Jan Taras

Warszawa 12.11.2017

**Projekt alfabetu Morse’a dla języka polskiego**

1. **Założenia projektowe**

Celem projektu jest stworzenie alfabetu Morse’a dostosowanego do charakterystyki języka polskiego o jak najwyższym stopniu kompresji i przy tym odpowiedniej łatwości użytkowania.

Oryginalny alfabet Morse’a stworzony był oczywiście do przesyłu informacji w języku angielskim, w którym zarówno liczba jak i częstotliwość występowania znaków wspólnych z językiem polskim różni się znacząco.

Jako, że projekt ma charakter dydaktyczny zostanie w nim omówionych kilka podejść do problemu z omówieniem wyników każdego z nich. Za materiał wzorcowy dla nowego kodu przyjęty został międzynarodowy standard kodu Morse’a zgodnie ze standardem Międzynarodowego Związku Telekomunikacyjnego na dzień 26 listopada 2009 [1].

Całość algorytmów, które posłużyły do zrealizowania projektu została napisana w języku Python, jest załączona w aneksie A do sprawozdania.

Jako, że modele kodu Morse’a są przeznaczone dla języka polskiego, nie istnieje żadna celowa korelacja pomiędzy kodowaniem polskim, a międzynarodowym.tego samego znaku alfabetu łacińskiego.

W kodowaniu nie użyto jednak żadnych znaków numerycznych standardu międzynarodowego, w celu zachowania kompatybilności chociaż w tym zakresie

1. **Tło historyczne**
2. **Przyjęta metodyka**

Podczas pracy nad językiem zrealizowano i zbadano efektywność dwóch modeli alfabetu Morse’a dla języka polskiego. Jeden model przewiduje mapowanie każdej litery do odpowiadającej jej sekwencji znaków Morse’a. Drugi przewiduje specjalny znak „dialkletyzacji” dla specyficznie polskich znaków.

* 1. **Wariant pierwszy – przypisanie znaków względem statystycznej częstotliwości ich występowania**

W wariancie pierwszym przyjęto najprostszą możliwą metodykę do stworzenia alfabetu Morse’a o jak najlepszych parametrach. Wszystkie znaki występujące w języku polskim posortowano pod względem ich częstotliwości występowania w języku polskim. Następnie z puli możliwych kombinacji znaków alfabetu Morse’a przypisywano poszczególnym znakom alfabetu polskiego, znaki Morse’a o długości odwrotnie proporcjonalnej do częstotliwości występowania znaku polskiego, tj. znakom najczęściej występującym w języku polskim zostały przypisane najkrótsze dostępne znaki alfabetu Morse’a.

|  |  |  |
| --- | --- | --- |
| Znak polski | Znak alfabetu Morse’a W. I | Częstotliwość występowania w języku polskim [%] |
| **ą** | \_.\_\_ | 0,79 |
| **ć** | \_\_\_. | 0,60 |
| **ę** | \_..\_ | 1,13 |
| **ź** | ..\_.. | 0,08 |
| **ó** | \_\_.. | 0,79 |
| **ł** | ..\_. | 2,38 |
| **ń** | ...\_. | 0,16 |
| **ś** | \_\_.\_ | 0,72 |
| **a** | .. | 8,37 |
| **b** | .\_\_. | 1,93 |
| **c** | \_.. | 3,89 |
| **d** | \_\_. | 3,35 |
| **e** | \_ | 8,68 |
| **f** | \_\_\_\_ | 0,26 |
| **g** | .\_\_\_ | 1,46 |
| **h** | \_... | 1,25 |
| **i** | . | 8,83 |
| **j** | ..\_\_ | 2,28 |
| **k** | \_\_\_ | 3,01 |
| **l** | .\_.. | 2,24 |
| **m** | ...\_ | 2,81 |
| **n** | \_. | 5,69 |
| **o** | .\_ | 7,53 |
| **p** | .... | 2,87 |
| **r** | ... | 4,15 |
| **s** | ..\_ | 4,13 |
| **t** | \_.\_ | 3,85 |
| **u** | .\_.\_ | 2,06 |
| **w** | .\_. | 4,11 |
| **y** | .\_\_ | 4,03 |
| **z** | \_\_ | 5,33 |
| **ż** | \_.\_. | 0,93 |

*Rys 3.1.1 Tabela częstotliwości dla wariantu pierwszego polskiego alfabetu Morse’a*

* 1. **Wariant drugi**

W wariancie drugim wzięto pod uwagę ważny aspekt pisowni języka polskiego jakim jest dialektyzacja wielu znaków (łącznie dotyczy to około 7,58 % znaków). Należy też zważyć na fakt że w wariancie pierwszym konieczne było użycie sekwencji o długości 5 znaków Morse’a do zakodowania liter ź i ń. Jest to fakt niewygodny dla liczby transmisji danego przekazu. Rozwiązaniem tego problemu zrealizowanym w wariancie drugim alfabetu Morse’a, który pozwala zredukować liczbę znaków w alfabecie do dwudziestu ośmiu, co pozwala na posługiwanie się wyłącznie sekwencjami o długości 4 lub mniej znaków Morse’a.

Rozwiązanie to ma też swoją wadę, gdyż w międzynarodowej specyfikacji alfabetu Morse’a[1] wymagana jest pauza długości jednego znaku długiego, co wydłuży czas transmisji. Żeby zbytnio nie komplikować metodyki języka zdecydowano się także na taką samą pisownię znaków ż i ź . Wynika to z faktu, że nie istnieją w języku polskim słowa, których zamiana litery ż na ź czy odwrotna dawałaby istniejące słowo o innym znaczeniu.

|  |  |  |
| --- | --- | --- |
| Znak polski | Znak alfabetu Morse’a W. I | Częstotliwość występowania w języku polskim [%] |
| **(diakletyzacja)** | .\_ | 7,58 |
| **ć** | \_.\_|.\_ | 0,60 |
| **ę** | \_|.\_ | 1,13 |
| **ź / ż** | …|.\_ | 1,01 |
| **ó** | \_.|.\_ | 0,79 |
| **ł** | .\_.. |.\_ | 2,38 |
| **ń** | \_\_|.\_ | 0,16 |
| **ś** | .\_. |.\_ | 0,72 |
| **a** | .. | 8,37 |
| **b** | .\_\_. | 1,93 |
| **c** | \_.\_ | 3,89 |
| **d** | \_\_\_ | 3,35 |
| **e** | \_ | 8,68 |
| **f** | \_..\_ | 0,26 |
| **g** | .\_\_\_ | 1,46 |
| **h** | \_... | 1,25 |
| **i** | . | 8,83 |
| **j** | ..\_\_ | 2,28 |
| **k** | \_\_\_ | 3,01 |
| **l** | .\_.. | 2,24 |
| **m** | ..\_. | 2,81 |
| **n** | \_. | 5,69 |
| **o** | \_. | 7,53 |
| **p** | ...\_ | 2,87 |
| **r** | ..\_ | 4,15 |
| **s** | .\_. | 4,13 |
| **t** | \_\_. | 3,85 |
| **u** | .\_.\_ | 2,06 |
| **w** | .\_\_ | 4,11 |
| **y** | \_.. | 4,03 |
| **z** | ... | 5,33 |

*Rys 3.1.2 Tabela częstotliwości dla wariantu pierwszego polskiego alfabetu Morse’a (znak | zonacza pauze pomiędzy dwoma znakami)*

1. Wyniki działania algorytmu

Skuteczność działania algorytmu kodowania poszczególnymi wariantami polskiego języka Morse’a mierzono porównując obie metody konwersji pod względem:

1. Długości transmisji
2. Liczby użytych znaków Morse’a
3. Stratności znaczenia ( konwersja ż/ź )
4. Łącznej (razem z pauzami) liczby użytych znaków

Parametry te zmierzono dla kilku korpusów tekstów w języku polskim:

1. Literatury klasycznej udostępnianej przez inicjatywę projektu Gutenberg [3]
2. Współczesnych treści prasowych z oficjalnej strony TVN [4]
3. Komentarzy z polskich stron społecznościowych - Reddit, Wykop – potocznej mowy polskiej [5,6]

Wykresy parametrów konwersji i charakterystykę poszczególnych korpusów zamieszczono po niżej

1. Wnioski

W kontekście … lepszy okazał się wariant … . Dla … lepsza jest konwersja .

Ostatecznie lepsze parametry na wszystkich badanych polach wykazuje … .

1. Bibliografia i źródła
2. <http://www.itu.int/dms_pubrec/itu-r/rec/m/R-REC-M.1677-1-200910-I!!PDF-E.pdf>
3. http:// <https://www.sttmedia.com/characterfrequency-polish> (stefantrost.de)
4. <https://www.gutenberg.org>
5. <http://prasa.tvn.pl>
6. <https://www.reddit.com/r/Polska>
7. http://www.wykop.pl

Dodatek A Kod źródłowy programów

**Program do tworzenia kodowań Morse’a z częstotliwości znaków zapisanych w pliku JSON**

# -\*- coding: utf-8 -\*-

#! usr/bin/python

import io

import json

import itertools

import re

def fetch\_max(dictio):

key = max(dictio, key=dictio.get)

del dictio[key]

return key

def is\_ascii(s):

return all(ord(c) < 128 for c in s)

# for filtering out numerical sings

def num\_sign(s):

return len(s) == 5 and (bool(re.search(r'^\_\*\.\*$',s)) or bool(re.search(r'^\.\*\_\*$',s)))

# Mapowanie bezposrednie

with io.open('./frequencies/pl.json','r',encoding='utf-8') as file:

dictionary = file.read()

frequencies = json.loads(dictionary)

encoded\_dict = {}

char\_array = '.\_'

sign\_length = 1

while frequencies:

product\_pool = itertools.product(char\_array, repeat=sign\_length)

for sign in product\_pool:

if not frequencies:

break

znak = ''.join(sign)

if num\_sign(znak):

continue

encoded\_dict[fetch\_max(frequencies)] = znak

sign\_length +=1

with io.open('./morse\_models/direct\_map.json', 'w', encoding='utf-8') as output:

json\_body = json.dumps(encoded\_dict, ensure\_ascii=False)

output.write(json\_body)

# Mapowanie ze znakiem diakletyzacji

frequencies = json.loads(dictionary)

encoded\_dict = {}

sign\_length = 1

sum\_diak = 0

for l in frequencies:

if not is\_ascii(l):

sum\_diak += frequencies[l]

frequencies = {k: v for k, v in frequencies.iteritems() if is\_ascii(k)}

frequencies['diak'] = sum\_diak

while frequencies:

product\_pool = itertools.product(char\_array, repeat=sign\_length)

for sign in product\_pool:

if not frequencies:

break

znak = ''.join(sign)

if num\_sign(znak):

continue

key = fetch\_max(frequencies)

encoded\_dict[key] = znak

sign\_length +=1

# tabela diakletyzacji

tab = {

'ą': encoded\_dict['a'] + '|' + encoded\_dict['diak'],

'ę': encoded\_dict['e'] + '|' + encoded\_dict['diak'],

'ó': encoded\_dict['o'] + '|' + encoded\_dict['diak'],

'ś': encoded\_dict['e'] + '|' + encoded\_dict['diak'],

'ż': encoded\_dict['z'] + '|' + encoded\_dict['diak'],

'ć': encoded\_dict['c'] + '|' + encoded\_dict['diak'],

'ń': encoded\_dict['n'] + '|' + encoded\_dict['diak'],

}

with io.open('./morse\_models/diak\_map.json', 'w', encoding='utf-8') as output:

json\_body = json.dumps(encoded\_dict, ensure\_ascii=False)

output.write(json\_body)

with io.open('./morse\_models/diak\_tab.json', 'w', encoding='utf-8') as output:

json\_body = unicode(json.dumps(tab, ensure\_ascii=False).decode('utf-8'))

output.write(json\_body)