing the program somewhere among the command line arguments (it does not matter where exactly) specify «+d» <path to the file with the specified data>. Programs will read these values and convert them to the appropriate values. As properties it is necessary to use then those which begin with the letter «H», and further the corresponding designation of property merged with it. For example, «Hw04», the last digit in the record in this case will mean the ordinal number of the set of values, starting from 1 (maximum 9).

Along with the custom values, you can use the provided by the library ones, as usual, in the mode of several properties.

Minimum grammar for possible preservation of meaning and intelligibility

Programs use permutations of words that neglect any (or at least part of) grammatical connections, word order, and so on. This can lead (in addition to the need to think) to situations where grammatically related language constructions are broken, their parts are transferred to other places, forming new connections and changing the meaning of the text.

To reduce this, to eliminate some of these effects, programs use concatenation of words that have a close grammatical connection, so as not to break them in the analysis. This allows you to maintain greater semantic ease and recognizability of the text, as well as a side effect to increase the overall length of the line, which can be analyzed. In the Ukrainian language, grammatically related auxiliary or dependent words precede the independent or main one, so the concatenation of these auxiliary or dependent words to the next one is used. The completeness of the definition of such cases is not exhaustive, but the most frequent cases are considered.

To reduce this, to eliminate some of these effects, programs use concatenation of words that have a close grammatical connection, so as not to break them in the analysis. For the general case, it should be borne in mind that auxiliar or dependent words can go after the independent or the main, so this should be considered separately (and attach such words to the previous, not to the next). Currently, the generalized version of phladiprelio-general-simple supports both.

Reduced set of permutations as a variant of the universal one

By default, the program analyzes the universal set of permutations of all words and their concatenations, while the number of analyzed options increases as a factorial of the number of such words or combinations. The text, organized more or less coherently in relation to one or another property, can be radically different from the original, which complicates understanding and has the effect of delaying calculations.

To quickly test the possible improvement of the text using the approach introduced sets of permutations of only one word relative to the text, as well as two words or their combinations as universal sets (in addition to the full one). When running programs, they are specified by the command line parameter «+p» somewhere among the arguments (the position does not matter). In this case, a reduced set of permutations is used. The number of words and their combinations that can be considered by the program as one line for analysis increases in this case to 10. To use the minimum possible set of permutations, you must specify as the next command line argument «1» (only one word), for pairwise permutations – «2» (pairwise permutations). Then among the arguments of the command line will be the expression «+p 1» in the first case, «+p 2» – in the second, and for the full set of permutations, you can still not specify this group (the full set is thus used for default).

In this case, the analysis is much faster (because the number of cases is significantly reduced), and the text changes less, which allows you to keep it more recognizable. Recursive sequential application of this case is possible, but it should be borne in mind that in the case of pairwise permutations this may not be the best option in terms of approach, and it may take even longer than analyzing the whole full set of permutations (because some options will be analyzed for several times, which does not happen in the first case). In the case of permutations of only 1 word relative to the text, it seems that if we recursively sequentially apply the search of the maximum element twice in a row for the same properties and we get the same option, then the chances that it will be maximum for the full set of permutations increase compared to pairwise permutations, but to estimate the increase (if this hypothesis is really true) is difficult in the general case. However, in practice, the search for such a 'minimally improved' option is promising, as it may well retain meaning and in advance – may make rhythmic structure better.

The following should be taken into account: when searching for the maximum element by the value of the property (i.e. without changing the structure) if the analysis of reduced permutation set received a text that coincides with the original, then there is a good chance that this option is optimal in terms of the property in question (although this is not guaranteed). And one more thing: in this case, a local maximum is reached (which may or may not be the global one). If the repeated application leads to the formation of another (already the third) option, then the previous local maximum was not exactly global and the program is moving in its direction.

See also: [60, 55, 59, 57, 62, 53, 56, 61, 54]

More about rhythm in music and language

Musical rhythm, meter, pulse and other similar topics are discussed in [11].

Pulse means one of a series of regularly repeating the same stimuli (sounds). The meter is a measure of the number of pulses between more or less regular accents. Accents are those pulses that for a person seem to stand out among others. Rhythm (musical) there means how one or more unaccented bits (pulses in a metric context) are grouped around one accented one. Accent can occur in different ways and does not coincide with the amplification of sound, and can be caused by a variety of factors in music.

It is noted that there are no strict and simple rules for determining rhythmic groups, but there are certain trends (at least in Western music – Author's note) that need to be considered. Sometimes the grouping can be ambiguous (multi-valued), multivariate, and allow different variants to coexist.

Grouping is architectonic, ie has different levels that interconnected. At different levels, grouping is the product of the similarities and differences of sounds, as well as the closeness and separation that are felt by the senses and organized mentally.

I will note here that the programs consider structures with the same number of syllables in one group, but several elements in one group can be selected, almost all. Moreover, the degree of ability to be remarkable is considered, which is considered as a measure of the importance of certain syllables in calculating the values of properties. These groups of syllables can be correlated with the meter, while different numbers of significant syllables correspond to the fact that the meter can exist regardless of rhythm. In the classical approach considered, a group with one accented sound is rhythmic. In the programs, there is no clear relationship between the accentuated composition and the degree of importance. This means that a different approach to this issue is proposed, the results of which can be fruitful.

However, it can often be assumed that a group of significant syllables corresponds to a rhythmic group at one of the architectectonic levels, and then the structures with the maximum value of the corresponding properties of this type will correspond to lines with more regular repetition of rhythmic groups in their general features, on this level of composition of the work, and smaller values of the corresponding properties of this type will correspond to unevenly distributed rhythmic groups, or their variations, i.e. more complex (and therefore perhaps less recognizable and tangible, less significant) rhythmic pattern at a higher architectonic level. It can also be tentatively assumed that for the maximum values of the corresponding properties of the strings rhythmic grouping is associated with metric, but if we do not consider the maximum elements, the degree of connection between rhythmic grouping and metric weakens, i.e. such strings tend (in the musical sense) to ”free” rhythm of Oriental and folk music and ”measured rhythm” of Gregorian chants [11]. In fact, it offers a wide range of opportunities to explore the relationship between metric and rhythmic structure in language.

Differences between rhythmicity in music and language and their cultural conditionality

Instead, [23] shows that instead of grouping based on perceptual principles (louder sound starts the group more often, and lengthened sound or interval ends the group more often), different groupings are possible for music in the English and Japan cultures, as well as patterns in corresponding languages. In the same paper, there are many references to works that have previously considered one or another approach. Also listed differences between English (also Ukrainian is closer to English in this context) and Japanese, in particular the official parts of the language precede the main words in English (and Ukrainian), while follow the main ones in Japanese.

In [19] the differences between languages in syllable durations are mentioned: languages can be divided into those in which the structure of syllables is more or less variable. This affects in particular the unequal duration of sounds in different positions, the phenomena of reduction of vowels in languages where the structure of syllables is more variable. In [4] it is also noted that, for example, in French the phenomenon of lengthening the duration of final syllables is very noticeable, especially at the end of a phrase or sentence. Programs do not take into account such differences in the duration of syllables, but their consideration may be a task for further improvement.

Differences in rhythmicity in music and language are discussed in [26]. It uses an interdisciplinary approach. In particular, the analysis of brain activity in reactions to incongruence (deviations from the structure of rhythm) in language shows that they significantly influence the process of understanding the meaning of language (prolong reaction and comprehension time). Similar results on the influence of rhythm on the ability to understand and speak are given in [30].

In [24] the peculiarities of the African-American rhythmic system in music are considered, in particular, such features as more saturated rhythmic development of smaller parts of works and less variability between smaller parts of one larger work, as well as the idea of expressive rhythm, in which, in contrast to the uniform beat of Western music, an approximate equivalent of a beat and a much shorter ”atomic” rhythm are introduced, which, presumably (conditionally), consists of notes of about 1/16 – 1/24 durations and introduces ”expressive details”. The phenomenon of superposition is also noted in combining of different cyclic rhythms into one common (well traced, for example, in the Afro-Cuban rumba), which is similar to the idea of coherence of polyrhythms to form a single rhythm. The phenomenon of expressive ”atomic” rhythm, its use and characteristics show that it makes sense to enter the duration of phonetic phenomena with high (at first glance excessive) accuracy.

In [48] it is shown that children aged 5-24 months are more responsive to musical rhythms than speech, also that they mostly evoke positive emotions in them, as well as motor activity.

In the work [42] it is considered how rhythms in music can cause emotional states or affect them. It is noted that this is not an automatic process, and it is influenced by individual tastes, familiarity and training.

In the work [8] an attempt is made to investigate the influence of melodic and rhythmic changes on the perception of melody. It was found that melodies unfold more slowly (in perception), if they have more changes in pitch, more incompatible changes in rhythmic structure.

In [13] it is proposed an approach to determining the beginning in a musical sound signal based on the analysis of rhythm as well.

In [44] it is studied how brain processes musical information in case of experienced jazz musicians compared to unqualified ones. It is shown that there are noticeable differences in the ability to predict rhythm. This indicates the possibility of training the sense of rhythm.

In [9], it has been experimentally shown that pre-listening to music with a similar rhythm improves the ability to detect phonemes in speech, and this effect is enhanced by pre-audio-motor preparation for listening to speech sentences. This can be used, for example, in speech therapy. There is also an assumption about the processing of linguistic and musical temporal (rhythmic) structures by the brain with the participation of common (same) resources.

In [14] it is investigated the relationship between the temporal and rhythmic structure of music at the note level, and it is proposed a method that significantly takes into account the rhythm to determine the oneshot of the note in the sound signal.

In [39] it is experimentally shown that if it is a meaningful phrase, phonemes in stressed syllables are perceived faster, and if the phrase consists of nonsense words (without lexical meaning), then there is no significant difference in reaction time for stressed and unstressed syllables. It is concluded that stressed syllables can be predicted (variation of apperception), while unstressed – no (this explains the difference in reaction time). This result suggests that the presence of dynamic accents improves the speed of comprehension of the text (for those who are native speakers of a culture with a language that has dynamic verbal accents).

The paper [32] draws some conclusions about the connection between rhythmic structure in language and music, in particular the obvious fact that there is much in common in rhythmic groupings in language and music, but this does not make them identical. For research, it is further proposed to consider the influence of linguistic rhythmic structure on music. It is mentioned that the available data partially confirm that the musical rhythmic structure is influenced by language. Linguistic (linguistic) rhythm is a combination of several factors that affect the temporal organization of speech. First, it is the alternation of words and pauses, secondly, the different durations of syllables, and thirdly, the alternation of stressed and unstressed syllables. These factors can cause languages to be rhythmically similar or different. Musical rhythmic grouping also means grouping into a phrase, whether there is a beat (a periodic one), as well as a metric structure. What is common is the fact of grouping into phrases, and the differences are mainly in periodicity. In particular, the authors emphasize that the early and influential hypothesis about the division of languages into those in which accents are approximately evenly distributed and those in which syllable durations are approximately the same, i.e. the beginnings of syllables are approximately evenly distributed, is not confirmed by available experimental data. This creates real prospects for using programs for many languages. However, linguists have retained these concepts and continue to describe differences in languages. There are discussions about whether these are really important concepts, or just two ends of a continuum in which all languages have their place. Also important is the fact that in normal conditions there is no periodicity in the speech rhythm, in contrast to the common in music. The authors also express their belief that the influence of language rhythms on music is not universal, but more characteristic of periods and events when composers try to emphasize their national identity and affiliation.

Advice how to use the programs

In correspondence with the previous information, the rhythmicity of the proposed by the programs variants can be in many cases more evident and perceived better if you read the words (their concatenations) at the lines without significant pauses between them (as one single phonetic flow), not trying to strengthen emphases (probably even without well articulated emphases, smoothly, with liasing). This can be even in a different way, but if the obtained variants do not seem to be rhythmic enough then try just this option and since then compare and come to conclusion whether such sounding is just suitable for your situation.

Some additional information

The work [38] proposes a method for classifying music that uses hidden Markov chains, as well as information on rhythmic structure, comparable in accuracy with manual classification.

The work [20] proposes a relatively effective way to determine the rhythmic similarity of music fragments.

In [15] it was studied whether pianists-performers distinguish different features of rhythmic structure, meter and melody by systematic variations of performance. The most characteristic were the allocation of rhythmic grouping.

In [41] the authors try to use wavelet analysis to obtain characteristics of rhythmic structures of musical pieces and speech, highlight their mathematical features, then apply the results to analyze pieces. It is shown that the wavelet characteristics have a perceptual basis. It is proposed to discuss the possibility of applying the obtained results to cascading rhythm generation.

The [36] paper considers a statistically common approach to determining the linguistic rhythmic structure based on vocal (vowels and their sequences) and intervocal (consonants and their sequences) intervals (distances, durations) and their variability. It is shown that this allows better typology of languages in terms of rhythmic structure. It is additionally considered to the already mentioned languages with approximately the same frequency of accent distribution, languages with approximately the same syllable duration, as well as languages with approximately the same distribution of moras (mora is a syllable with a short vowel or one short vowel). Variants of pairwise variability indices (rPVI, nPVI) and / or standard deviations are used for estimates. Much of the work is devoted to description points of view on the question of whether it is possible to cluster and typify languages in rhythm or that they all form a certain continuum of values, distributed within one large group. Research is ongoing in this area.

The work [16] shows that sonority can be used instead of durations to determine rhythm, which also makes it easier to automate the segmentation process into rhythmic groups. Later works continue to study this question, using, in particular, hidden Markov chains.

In the work [33] it is considered the influence of the peculiarities of the linguistic rhythmic structure on the musical one for the outstanding English and French composers of the past continues. There is a discussion on this issue.

The work [34] analyzes this issue for German and Austrian composers over a period of about 250 years, pointing out that there are fluctuations that can be explained by historical and cultural rather than linguistic influences.

In [12] the same theme continues in comparison with Italian composers.

Theories of melodic musical emphasis are considered in [21]. Experimental consideration confirms mainly the theory of Joseph Thomassen (1982). In this model (theory) the most emphasized are the turning points in the melody, when the change of pitch occurs in opposite directions, and this is more noticeable for ascending-descending rotation than for descending-ascending. But in different cases there may be a melodic accent of a different kind. It has been shown that melodic emphasis can be a rather weak factor in the rhythmic structure of music.

The work [19] analyzes whether listeners can determine the rhythms of music (the language they know well) in which language the song is. As it turned out, in many cases they can. Listeners can use the similarities and differences in the rhythmic structures of languages to determine which language the song was composed of.

An interesting additional information about the rhythm and related musical topics is in the video:

1. Adam Neely. Solving James Brown’s Rhythmic Puzzle. Adam Neely. 2021.

Analogy with the flow regime of liquid, plasma or gas

In the work [**?**], the elements of the energy theory of speech are described, according to which, ”a syllable is a minimal articulatory unit,... representing a single contraction of muscles and mouth placement ”consonant-vowel””.

Then the sequence of pronunciation of syllables forms a certain *flow of articulatory energy*, i.e. an integral sequence of speech acts, which can be compared with the flow of liquid, plasma or gas (further on, for possible greater brevity, we will write only about the flow of liquid, although while actually spoken about a wider concept).

Let’s make such a comparison in more detail.

Two main modes of flow (movement) are possible for a liquid flow — laminar and turbulent. A good visual explanation can be viewed, for example, in a video at the link: Understanding Laminar and Turbulent Flow - Efficient Engineer.

The work [**?**] gives the following definition: ”Turbulence is the phenomenon of the appearance of eddies of various sizes in the flows of liquids and gases. As a result of turbulence, the characteristics of these flows change uniformly in space and time. Turbulence occurs when the Reynolds number characterizing the flow exceeds some critical value”.

The work [18] states that the flow of liquids and gases can occur in both laminar and turbulent regimes.

Can you find something similar in speech? Are there any language ”eddies”?

If we believe that ”speech eddies” exist, at least in some cases, then we can talk about ”turbulence of the speech flow” in these cases.

But based on the peculiarities of sound formation during oral speech, it is difficult to establish the presence of certain ”speech eddies”. If they really do not exist, then it is appropriate to compare the speech flow with a laminar flow mode, or with a transitional state.

The video mentioned above mentions that there is also a transition state between laminar and turbulent fluid flow.

Is it possible to add something to the speech so that analogs of eddies are formed?

First, you can look for the sources of ”eddies” in music - because there are more repetitions in it than in speech, and they can be partially compared to eddies, in which the movement is partially repeated.

But for a turbulent flow, the so-called energy eddy cascade is characteristic, so for music and speech there is no way to form the smallest eddies that could dissipate energy through viscous friction. That is, in music and in speech we deal with quantized elements, for example, phonemes and speech acts, with sounds, and therefore the classical picture of turbulence cannot be obtained anyway.

Constraints

In the work [50], an interesting system of constraints was introduced, which allow considering only some subsets of all possible options, which are built taking into account the relative order of words and their place in a line.

When starting the phladiprelioUkr program, constraints can be specified as command line arguments. They allow you to reduce the number of calculations, consider only certain options (for example, with a certain defined order of some words, etc.), which allows you to actually expand the capabilities of the program.

There are two options for encoding information about constraints. They use the same basic constraints, but differ in that the extended version allows you to perform algebraic operations on Boolean sets, which allows you to get the complete result, whatever it is, in one input. Instead, the older variant is based on simple linear logic, which can also be sufficient in many cases - all used constraints in the base form must be fulfilled simultaneously, which is equivalent to having a relationship between them through the operator (&&) (boolean ”AND”) and the intersection of the corresponding Boolean sets.

Since words and their combinations in this implementation are assumed to be no more than 7 in a row, all numbers in the basic constraints must be no larger than 7 for the constraint to be ”non-zero”.

It is also worth saying that to get more or less interesting results, it is often advisable to reduce the number of restrictions and use less than necessary.

There are 15 types of basic constraints, and they can be combined in any way, but in compliance with the specifications for each of them. If the specifications are violated, the program will read this constraint as its absence (equal to ”E”).

The basic principle is that the digits indicate the ordinal number of a word or a combination of words written as one word in a string starting from 1 (a natural number), and the letter indicates the type of basic constraint. The rule has an exception – a ”zero” constraint (its absence, i.e. the entire set of all possible permutations is indicated), which has only a letter and no digits – ”E”.

Another necessary condition for the constraint to be non-zero is that no digital symbols within the same encoded constraint cannot be repeated twice. For example, the following constraints are known to be invalid constraints: Q2235 (repeated digits), E2 (numeric characters in a zero constraint where there are none), T2435 (8 is greater than 7). T248 (8 is greater than 7), F1 (one character instead of the required two), A3852 (8 is greater than 7), B5 (one character, but there should be more).

The types of restrictions and their meanings are described in more detail below.

- Constraint E – No additional numeric characters - Corresponds to the absence of an additional constraint. Denotes the entire set of all theoretically possible permutations. If used in the extended version (+b ... -b), it can be negated by the symbol ”-”, and then it corresponds together with this symbol to an empty set of permutations, i.e. no actual options are present.

- Constraint Q – 4 pairwise unequal digits in the range from 1 to the number of words or their concatenations. The digits are the numbers of 4 words or their concatenations, the mutual order of which will be preserved in the permutations.

Also, if these words are the same (without taking into account uppercase and lowercase letters), then this is a convenient way to reduce the amount of data to be analysed.

- Constraint T – 3 pairwise unequal digits between 1 and the number of words or their concatenations – The digits are the numbers of 3 words or their concatenations, the mutual order of which will be preserved in the permutations.

  Also, if these words are the same (without taking into account uppercase and lowercase letters), then this is a convenient way to reduce the amount of data that will be analysed.

- Constraint F – 2 unequal digits between 1 and the number of words or their concatenations – The digits are the numbers of 2 words or their concatenations, the mutual order of which will be preserved in the permutation.

  Also, if these words are the same (without taking into account uppercase and lowercase letters), then this is a convenient way to reduce the amount of data to be analysed.

- Constraint A – 1 digit and several more unequal digits (all unequal to each other) to its right in the range from 1 to the number of words or their concatenations - The first digit is the sequence number of the element, relative to which the placement of all other elements (words or their concatenations); all other digits to the right – the numbers of the elements that should appear in the resulting permutations TO THE RIGHT of the element with the number equal to the first digit.

- Constraint B – 1 digit and several more pairwise unequal digits to its right in the range from 1 to the number of words or their concatenations – The first digit is the sequence number of the element in relation to which the placement of all other elements (words or their combinations); all other digits to the right – the numbers of the elements to be placed in the resulting permutations TO THE LEFT of the item with the number equal to the first digit.

- Constraint P (fixed Point) – 1 or more several pairwise unequal digits in the range from 1 to number of words or their concatenations – each of them means the ordinal number of the word that will remain in its place during permutations place.

- Constraint V – (as half of W) – 2 unequal digits in the range from 1 to the number of words or their concatenations, each of which means the ordinal numbers of two words, the distance between which and the mutual order of which will be preserved in all permutations. Most often it can be used for two neighbouring words and means then, that they will be present as such a pair in all the variants being analysed. It is also important if you don't want to concatenate such words so as not to change the syllables of their common ("tangent") boundaries at the same time.

- Constraint W – (from the word tWo – 2) – 2 unequal digits in the range from 1 to the number of words or their concatenations, each of which means the ordinal numbers of the two words, the distance between which will be preserved in all permutations. Most often it can be used for two neighbouring words and means then, that they will be present in all variants together, but their relative order can be also reversed.

- Constraint H – (from the word tHree – 3) – 3 unequal digits in pairs in the range from 1 to the number of words or their concatenations, each of which means the ordinal numbers of the three words, the distances between which and the mutual order of which will be preserved in all permutations. A certain subset of the constraint R. Most often used for three words in a row.

- Constraint R – (from the word thRee – 3) – similar to H, but only the distances between the first two words are preserved, and the same for the distances between the second and third words. Their order can be changed to reverse. It's a rather complicated combination, but it can be important in terms of grammar, as well as the string to be analysed.

- Constraint M – (from the word Mixed) – mixed H-R constraint – similar to H, but between the first two words are preserved both the distance and their relative position, and only the distance is preserved between the second and third ones. It is also a rather complex combination, a certain subset of the constraint R.
- Constraint N — (fixed poiNts) — an even non-zero number of digits, where each digit in an odd position (first digit, third, fifth, etc.) means the ordinal number of the word or combination in the string that must be fixed in a new position, and the next digit is the new ordinal number of this word or combination; thus, if this pair of digits is the same, it means that this word should remain in its place, otherwise it will be fixed in a new position. Generalised constraint P, where the user can change the position of words and their combinations in the string and fix them.
- Constraint D — (DistanceD worDs) — three digits, the first two of which must be unequal and indicate the sequence numbers of two words or their combinations, which must preserve the mutual order, and the third digit is the new distance between these words (1 means that the words are consecutive, 2 means that there is one other word between the words, etc.), it must be in the range from 1 to the number of words and their combinations minus 1. A generalisation of constraint V, where the user can specify a new distance between words.
- Constraint I — (dIstanced words) — is similar to D, with the difference that the words can be in reverse order. Therefore, in the symbolic notation, Iabc == Ibac. Also, Dabc + Dbac == Iabc. A generalisation of the W constraint, where the user can specify their own distance between words.

Simple linear ”AND” logic

If only one of the basic constraints is used, or it is known for sure that it is necessary that all basic constraints in their set are fulfilled simultaneously, then you can use this option. It is logically the same as the one implemented previously.

To do this, you use +a ... -a ”brackets” between which there are constraints in their record, and all of them are separated from each other by a space. For example:

- +a P23 A456 -a – means that P23 and A456 are satisfied simultaneously, i.e. that the second and third words must remain in their places (constraint P), and that the fifth and sixth words must come after the fourth (constraint A).
- +a B23 F45 E T356 -a means that B23, F45, E and T356 are satisfied simultaneously, i.e. the third word must come after the second, the fifth after the fourth, a ”zero constraint” (meaning that it can be omitted), and the relative order of the third, fifth and sixth words must be preserved.

In fact, the space character between the constraints in this variant means a logical ”AND”.

If you need to set more complex constraints, or in general, in view of the possible faster work with the program, you should use the general algebraic version of setting constraints.

Algebraic (universal) variant of constraints

I would like to point out that in order to use this functionality efficiently, you need to know the basics of logic and/or set theory, but these materials are widely available in different languages, including Ukrainian and English, at different levels of complexity. As for mathematics, you should at least understand what logical AND (conjunction), logical OR (disjunction), logical negation (NOT), universal set, and complementary set are. If you know how to program in at least one language, you are also familiar with these concepts and have used them.

To use the algebraic version of constraints, you use +b ... -b ”brackets” between which the constraints in their record are placed, and you can additionally use brackets and the ”-” sign, which means a logical negation (logical ”NOT”) of the next after minus expression in brackets or the basic constraint. Parentheses, as is often the case, indicate the order of calculation, as in algebra. After the minus, there must be either an open parenthesis and an expression or an underlying constraint.

If the constraints are written consecutively without spaces, the program understands this as a ”multiplication” of the constraints, i.e. a logical ”AND”, and if there is a space between them, it is an ”addition” (logical ”OR”).

There can be no more than 1 ”-” in a row, and there can be only one space between groups of constraints. Parentheses must be opened and closed according to the general rules of algebra (for example, the number of open and closed parentheses must be equal to each other). If these rules are violated, the program will display a message that it is impossible to output options that meet these constraints and ask you to specify other data and constraints. A bracketed group must contain at least one basic constraint. The priority of operations is the same as for logical operators in Haskell and in Boolean logic in general.

Note: if you use Linux shells (e. g. bash) or Windows PowerShell then for the usage of parentheses you need to use quotation marks for the whole generalized algebraic constraint. Please, refer to the further documentation for your shell or PowerShell.

Examples of correct constraints:

- +b ’(A23-H345 P45-(A24 B56))’ -b is a symbolic representation of a Boolean set

  ```
  (A23 && not H345 || P45 && not (A24 || B56)),
  ```

  where ’not’ means a completion to the set in the universal set of permutations, i.e. here it means all those theoretically possible permutations, except for those denoted by the expression after ”-”. Mathematically, this can be written as:

  $$(A23 \wedge \neg H345 \vee P45 \wedge \neg(A24 \vee B56))$$

- +b A345B62 P4-Q3612 -b is a symbolic representation of a Boolean set

  ```
  A345 && B62 || P4 && not Q3612,
  ```

  which is mathematically equivalent to:

  $$A345 \wedge B62 \vee P4 \wedge \neg Q3612$$

- +b ’(E)-P4’ -b is a symbolic representation of a Boolean set

  ```
  (E) && not P4,
  ```

  which is mathematically equivalent:

  $$U \wedge \neg P4 = \neg P4,$$

  where U is the universal set of all theoretically possible permutations.

Examples of mis-specification of constraints – if in each of the previous examples above remove one character except for numbers and spaces, or write two consecutive minuses or spaces anywhere. Or open and close brackets without letters inside.

Previous implementation

The previous implementation is at the link: https://hackage.haskell.org/package/phonetic-languages-simplified-examples-array

Prerequisite for using the software package

At the moment, the programs work for workstations (desktops e. g.), and there are no mobile versions.

You must have Haskell applications installed and configured:

1. GHC (versions not earlier than 8.10)
2. Cabal

The executables of these programs must be searchable through the PATH workspace variable (this is the default setting).

If possible, use the system package manager (programs) to install also important packages Haskell bytestring, vector, heaps, parallel.

If the required Haskell packages are not installed using the system manager, they will be installed when installing the packages (downloaded and installed automatically, with additional time also spent on their compilation).

Installing the package

Open a command prompt or terminal and enter as commands:

```
cabal update
cabal --reinstall --force-reinstalls v1-install phladiprelio-ukrainian-simple
```

If you would like to use generalized variant, use:

```
cabal --reinstall --force-reinstalls v1-install phladiprelio-general-simple
```

Acknowledgements

The author dedicates the work on this project to the Gastrostars foundation and the founder of the foundation Emma Kok. As stated on the foundation site: ”The Gastrostars Foundation is a gastroparesis patient organization. This is a disorder of the stomach, in which its muscles do not work well or at all. This interferes with the normal emptying of the stomach and has a huge impact on the lives of patients. The Foundation is a place where you can communicate with each other, share stories and experiences, give each other tips and advice. It should be not only more understanding of this disease, but also more help in the fight against it” (translated by Google). If you wish to financially support the foundation, please contact the founder through the form on the website: https://gastrostars.nl/hou-mij-op-de-hoogte.

The author would like to thank the authors of the work Provably Correct, Asymptotically Efficient, Higher-Order Reverse-Mode Automatic Differentiation Faustyna Krawiec, Simon Peyton-Jones, Neel Krishnaswami, Tom Ellis, Richard A. Eisenberg та Andrew Fitzgibbon for the idea of the optimization and Mikolaj Konarski, who drew the author's attention to this article. The author also thanks his friends, whom he wanted to please with this research. Moreover, the author is acknowledged to everyone who sinserely and kindly has facilitated and promoted the research in any way especially to the Ukrianian soldiers who are on the front in Ukraine.

Bibliography

[1] Shlomo Alexander and Raymond Orbach. Density of states on fractals:«fractons». *Journal de Physique Lettres*, 43(17):625–631, 1982.

[2] Jan Andres, Jiří Fišer, Grzegorz Gabor, and Krzysztof Leśniak. Multivalued fractals. *Chaos, Solitons & Fractals*, 24(3):665–700, 2005.

[3] Jan Andres and Miposlav Rypka. Multivalued fractals and hyperfractals. *International Journal of Bifurcation and Chaos*, 22(01):1250009, 2012.

[4] Corine Astésano. *Rythme et accentuation en français: invariance et variabilité stylistique*. Editions L'Harmattan, 2001.

[5] David Avnir, Ofer Biham, Daniel Lidar, and Ofer Malcai. Is the geometry of nature fractal? *Science*, 279(5347):39–40, 1998.

[6] Michael F Barnsley, John E Hutchinson, and Örjan Stenflo. V-variable fractals: fractals with partial self similarity. *Advances in Mathematics*, 218(6):2051–2088, 2008.

[7] Christopher J Bishop and Yuval Peres. *Fractals in probability and analysis*, volume 162. Cambridge University Press, 2017.

[8] Marilyn G Boltz. Tempo discrimination of musical patterns: Effects due to pitch and rhythmic structure. *Perception & Psychophysics*, 60(8):1357–1373, 1998.

[9] Nia Cason, Corine Astésano, and Daniele Schön. Bridging music and speech rhythm: Rhythmic priming and audio–motor training affect speech perception. *Acta psychologica*, 155:43–50, 2015.

[10] Andrés E Coca, Gerard O Tost, and Zhao Liang. Controlling chaotic melodies. *Proc. Encuentro Nacional de Investigación en Posgrados (ENIP)*, 2009.

[11] Grosvenor W Cooper, Grosvenor Cooper, and Leonard B Meyer. *The rhythmic structure of music*. University of Chicago press, 1963.

[12] Joseph R Daniele and Aniruddh D Patel. An empirical study of historical patterns in musical rhythm: Analysis of german & italian classical music using the npvi equation. *Music Perception: An Interdisciplinary Journal*, 31(1):10–18, 2013.

[13] Norberto Degara, Matthew EP Davies, Antonio Pena, and Mark D Plumbley. Onset event decoding exploiting the rhythmic structure of polyphonic music. *IEEE Journal of Selected Topics in Signal Processing*, 5(6):1228–1239, 2011.

[14] Norberto Degara, Antonio Pena, Matthew EP Davies, and Mark D Plumbley. Note onset detection using rhythmic structure. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5526–5529. IEEE, 2010.

[15] Carolyn Drake and Caroline Palmer. Accent structures in music performance. *Music perception*, 10(3):343–378, 1993.

[16] Antonio Galves, Jesus Garcia, Denise Duarte, and Charlotte Galves. Sonority as a basis for rhythmic class discrimination. In *Speech Prosody 2002, International Conference*, 2002.

[17] Alex Goldsmith. Synthesising music: exploiting self-similarity using modular forms.

[18] Abhijit Guha. Transport and deposition of particles in turbulent and laminar flow. *Annu. Rev. Fluid Mech.*, 40:311–341, 2008.

[19] Erin E Hannon. Perceiving speech rhythm in music: Listeners classify instrumental songs according to language of origin. *Cognition*, 111(3):403–409, 2009.

[20] Andre Holzapfel and Yannis Stylianou. Rhythmic similarity of music based on dynamic periodicity warping. In *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2217–2220. IEEE, 2008.

[21] David Huron and Matthew Royal. What is melodic accent? converging evidence from musical practice. *Music Perception*, 13(4):489–516, 1996.

[22] John E Hutchinson. Fractals and self similarity. *Indiana University Mathematics Journal*, 30(5):713–747, 1981.

[23] John R Iversen, Aniruddh D Patel, and Kengo Ohgushi. Perception of rhythmic grouping depends on auditory experience. *The Journal of the Acoustical Society of America*, 124(4):2263–2271, 2008.

[24] Vijay Iyer, Jeff Bilmes, Matt Wright, and David Wessel. A novel representation for rhythmic structure. In *Proceedings of the 23rd International Computer Music Conference*, pages 97–100, 1997.

[25] Jun Kigami. *Analysis on fractals*. Number 143. Cambridge University Press, 2001.

[26] Cyrille Magne, Mitsuko Aramaki, Corine Astesano, Reyna Leigh Gordon, Sølvi Ystad, Snorre Farner, Richard Kronland-Martinet, and Mireille Besson. Comparison of rhythmic processing in language and music: An interdisciplinary approach. *Journal of Music and Meaning*, 3, 2005.

[27] Benoit B Mandelbrot and Benoit B Mandelbrot. *The fractal geometry of nature*, volume 1. WH freeman New York, 1982.

[28] Kevin Merges. Fractals and art. 2005.

[29] Ihor Nabytovych. ФРАКТАЛИ ТА ФРАКТАЛЬНІ СТРУКТУРИ У ХУДОЖНЬОМУ ТЕКСТІ (на прикладі прози Л. Керрола, КС Льюїса та ХЛ Борхеса). *ВІСНИК ЛЬВІВСЬКОГО УНІВЕРСИТЕТУ. Серія іноземні мови*, (18).

[30] Katie Overy. Dyslexia and music: From timing deficits to musical intervention. *Annals of the New York academy of sciences*, 999(1):497–505, 2003.

[31] Narayan Partap and Renu Chugh. Fixed point iterative techniques–an application to fractals. *International Journal of Research in Mathematics & Computation*, 4(1):1–7, 2016.

[32] Aniruddh D Patel. Rhythm in language and music: parallels and differences. *Annals of the New York Academy of Sciences*, 999(1):140–143, 2003.

[33] Aniruddh D Patel and Joseph R Daniele. An empirical comparison of rhythm in language and music. *Cognition*, 87(1):B35–B45, 2003.

[34] Aniruddh D Patel and Joseph R Daniele. Stress-timed vs. syllable-timed music? a comment on huron and ollen (2003). *Music Perception*, 21(2):273–276, 2003.

[35] Rohit Sunkam Ramanujam and Bill Lin. Randomized partially-minimal routing on three-dimensional mesh networks. *IEEE Computer Architecture Letters*, 7(2):37–40, 2008.

[36] Franck Ramus. Acoustic correlates of linguistic rhythm: Perspectives. In *Speech Prosody 2002, International Conference*, 2002.

[37] Dietmar Saupe. Algorithms for random fractals. In *The science of fractal images*, pages 71–136. Springer, 1988.

[38] Xi Shao, Changsheng Xu, and Mohan S Kankanhalli. Unsupervised classification of music genre using hidden markov model. In *2004 IEEE International Conference on Multimedia and Expo (ICME)(IEEE Cat. No. 04TH8763)*, volume 3, pages 2023–2026. IEEE, 2004.

[39] Joyce L Shields, Astrid McHugh, and James G Martin. Reaction time to phoneme targets as a function of rhythmic cues in continuous speech. *Journal of Experimental Psychology*, 102(2):250, 1974.

[40] Robert S Strichartz. Fractals in the large. *Canadian Journal of Mathematics*, 50(3):638–657, 1998.

[41] Neil P McAngus Todd. Segmentation and stress in the rhythmic structure of music and speech: A wavelet model. *The Journal of the Acoustical Society of America*, 93(4):2363–2363, 1993.

[42] Wiebke Trost and Patrik Vuilleumier. Rhythmic entrainment as a mechanism for emotion induction by music: a neurophysiological perspective. *The emotional power of music: Multidisciplinary perspectives on musical arousal, expression, and social control*, pages 213–225, 2013.

[43] Giuseppe Vitiello. The brain is like an orchestra. better yet, it is like a jazz combo, which doesn't need a conductor. *Chaos*, 11(1):2017, 2017.

[44] Peter Vuust, Leif Ostergaard, Karen Johanne Pallesen, Christopher Bailey, and Andreas Roepstorff. Predictive coding of music–brain responses to rhythmic incongruity. *cortex*, 45(1):80–92, 2009.

[45] Huang Wan-Li. The extremity laws of hydro-thermodynamics. *Applied Mathematics and Mechanics*, 4(4):501–510, 1983.

[46] Kerri Welch. *A fractal topology of time: Implications for consciousness and cosmology*. California Institute of Integral Studies, 2010.

[47] Wolfgang Wildgen. Chaos, fractals and dissipative structures in language. or the end of linguistic structuralism. *Gabriel Altmann und Walter A. Koch (Hg.), Systems. New Paradigms for the Human Sciences, de Gruyter, Berlin*, pages 596–620, 1998.

[48] Marcel Zentner and Tuomas Eerola. Rhythmic engagement with music in infancy. *Proceedings of the National Academy of Sciences*, 107(13):5768–5773, 2010.

[49] Oleksandr Zhabenko. dobutokO-poetry. https://hackage.haskell.org/package/dobutokO-poetry.

[50] Oleksandr Zhabenko. Phonetic languages approach to discovering preferred line options (phladiprelio) using Haskell.

[51] Oleksandr Zhabenko. phonetic-languages-rhythmicity. https://hackage.haskell.org/package/phonetic-languages-rhythmicity.

[52] Oleksandr Zhabenko. phonetic-languages-simplified-generalized-examples-array. https://hackage.haskell.org/package/phonetic-languages-simplified-generalized-examples-array.

[53] П Івончак. Український силабо-тонічний вірш 50-х років XIX століття. *Науковий вісник Чернівецького національного університету. Слов'янська філологія*, (585-586):80–84, 2012.

[54] ГМ Білик. Короткий тематичний літературознавчий словник. Ч. III. Версифікація.

[55] Смаглій Г. *«Теорія музики : Підруч. для навч. закл. освіти, культури і мистецтв.»*. Х.: Вид-во «Ранок», 2013.

[56] МИКОЛА ВАСИЛЬОВИЧ Гуцуляк. *Українське віршування 30-80-х рр. XVII ст*. PhD thesis, PhD thesis,«Теорія літератури». Чернівці: ЧНУ ім. Юрія Федьковича, 2017.

[57] Наталія Костенко. ЕЛЕМЕНТИ ТОНІЧНОГО І СИЛАБІЧНОГО ВІРШУВАННЯ В ДУМОВОМУ ВІРШІ ТГ ШЕВЧЕНКА. *НАШ УКРАЇНСЬКИЙ ДІМ*, page 38.

[58] АГ Кошовий and ГІ Кошовий. Одновимірні самоподібні фрактали та їх використання у моделюванні. 2011.

[59] Оксана Валентинівна Кудряшова. Functional poetics. 2016.

[60] ОВ Лазер-Паньків. та ін. ЛЛ Звонська, НВ Корольова. Ямбічна строфа. *Енциклопедичний словник класичних мов*, 2017.

[61] ОВ Любімова. Відтворення античних розмірів в українській поезії 80-х–90-х років XIX століття. *Науковий вісник Чернівецького університету. Романо-слов'янський дискурс*, (565):190–193, 2011.

[62] К Паладян. Початки силабо-тонічної версифікації в румунській літературі. *Питання літературознавства*, (81):164–172, 2010.

[63] ВД Шарко. Актуальні проблеми природничо-математичної освіти в середній і вищій школі.