Instytut Podstawowych Problemów Techniki
Polskiej Akademii Nauk

Praca doktorska

Zastosowanie algorytmu odległości edycyjnej do ilościowej analizy danych tekstowych

mgr inż. Artur Niewiarowski

Promotor: dr hab. inż. Marek Stanuszek

Kraków 2024

Autor pracy był stypendystą w ramach projektu „*Doctus - Małopolski fundusz stypendialny dla doktorantów*” współfinansowanego ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego

Spis treści

1. Wstęp

Internet to niewątpliwie gigantycznych rozmiarów niekonwencjonalna baza danych, zawierająca zarówno obiekty graficzne, filmy, jak również przede wszystkim niezliczone zasoby tekstu. Tekstu ulokowanego w postaci artykułów na portalach internetowych, blogach, encyklopediach oraz książkach (np. w tzw. e-bookach) czytanych przez internautów na komputerach lub urządzeniach mobilnych. Tekstu będącego przydatnym zasobem wiedzy, jak również tego małowartościowego. Niezależnie od jego przydatności, na przełomie lat zaistniała potrzeba budowy wydajnych algorytmów i nowoczesnych modeli danych umożliwiających analizowanie i wyciąganie istotnych informacji oraz wniosków z każdego zasobu tekstowego. Dotyczy to danych udostępnionych poprzez globalną sieć, jak również zlokalizowanych na lokalnych dyskach domowych komputerów. Ważność i zasadność prac nad takimi mechanizmami w obecnych czasach nie powinna budzić żadnych wątpliwości, ponieważ korzystają z nich obecnie miliony ludzi na całym świecie poprzez różnego rodzaju aplikacje wspomagające codzienne procesy społeczne. Niniejsza praca jest efektem badań nad algorytmami analizy danych tekstowych, a dokładniej propozycją nowych rozwiązań w zakresie komparacji zbiorów danych tekstowych.

1.1. Analiza danych na przełomie wieków

Algorytmy analizy danych tekstowych – dziś tak popularne m.in. w wyszukiwarkach internetowych i systemach anty-plagiatowych – rozwijane są wbrew pozorom nie od zeszłego wieku, który wydaje się być w tej dziedzinie niezwykle odkrywczy, ale dużo wcześniej, bo już od wieku XV. Wtedy to Lorenzo Valla[1] w 1439 roku dzięki opracowanym przez siebie technikom kategoryzowanym dzisiaj jako metody stylometryczne, poddał analizie Donacje Konstantyna – dokument, w którym napisane jest, iż cesarz Rzymu Konstantyn Wielki nadaje Kościołowi katolickiemu liczne przywileje oraz oddaje miasto we władanie papieży. Wnikliwa analiza morfologiczna, semantyczna i składniowa dokumentu wykazała, że jest to falsyfikat i powstał kilkaset lat później. Ostatecznie analiza ta przyczyniła się do tego, że Marcin Luter zakwestionował władzę papieży [1], a opracowana

[1] Valla Lorenzo – właśc. L. della Valle, ur. ok. 1406, zm. 1457, włoski filozof, humanista, filolog. Encyklopedia PWN

metoda analizy danych tekstowych wzmocniła kształtujący się ówcześnie ruch protestancki, jednocześnie zapisując się nieoficjalnie na kartach historii rozwoju nowej religii w Europie i na świecie - protestantyzmu. Lata bliższe obecnym, które należy wymienić w aspekcie rozwoju metod zaliczanych do ilościowej analizy danych tekstowych to koniec wieku XIX, kiedy to angielski matematyk prof. Augustus de Morgan analizował pod kątem autentyczności pisma św. Pawła poprzez pomiar długości wyrazów w listach. W kolejnych latach XX wieku naśladowcami opracowanych przez niego metod byli m.in. T. C. Mendenhall – badający dzieła takich autorów jak: Cezar, Dickens, Dumas, Shakespeare [30,31,32]; Lewis Campbell, W.D. Ross, Constantin Ritter oraz Wincenty Lutosławski – analizujący dzieła Platona [33], przyczyniając się tym samym do rozwoju stylometrii. Po drugiej połowie XX wieku nastąpił nagły wzrost badań związanych z metodami ilościowej analizy danych tekstowych. Był on konsekwencją rozwoju elektroniki, czyli komputerów, a w następstwie możliwością budowania algorytmów, które szybciej i dokładniej od człowieka były w stanie analizować dane. Wtedy też zaczęto rozwijać nowe metody związane z analizowaniem tekstów pod kątem: autorstwa, datowania tekstów literackich, jak również te związane z dzisiejszymi systemami anty-plagiatowymi, wszystkie te elementy stanowią podstawę dzisiejszej stylometrii. Konsekwencją tego były pierwsze podręczniki (w tym najpopularniejszy: Susan Hockey, *Guide to Computer Applications in the Humanities* [2]) i publikacje naukowe z przełomu lat 80 i 90 (np. [4][15-16][21-22]) łączące ówczesne literaturoznawstwo z informatyką. W czasach teraźniejszych dzięki ogólnodostępnemu internetowi i bardzo wydajnym wielordzeniowym komputerom osobistym (oraz dostępności klastrów obliczeniowych dla nauki[2]), testowanie istniejących mechanizmów analizy danych tekstowych i propozycja nowych stały się dużo prostsze. Dlatego też w ostatnich latach powstały prace naukowe integrujące ze sobą zagadnienia z różnych dziedzin nauki, związane m.in. ze sztuczną inteligencją[3], językoznawstwem i eksploracją danych tekstowych (ang. *text-mining*), czyli ogólnie rozumianą lingwistyką komputerową (ang. *Natural Language Processing - NLP*). Algorytmy będące wynikiem badań i prac komercyjnych zaimplementowane zostały m.in. w wysokiej jakości

[2] Więcej informacji znajduje się na stronie projektu *PLGrid*: http://www.plgrid.pl/

[3] Sztuczna inteligencja - dział informatyki badający reguły rządzące zachowaniami umysłowymi człowieka i tworzący programy lub systemy komputerowe symulujące ludzkie myślenie. Encyklopedia PWN

wyszukiwarkach internetowych (np. *Google*[4]), czy też w inteligentnych translatorach [5] tłumaczących teksty napisane w bardzo egzotycznych językach[5].

Jednak pomimo wielu lat rozwoju metod związanych z ilościową analizą danych tekstowych sięgających kilku wieków wstecz, te pokrewne dziedziny badań powiązane z lingwistyką komputerową nadal wymagają zaangażowania wielu zespołów badawczych z całego świata i propozycji coraz to nowych, szybszych, odporniejszych na błędy mechanizmów do analizy danych tekstowych.

1.2. Aktualny stan badań

Większość mechanizmów analizujących dane tekstowe, w tym w szczególności systemy porównujące ze sobą dokumenty w celu późniejszej klasteryzacji, bazuje w głównej mierze na tzw.: korpusie[6], słowniku wyrazów pochodnych o podobnym znaczeniu (tzw. tezaurus, ang. *thesaurus*), stemmingu i lematyzacji. Opierają się one przede wszystkim na takich znanych algorytmach, jak: Lovinsa [3], Portera [4], Dawsona [17], Paice-Huska [15], Krovetza [16], algorytmie fonetycznym Soundex[7] oraz na ich modyfikacjach i ulepszeniach (np. [19-20]). Stemming i lematyzacja to procesy mające na celu umożliwienie porównywania ze sobą wyrazów o podobnym znaczeniu, ale składających się z odmiennego zestawu liter. Różnią się między sobą tym, że stemming to proces wydobycia z wyrazu tzw. rdzenia (stem-u), czyli jego nieodmiennej części, potencjalnie wspólnej dla innych analizowanych wyrazów (np. poprzez ucinanie końcówek)[8]. Rdzeń niekoniecznie reprezentuje wyraz zdefiniowany w słowniku danego języka, lecz jego fragment. Natomiast lematyzacja, to proces sprowadzenia wyrazu do jego lematu, czyli wspólnej grupy wyrazów, posiadającego definicję w słownika wyrazów[9]. Ten ostatni bazuje dodatkowo na tzw. kontekście, czyli otoczeniu wyrazu, jak również

[4] Adres internetowy: https://www.google.com

[5] Adres internetowy: https://translate.google.com/intl/en/about/languages/

[6] Korpus – zbiór reprezentatywnych tekstów służący programom komputerowym analizującym dane tekstowe na określenie konstrukcji zdań oraz kontekstów w jakich występują analizowane wyrazy

[7] Soundex – https://www.archives.gov/research/census/soundex.html. Popularnym miejscem implementacji algorytmu jest system zarządzania bazą danych MySQL/MariaDB (https://mariadb.com/kb/en/soundex/)

[8] Przykłady: Politechnika, Politechnice, politechniczny, Politechniki → politech

[9] Przykłady: wiórkami → wiórek, jeżdżący → jeździć, piszący → pisać

budowie gramatycznej zdania i tym różni się od stemmingu. W obu przypadkach można wyodrębnić dwa główne podejścia analizy wyrazów:

- pierwsze – algorytmiczne, w którym dany algorytm posiada zaimplementowany zestaw reguł potrafiących wykryć różnice gramatyczne pomiędzy wyrazami,
- drugie – słownikowe, bardziej dokładne, bazujące na słowniku form gramatycznych, zawierających dodatkowo rdzeń i lemat.

Wymienione powyżej metody, a także ich modyfikacje mają poważne ograniczenia, tzn. są zaprojektowane w oparciu o reguły gramatyczne konkretnych języków, ich struktur, czyli są zależne językowo. W większości przypadków dotyczą j. angielskiego, natomiast na uwagę zasługuje dynamiczny, również w ostatnich latach wzrost badań naukowych nad algorytmami analizy danych tekstowych dla j. polskiego – np.: *słowosieć,* będąca projektem Grupy Technologii Językowych Politechniki Wrocławskiej[10] [7-8], stemmer dla języka polskiego opracowany przez D. Weissa [26-27], czy algorytm Stempel[11]. Najwięcej algorytmów analizy wyrazów zostało opracowanych nieprzypadkowo dla języka angielskiego, ponieważ charakteryzuje go uboga fleksja i proste zasady tworzenia słów pochodnych (najprostszymi przykładami są chociażby reguły tworzenia liczby mnogiej lub trzeciej osoby), natomiast co z pozostałymi językami? Powyższa cecha uniemożliwia lub znacząco utrudnia badanie tekstów napisanych w językach mniej popularnych, np. takich, którymi posługuje się relatywnie mała liczba ludzi – np. Kraje Nadbałtyckie (w tym Estonia – ok. 1,5 mln mieszkańców), czy byłego Związku Radzieckiego, w których język ojczysty został wyparty językiem rosyjskim (np. Białoruś, Uzbekistan [18]).

Dodatkowo (tj. analizując aspekt techniczny, czyli tworzenia kodu specjalistycznego oprogramowania), biorąc pod uwagę złożoność przypadków reguł gramatycznych koniecznych do implementacji dla danego języka lub rozmiar słownika form pochodnych, który należy zaalokować w pamięci komputera, a później analizować – rozwiązania te mogą okazać się problematyczne w implementacji w postaci programu komputerowego oraz mało-wydajne[12]. Natomiast, nie zmienia to faktu, że część z tych mechanizmów jest bardzo dokładna i przyczynia się istotnie do rozwoju badań naukowych nad analizą języka mówionego i pisanego oraz rozwojem specjalistycznego oprogramowania otwartego jak i

[10] Więcej informacji znajduje się na stronie internetowej grupy: http://www.nlp.pwr.wroc.pl/
[11] Informacja o algorytmie i licencji: https://lucene.apache.org/core/7_5_0/analyzers-stempel/index.html
[12] Przykład implementacji WordNet dla języka angielskiego w języku programowania Java: https://github.com/mgruben/WordNet

komercyjnego – czego przykładem jest chociażby projekt *WebSty*[13] [23-25], w tym system *SuperMatrix* ([28-29]) mający na celu szybkie poszerzanie polskiej *słowosieci* (ang. *wordnet*)[14] [5-8] – będącej obecnie największym na świecie wordnetem. Słowosieć to relacyjny słownik semantyczny odzwierciedlający system leksykalny danego języka, znajdujący zastosowanie m.in. w znanym na całym świecie translatorze *Google Translate*[15].

1.2.1. Zastosowanie sztucznej inteligencji w analizie danych tekstowych

Sztuczna inteligencja (AI) zyskuje na znaczeniu w analizie danych, umożliwiając automatyzację skomplikowanych procesów oraz uzyskiwanie precyzyjnych wyników w krótszym czasie niż tradycyjne metody. Jednym z kluczowych obszarów, w którym AI ma szczególne zastosowanie, jest analiza tekstów m.in. pod względem podobieństwa. Techniki te są niezwykle istotne w różnych dziedzinach, takich jak nauka, biznes, prawo czy edukacja. Porównywanie tekstów za pomocą AI opiera się na zaawansowanych algorytmach, które mogą analizować ogromne ilości danych tekstowych, identyfikując wzorce i relacje między różnymi fragmentami tekstu. Do najczęściej stosowanych metod należą analiza semantyczna oraz techniki oparte na uczeniu maszynowym, takie jak wektoryzacja tekstu i modele językowe.

Analiza semantyczna polega na badaniu znaczenia słów i zdań w kontekście, co pozwala na bardziej precyzyjne porównanie tekstów niż proste porównanie słów kluczowych. Modele semantyczne, takie jak Word2Vec[16][40] (stworzone przez pracowników firmy Google Inc. na potrzeby wyszukiwarki internetowej), czy GloVe[17][41], przekształcają słowa w wektory o dużej liczbie wymiarów, które odzwierciedlają ich znaczenie w kontekście całego korpusu tekstu. Dzięki temu możliwe jest porównywanie tekstów na poziomie znaczeniowym, a nie tylko leksykalnym.

W ostatnich latach modele językowe, takie jak BERT[18][42] (ang. *Bidirectional Encoder Representations from Transformers*) czy GPT[19][43-45,48] (ang. *Generative Pre-trained*

[13] Interfejs webowy dostępny z adresu: http://ws.clarin-pl.eu/websty.shtml
[14] Strona projektu polskiej słowosieci: http://plwordnet.pwr.wroc.pl/wordnet/about
[15] Interfejs webowy dostępny jest z adresu: https://translate.google.pl/
[16] https://arxiv.org/pdf/1301.3781
[17] https://aclanthology.org/D14-1162.pdf
[18] https://arxiv.org/pdf/1810.04805v2
[19] https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf

Transformer), zrewolucjonizowały analizę tekstów. Te modele potrafią nie tylko rozumieć kontekst, ale także generować teksty oraz przeprowadzać skomplikowane analizy porównawcze. Modele te są trenowane na ogromnych zbiorach danych, co pozwala im na uchwycenie subtelnych różnic i podobieństw między tekstami.

Mimo ogromnych postępów jakie dokonały się w ostatnich latach, analiza tekstów za pomocą AI stawia również nadal poważne wyzwania. Modele językowe muszą być stale aktualizowane i dostosowywane do zmieniających się języków i kontekstów. Ponadto, konieczne jest zapewnienie odpowiedniej ochrony danych oraz przestrzeganie etyki w zakresie przetwarzania informacji. Dodatkowo analiza tekstów za pomocą AI wymaga ogromnej mocy obliczeniowej, która często przekracza możliwości standardowych komputerów domowych. Zaawansowane modele językowe i algorytmy uczenia maszynowego wymagają pracy na jednostkach serwerowych o dużej mocy obliczeniowej, wyposażonych w specjalistyczny sprzęt, taki jak procesory graficzne (GPU) i jednostki przetwarzania tensorowego (TPU). Z tego powodu technologie te są zazwyczaj niedostępne do instalacji na komputerach osobistych. Użytkownicy muszą korzystać z zewnętrznych usług, takich jak platformy chmurowe oferujące dostęp do zaawansowanych zasobów obliczeniowych. Dzięki temu możliwe jest skalowanie obliczeń i wykonywanie skomplikowanych analiz bez potrzeby inwestowania w kosztowny sprzęt.

1.3. Cel pracy

Celem niniejszej pracy jest opracowanie i implementacja nowej efektywnej metody pozwalającej na analizę danych tekstowych, niezależnej od systemów leksykalnych większości języków o korzeniach europejskich. Ma ona stanowić bazę wyjściową do budowy efektywnych systemów anty-plagiatowych oraz algorytmów stylometrycznych, wykorzystujących pełną moc obliczeniową dzisiejszych komputerów - zarówno domowych jak i przemysłowych (w tym naukowych). Poprzez niezależność leksykalną, należy rozumieć np. brak konieczności implementacji (czego mechanizm nie wyklucza) procesów sprowadzania analizowanych wyrazów do form podstawowych[20], jak również wspólnych grup wyrazowych, poprzez analizowanie kontekstu, w celu osiągnięcia satysfakcjonującego wyniku analizy. Przekłada się to na uniwersalność całości mechanizmu względem badanych dokumentów napisanych w danych językach, jak również znacząco przyspiesza jego działanie, ze względu na brak konieczności implementacji wspomnianych procesów. Dodatkowo, metoda powinna być na tyle wydajna, aby można ją było zainstalować jako oprogramowanie na komputerze o standardowych parametrach obliczeniowych (np. laptopie), dzięki czemu nie trzeba będzie wysyłać do wspomnianych w poprzednim rozdziale chmur plików w celach przeprowadzenia analizy porównawczej. Wiele firm wprowadziło przepisy zabraniające swoim pracownikom wysyłania dokumentów elektronicznych de facto do firm trzecich, co sprawia, że zamysł tego typu metody powinien dodatkowo zyskać na znaczeniu.

Produktami analizy zaproponowanej w pracy metody będą:

- postać tekstowa (formaty: TXT oraz XML[21]) – raport porównania dokumentów z wyszczególnieniem zakresu fragmentów zdań uznanych za podobne,
- postać graficzna (obraz PNG) – wykres korelacji podobieństwa dokumentów tekstowych, będący jednocześnie graficzną prezentacją ich podobieństwa.

[20] wyraz podstawowy – wyraz będący podstawą słowotwórczą wyrazów od niego utworzonych. Słownik języka polskiego PWN

[21] XML [ang. *Extensible Markup Language*] – język formalny określający uniwersalny sposób zapisu informacji przez programy komputerowe. Encyklopedia PWN.

1.4. Zakres pracy

Dla osiągnięcia sformułowanych w poprzednim punkcie celów niniejszej pracy, we wstępie konieczne było przedstawienie krótkiego zarysu historycznego obrazującego rozwój metod związanych z analizą danych tekstowych na przełomie wieków, a w szczególności miejsc, czasów i powodów ich powstania, ewolucji oraz obecnego miejsca w świecie nauki i zastosowań komercyjnych. Dalej w odniesieniu do nakreślonego celu przeprowadzę analizę aktualnego stanu badań naukowych. Będzie to przede wszystkim przegląd popularnych i istotnych z punktu widzenia algorytmów i metod analizy danych tekstowych, które rozwijane były przez ostatnie kilkadziesiąt lat intensywnych badań. W dalszej części pacy przedstawiona zostanie ewolucja koncepcji powstania rozwiązań będących celem pracy, tj. od analizy pojedynczych wyrazów, poprzez krótkie fragmenty zdań, skończywszy na badaniu podobieństwa tekstów dowolnej wielkości napisanych w dowolnych językach europejskich (w tym również w różnych językach, ale w ramach tych samych grup językowych). Szczegółowe przedstawienie obiektów badań i przebieg testów metody zostanie poprzedzony jego implementacją w postaci pseudokodu.

Podsumowanie rozprawy w rozdziale 5. stanowi jednocześnie konkluzję jako element wyjściowych do dalszych prac naukowych nad analizą złożoności dokumentów tekstowych i chociażby wynikających stąd możliwości badań stylometrycznych w zakresie identyfikacji autorstwa tekstów, również w aspekcie AI.

2. Przegląd wybranych algorytmów analizy danych tekstowych

Niejednokrotnie algorytmy analizy danych tekstowych o prostrzej strukturze implementacyjnej pełnią ważną rolę składową wielu większych mechanizmów analizujących duże zbiory danych. Większość z nich jest ściśle związana z danym językiem oraz prawami, które w nim się żądzą.

Powyższy rozdział ma na celu przedstawienie wybranych popularnych algorytmów analizujących zbiory tekstowe, które powstawały na przełomie lat i zestawienie ich z koncepcją będącą przewodnią tematyką pracy, czyli próbą stworzenia metody uniwersalnej językowo.

2.1. Grupowanie dokumentów a ważenie terminów w modelu wektorowym

Większość mechanizmów analizujących poprawność zdań pod względem ortograficznym, czy też gramatycznych opiera się o zaimportowaną wcześniej bazę tekstów – tzw. korpus. Poprawność, ale również szybkość analizy zależy od doboru właściwych tekstów, jak również metod analizy i modeli reprezentacji danych. Istnieje możliwość stworzenia interfejsu mechanizmu, który umożliwiałby wstępne ustawienie dziedzinowości analizowanych danych. Niestety wielkość zasobów, jak również szczegółowość danych tekstowych w danej dziedzinie, może znacząco spowolnić proces mechanizmu. W tym celu zastosowanie mają algorytmy grupowania dokumentów tekstowych (fragmentów tekstów), umożliwiające wstępne oszacowanie (tzn. przyporządkowanie) badanego tekstu do zasobów. Istnieje wiele metod grupowania dokumentów, do najważniejszych należą metody: płaskie, hierarchiczne, grafowe, oparte na gęstości oraz inne. Określają one sposób segregacji danych i ich ostateczną reprezentację jako wynik analizy. Dla wybranej metody należy określić model danych, umożliwiający reprezentację danych tekstowych (zdań, jak również całych dokumentów), jak również konkretne operacje oszacowania stopnia relacyjności (podobieństwa) pomiędzy dokumentami. Do najczęściej stosowanych modeli należą: wektorowy, grafowy i przestrzeni metrycznej.

Najpopularniejszy modelem reprezentacji danych jest model wektorowy *(ang. Vector Space Model).* Dokument (fragment dokumentu) jest przedstawiony jako n – wymiarowy wektor cech.

$$\mathbf{D}_j = (\omega_{j,i}, \omega_{j,i+1}, ..., \omega_{j,n-1}, \omega_{j,n}), i \in \langle 1, n \rangle \tag{2.1}$$

gdzie:
D – wektor reprezentujący dokument (fragment dokumentu),
j – identyfikator dokumentu,
i – liczba naturalna oznaczająca kolejne słowa lub grupy słów określające termin,
n – liczba terminów w j-tym dokumencie,
$\omega_{j,i}$ – cecha, liczba rzeczywista będąca wartością znaczeniową terminu.

Wektor jest tworzony na podstawie wartości liczbowych (cech), obliczanych poprzez proces ważenia. Ważenie nadaje każdemu terminowi (tj. słowu, grupie słów, ang. *term*) wartość znaczeniową, najczęściej jest nią liczba wystąpień w dokumencie. Tego typu schemat określany jest w języku angielskim jako *term frequency*:

$$tf_{t,\mathbf{D}} \tag{2.2}$$

gdzie:

t – termin,
D – dokument.

Poważnym problemem tego typu modelu jest brak rozróżniania istotności poszczególnych terminów, charakteryzujących zagadnienia (dziedzinę) zawarte w dokumencie. Innym rozwiązaniem jest oszacowanie danego wystąpienia w badanych dokumentach i skorzystanie z odwrotnej częstości dokumentowej (ang. *inversed document frequency*), opisanej poprzez wzór:

$$idf_t = \log \frac{N}{df_t} \tag{2.3}$$

gdzie:
N – liczba wszystkich badanych dokumentów,
df_t – częstość dokumentowa – określa liczbę dokumentów dla danego wystąpienia terminu.

Ostatecznie, w celu doprecyzowania schematu ważenia terminów, stosuje się następujący wzór, którego wynik jest iloczynem kartezjańskim odwrotnej częstości dokumentowej i częstości wystąpienia terminu[22]:

$$tf\text{-}idf_{t,\mathbf{D}} = \log \frac{N}{df_t} \times tf_{t,\mathbf{D}} \tag{2.4}$$

2.1.1. Popularne algorytmy stemmingu i lematyzacji

Jak już zostało wspomniane wcześniej, algorytmy stemmingu i lematyzacji są istotnym elementem składowym mechanizmów analizy porównawczej danych tekstowych. Poniżej przedstawione zostały wybrane algorytmy, będące nadal w użyciu, jak również będące bazą wyjściową do nowych modyfikacji.

2.1.2. Algorytm Lovins

Autorem algorytmu jest nieżyjąca już Julie Beth Lovins, która jako pierwsza skonstruowała i opisała algorytm stemmingu w roku 1968 [3]. Algorytm powstał dla języka angielskiego i charakteryzuje się posiadaną znaczną liczbą końcówek (*ang. endings*). Przebiega w dwóch krokach. W pierwszym kroku wyróżnić można pogrupowane względem długości końcówki, których dotyczy 29 warunków. Badany w tym kroku wyraz zestawiany jest z bazą końcówek odpowiedniego od najdłużej do najkrótszej, a następnie poddawany warunkowi, który ostatecznie decyduje, czy końcówka ze słowa zostanie usunięta. W kroku drugim powstały ciąg znaków jest dopracowywany poprzez tzw. zasady transformacyjne, które modyfikują występujące po sobie litery, np. usuwają lub podmieniają. Ciąg tekstowy będący wynikiem kroku drugiego jest wynikiem wykonania całości algorytmu, zwracającego rdzeń. Zaletą algorytmu jest szybkość oraz prosta implementacja w postaci programu komputerowego, natomiast minusem niekompletna lista końcówek, co sprawia, że obliczony rdzeń jest błędny.

2.1.3. Algorytm Portera

Algorytm stemmingu opracowany również dla języka angielskiego przez Martina Portera w 1980 roku [4]. Charakteryzuje się ośmioma krokami przekształcającymi podany ciąg tekstowy według zdefiniowanych w kodzie programu warunków bazujących na tzw. mierze

[22] http://nlp.stanford.edu/IR-book/pdf/06vect.pdf, s. 119

słowa, zamiast na długości rdzenia będącego cechą poprzedniego algorytmu. Dodatkowo na potrzeby algorytmu definiuje się w kodzie znaki charakteryzujące odpowiednio spółgłoski i samogłoski w połączeniu z wspomnianą miarą słowa.

2.1.4. Algorytm Dawsona

Stemmer Dawsona [17] to rozszerzenie podejścia Lovinsa, które obejmuje znacznie większą liczbę sufiksów, dochodzącą do około 1200. Sufiksy te są przechowywane w odwróconym porządku, indeksowane według ich długości i ostatniej litery, co pozwala na szybsze dopasowywanie i usuwanie najdłuższych pasujących sufiksów w pojedynczym przejściu przez słowo. Taka organizacja danych w formie rozgałęzionych drzew znaków pozwala na szybki dostęp do nich, co z kolei przekłada się na wydajność działania stemmera.

Dzięki temu stemmer Dawsona jest bardzo szybki i zapewnia dokładniejsze wyniki stemmingu poprzez pokrycie szerszego zakresu sufiksów. Jednak struktura danych i sposób indeksowania końcówek mogą sprawiać, że implementacja tego algorytmu jest bardziej skomplikowana niż w przypadku bardziej tradycyjnych stemmerów, co może stanowić barierę dla niektórych zastosowań.

2.1.5. Algorytm Paice-Huska

Algorytm Paice-Huska [15] to algorytm do analizy danych tekstowych, który skupia się na stemizacji słów, czyli procesie usuwania morfologicznych i fleksyjnych końcówek słów w celu pozostawienia ich podstawowej formy, zwaną stemem. Algorytm został opracowany przez Chrisa D. Paice'a i Paula Huska w 1990 roku. Celem stemizacji jest zredukowanie różnych form tego samego słowa do wspólnego rdzenia, co ułatwia analizę tekstu, wyszukiwanie informacji, a także indeksowanie i grupowanie tekstów. Algorytm Paice-Huska opiera się na podejściu opartym na regułach, wykorzystując skończony zbiór reguł wyrażonych za pomocą wyrażeń regularnych. Reguły te opisują, jakie końcówki mają zostać usunięte, a także jak zmodyfikować pozostałą część słowa, aby uzyskać stem.

Algorytm Paice-Huska został pierwotnie opracowany dla języka angielskiego, ale z czasem został zaadaptowany do innych języków. Dla języka polskiego możliwe są odmienne reguły stemizacji, które uwzględniają specyfikę polskiej gramatyki i morfologii.

2.1.6. Algorytm Krovetza

Kolejnym algorytmem jest algorytm Krovetza [16], który stanowi podejście do lematyzacji słów opracowane przez Roberta Krovetza w 1993 roku. Algorytm działa na zasadzie wykorzystania słownika do przekształcania słów w tekstach. Algorytm Krovetza został pierwotnie opracowany dla języka angielskiego, ale zasada działania może być zaadaptowana do innych języków, takich jak polski, z odpowiednim słownikiem i regułami morfologicznymi. Algorytm ten jest powszechnie stosowany w wyszukiwarkach internetowych, analizie tekstu i eksploracji danych. Przykładem zastosowania algorytmu Krovetza jest system wyszukiwania informacji, który lematyzuje słowa w zapytaniach użytkowników oraz w indeksowanych dokumentach, aby ułatwić odnalezienie najbardziej istotnych wyników. Załóżmy, że mamy zbiór dokumentów tekstowych, które chcemy indeksować. Przed przystąpieniem do indeksowania, stosujemy algorytm Krovetza, aby przekształcić słowa w tych dokumentach do ich podstawowej formy (lematu). W wyniku tego procesu, słowa takie jak "running", "ran" i "runs" są zamieniane na "run", natomiast "better" na "good". Następnie, gdy użytkownik wprowadza zapytanie do wyszukiwarki, stosujemy również algorytm Krovetza do przekształcenia słów z zapytania. Przykładowo, jeśli użytkownik wpisze zapytanie "running marathons", zamieniamy je na "run marathon". Teraz, gdy mamy zarówno zapytanie, jak i indeksowane dokumenty w postaci zlematyzowanej, możemy łatwiej porównać zapytanie z dokumentami, aby znaleźć te, które są najbardziej istotne dla potrzeb użytkownika. W efekcie, wyniki wyszukiwania będą bardziej precyzyjne, ponieważ lematyzacja pomaga wyeliminować różnice morfologiczne i skupić się na znaczeniu słów. Warto zauważyć, że algorytm Krovetza może być również używany w innych zastosowaniach analizy tekstu, takich jak klasyfikacja tekstu, analiza sentymentu, czy ekstrakcja informacji.

2.1.7. Algorytm Soundex

Ciekawą koncepcją analizy ciągów tekstowych jest algorytm Soundex, który został opracowany przez Robert C. Russell i Margaret King Odell w USA celem sortowania nazwisk w spisie ludności względem ich brzmienia, a nie alfabetycznie. Obecnie rozpowszechniony jest w różnych systemach przetwarzających dane, w tym m.in. w systemach zarządzania bazami danych, jak: IBM Db2, PostgreSQL, SQLite, Ingres, MS SQL Server, Oracle, MySQL i bazującym na nim MariaDB, gdzie ciągi tekstowe zawarte w tabelach można ze sobą zestawiać względem

brzmienia (np. operator w SQL: *sounds like*). Algorytm w oryginale przebiega w czterech krokach, w których poszczególne znaki z wyjątkiem pierwszego podmieniane są poprzez cyfry według zdefiniowanego schematu, a następnie poszczególne kody są odpowiednio eliminowane. Ostatecznie powstało kilka odmian algorytmu eliminujących jego wady, które polegały na błędnej ocenie podobieństwa pomiędzy zestawianymi wyrazami. Przykład działania algorytmu Soundex w systemie zarządzania bazami danych MariaDB przedstawia wynik zapytania SQL poniżej.

```
mariadb> select *, soundex(tekst1), soundex(tekst2), tekst1 sounds like tekst2 from teksty;
+-----------+--------+---------+-----------------+-----------------+--------------------------+
| ID_teksty | tekst1 | tekst2  | soundex(tekst1) | soundex(tekst2) | tekst1 sounds like tekst2 |
+-----------+--------+---------+-----------------+-----------------+--------------------------+
|         1 | Men    | Man     | M000            | M000            |                        1 |
|         2 | Break  | Break   | B620            | B620            |                        1 |
|         3 | Course | Coarse  | C620            | C620            |                        1 |
|         4 | Race   | Raise   | R200            | R200            |                        1 |
|         5 | Bear   | Bare    | B600            | B600            |                        1 |
|         6 | Desert | Dessert | D263            | D263            |                        1 |
|         7 | Price  | Prize   | P620            | P620            |                        1 |
|         8 | Lose   | Loose   | L200            | L200            |                        1 |
|         9 | Plain  | Plane   | P450            | P450            |                        1 |
|        10 | City   | Town    | C300            | T500            |                        0 |
|        11 | Artur  | Arthur  | A636            | A636            |                        1 |
|        12 | Kraków | Cracow  | K620            | C620            |                        0 |
+-----------+--------+---------+-----------------+-----------------+--------------------------+
```

Tabela 2.1. Przykład przedstawiający działanie algorytmu Soundex dla języka angielskiego

Tabela posiada następujące kolumny: ID_teksty – jest to unikalny identyfikator rekordu, tekst1 – pierwszy wyraz do porównania, tekst2 – drugi wyraz do porównania, soundex(tekst1) – wynik algorytmu Soundex dla tekst1, soundex(tekst2) – wynik algorytmu Soundex dla tekst2, tekst1 sounds like tekst2 – wartość logiczna (1 lub 0) wskazująca, czy tekst1 brzmi podobnie do tekst2. Algorytm Soundex przypisuje słowom kod fonetyczny. Jeśli dwa słowa mają ten sam kod, są uznawane za fonetycznie podobne. Na przykład, w wierszu pierwszym "Men" i "Man" mają ten sam kod M000, co oznacza, że brzmią podobnie. Wiersze 10 i 12 pokazują przypadki, gdzie słowa mają różne kody fonetyczne, a więc nie są uznawane za podobne fonetycznie.
Algorytm Soundex znajduje zastosowanie w wielu dziedzinach, np. takich jak:
- wyszukiwanie genealogiczne – umożliwia identyfikowanie nazwisk zapisywanych w różny sposób,

- systemy zarządzania relacjami z klientami (CRM) – ułatwia wyszukiwanie kontaktów na podstawie ich fonetycznego brzmienia, co jest użyteczne w przypadku literówek,

Algorytm Soundex jest ważnym narzędziem w aspekcie poprawy wyszukiwania i analizy danych tekstowych, zwłaszcza w sytuacjach, gdy fonetyczne podobieństwo jest ważniejsze niż dokładne dopasowanie literowe.

2.2. Metody analizy porównawczej zbiorów tekstowych

Analiza porównawcza zbiorów tekstowych polega na badaniu podobieństw i różnic między tekstami w celu zrozumienia struktury, treści, stylu, autora lub innych aspektów tych tekstów. Istnieje wiele metod analizy porównawczej, które mogą być wykorzystane w zależności od potrzeb badawczych. Oto kilka popularnych metod:

- Porównywanie częstotliwości słów: polega na analizie częstotliwości występowania poszczególnych słów lub fraz w różnych tekstach. Można to zrobić, tworząc reprezentację wektorową tekstów i porównując je za pomocą miar podobieństwa, takich jak odległość kosinusowa lub odległość Euklidesa.
- Analiza tematyczna: bazuje na identyfikacji tematów lub kategorii, które przewijają się przez różne teksty. Metoda LDA (ang. *Latent Dirichlet Allocation*)[49] jest jednym z popularnych podejść do modelowania tematów, które pozwala odkryć ukryte tematy w zbiorach tekstowych.
- Analiza stylometryczna: wykorzystuje analizę stylistyczną cech tekstów, takich jak długość słów, częstotliwość poszczególnych słów czy struktura zdań. Można to zrobić, analizując różne cechy tekstowe i porównując je za pomocą statystycznych metod, takich jak analiza skupień czy analiza głównych składowych (PCA).
- Analiza sentymentu: bazuje na na ocenie emocji, tonu czy nastroju wyrażanego w tekstach. Ta analiza może być przeprowadzana na różnych poziomach granularności, od zdania po cały dokument. Można zastosować metody oparte na słownikach, statystyczne modele uczenia maszynowego lub techniki głębokiego uczenia.
- Porównywanie struktury tekstów: wykorzystujące w analizie organizację tekstów , taką jak układ akapitów, sekcji czy rozdziałów. Można to zrobić, stosując metody analizy sieciowej, grafowej lub analizy sekwencji.

- Analiza autorska: polegająca na identyfikacji autorów tekstów na podstawie ich stylu pisania. Można to osiągnąć, analizując różne cechy tekstowe, takie jak zastosowanie słów, struktura zdań czy gramatykę, a następnie stosując metody klasyfikacji, takie jak SVM (ang. *Support Vector Machine*) [50] czy *Naive Bayes* [56].

W analizie porównawczej zbiorów tekstowych można łączyć różne metody i podejścia w zależności od celu badawczego. Ponadto, można wykorzystać techniki przetwarzania języka naturalnego (NLP) oraz uczenia maszynowego, aby usprawnić analizę porównawczą zbiorów tekstowych. Dostępne są różne narzędzia i biblioteki, takie jak NLTK[23], spaCy[24], Gensim[25] czy scikit-learn[26], które ułatwiają implementację tych metod. Ważne jest, aby dostosować analizę porównawczą do konkretnego problemu badawczego, biorąc pod uwagę kontekst, cele i rodzaj danych tekstowych.

Istotnymi algorytmami w analizie porównawczej tekstów są metody poszukiwania wzorca, które służą do lokalizowania wystąpień określonego wzorca (szukanego ciągu znaków) w tekście lub innym ciągu danych. Mają szerokie zastosowanie w różnych dziedzinach informatyki, przetwarzania języka naturalnego, bioinformatyki oraz wyszukiwaniu informacji. Poniżej znajduje się kilka przykładów zastosowań algorytmów poszukiwania wzorca:

- Wyszukiwanie informacji – wyszukiwarki internetowe, takie jak Google, używają algorytmów poszukiwania wzorca, aby odnaleźć strony zawierające słowa kluczowe wpisane przez użytkownika.
- Edytory tekstu – implementują funkcję "znajdź" lub "wyszukaj" w edytorach tekstu, takich jak Microsoft Word czy Notepad++, która wykorzystuje algorytmy poszukiwania wzorca, aby zlokalizować określone słowa lub frazy w dokumencie.
- Wykrywanie plagiatu – systemy wykrywania plagiatu często używają algorytmów poszukiwania wzorca, aby porównać teksty i zidentyfikować fragmenty, które są podejrzanie podobne do innych źródeł.
- Analiza DNA – w bioinformatyce, algorytmy poszukiwania wzorca są używane do analizy sekwencji DNA, RNA oraz białek. Pozwalają one na identyfikację konkretnych

[23] Natural Language Toolkit - https://www.nltk.org/
[24] Industrial-Strength Natural Language Processing - https://spacy.io/
[25] Topic modelling for humans - https://radimrehurek.com/gensim/
[26] scikit-learn Machine Learning in Python - https://scikit-learn.org/stable/

sekwencji nukleotydów czy aminokwasów, które są istotne dla funkcji biologicznych organizmów.

- Wyszukiwanie wzorców w obrazach – algorytmy poszukiwania wzorców są również stosowane w przetwarzaniu obrazów, aby lokalizować konkretne cechy lub obiekty na obrazie. Można je zastosować, na przykład, do wykrywania twarzy na zdjęciach czy lokalizowania znaków drogowych w systemach rozpoznawania obrazów.
- Systemy IDS (ang. *Intrusion Detection System*) – algorytmy poszukiwania wzorców są używane w systemach wykrywania włamań, aby analizować ruch sieciowy i identyfikować podejrzane wzorce, które mogą wskazywać na próbę włamania czy atak na system.
- Wyrażenia regularne – algorytmy poszukiwania wzorców są podstawą działania wyrażeń regularnych, które są szeroko stosowane w programowaniu, przetwarzaniu tekstu i analizie danych do manipulowania i analizowania ciągów znaków.

2.2.1. Algorytm poszukiwania wzorca: Boyera i Moore’a,

Algorytm poszukiwania wzorca Boyera-Moore'a [52] to wydajny algorytm opracowany przez Roberta Boyera i J Strother Moore'a w 1977 roku. Algorytm służy do wyszukiwania wzorca (szukanego ciągu znaków) w tekście. Jest to popularny algorytm w przetwarzaniu tekstów, wyszukiwaniu informacji i bioinformatyce, ze względu na jego szybkość i efektywność w porównaniu do innych algorytmów wyszukiwania wzorca. Algorytm opiera się na dwóch heurystykach, które pozwalają przesuwać wzorzec o więcej niż jedną pozycję w każdym kroku wyszukiwania. Heurystyki te to:

1. Heurystyka złej litery (ang. *bad character heuristic*): Algorytm analizuje wystąpienie niepasującego znaku w teście i przesuwa wzorzec tak, że niepasujący znak w tekście jest dopasowywany do ostatniego wystąpienia tego znaku w wzorcu. Jeśli znak nie występuje w wzorcu, wzorzec jest przesuwany za niepasującym znakiem.
2. Heurystyka dobrego sufiksu (ang. *good suffix heuristic*): Algorytm analizuje już przetworzony sufiks wzorca, który pasuje do tekstu i przesuwa wzorzec, tak aby wystąpienie tego sufiksu wewnątrz wzorca zostanie dopasowane do wystąpienia sufiksu w tekście.

Algorytm Boyera-Moore'a jest szczególnie efektywny, gdy wzorzec jest długi, a alfabet (zbiór możliwych znaków) jest ograniczony. W praktyce jego średnia złożoność obliczeniowa jest znacznie niższa niż liniowa, co czyni go jednym z najszybszych algorytmów wyszukiwania wzorców.

2.2.2. Algorytm poszukiwania wzorca Knutha-Morrisa-Pratta

Algorytm Knutha-Morrisa-Pratta (KMP) to również efektywny algorytm poszukiwania wzorca, który pozwala na znalezienie wystąpień danego wzorca w tekście. Algorytm został opracowany w 1977 roku przez Donalda Knutha, Vaughan'a Pratta i Jamesa Morrisa [53]. Główną zaletą tego algorytmu jest jego liniowa złożoność czasowa, co oznacza, że szybkość przeszukiwania wzorca względem długości tekstu rośnie proporcjonalnie. Algorytm KMP opiera się na obserwacji, że w przypadku wystąpienia częściowego dopasowania wzorca, nie ma potrzeby sprawdzania niektórych znaków ponownie. W celu wykorzystania tej obserwacji, algorytm KMP konstruuje tablicę pomocniczą o nazwie tablica najdłuższego właściwego prefiksu-sufiksu (LPS). Tablica LPS zawiera informacje o najdłuższych prefiksach wzorca, które są jednocześnie jego sufiksami, co pozwala na przesunięcie wzorca o odpowiednią liczbę pozycji bez konieczności ponownego sprawdzania tych samych znaków. Gdy algorytm znajduje dopasowanie, zwraca indeks, w którym wzorzec zaczyna się w tekście. Jeśli występuje wiele wystąpień wzorca, algorytm zwróci indeksy wszystkich dopasowań. Algorytm jest szczególnie przydatny w przypadkach, gdy wzorzec zawiera powtarzające się znaki, ponieważ pozwala na przesunięcie wzorca o większą liczbę pozycji i uniknięcie niepotrzebnego porównywania tych samych znaków. Dzięki swojej wydajności, algorytm KMP jest często stosowany w różnych dziedzinach informatyki, takich jak przetwarzanie języka naturalnego, wyszukiwanie informacji czy analiza danych.

2.2.3. Algorytm poszukiwania wzorca Karpa-Rabina

Algorytm Karpa-Rabina to kolejny efektywny algorytm poszukiwania wzorca, który został opracowany przez Richarda Karpa i Michaela Rabina w 1987 roku [54]. Jego główną cechą jest wykorzystanie techniki "rolling hash" (przesuwającego się haszowania), co pozwala na sprawne porównywanie wartości haszujących wzorca i fragmentu tekstu o tej samej długości. Jeśli algorytm znajdzie dopasowanie, zwraca indeks, w którym wzorzec zaczyna się w tekście. Jeśli występuje wiele wystąpień wzorca, algorytm zwróci indeksy wszystkich dopasowań. Algorytm Karpa-Rabina osiąga średnio liniową złożoność czasową, ale w przypadku wystąpienia kolizji haszy może być wolniejszy. Mimo to jest stosowany w różnych dziedzinach, takich jak przetwarzanie języka naturalnego, wyszukiwanie informacji czy analiza danych. Ze względu na jego szybkość i prostotę, algorytm ten jest szczególnie przydatny w przypadkach, gdy wymagane jest przeszukiwanie dużych zbiorów tekstów lub szybkie wyszukiwanie wzorca w tekście. Jest on często stosowany w połączeniu z innymi algorytmami poszukiwania wzorca, aby zrównoważyć odpowiednio ich wady i zalety.

2.2.4. Współczynniki Dice, Jaccarda, odległość kosinusowa

Oszacowanie wartości podobieństwa dla określonego modelu wektorowego jest odrębnym zagadnieniem. Współczynniki Dice'a i Jaccarda [14,55] są miarami podobieństwa, które mogą być stosowane do porównywania zbiorów danych, takich jak dokumenty tekstowe. Chociaż są to inne miary niż odległość kosinusowa [57], można je stosować do podobnych celów, jak ocena podobieństwa między tekstami. Współczynnik Dice'a (ang. *Dice coefficient*) jest to miara podobieństwa, która oblicza stosunek podwójnej liczby wspólnych elementów między dwoma zbiorami, do sumy liczebności tych zbiorów. Wzór na współczynnik Dice'a dla zbiorów A i B jest następujący:

$$\mathrm{Dice\_dist}(d_i, d_k) = 2\frac{\sum_{j=1}^{n} d_{ij} d_{kj}}{\sum_{j=1}^{n} d_{ij}^2 \sum_{j=1}^{n} d_{kj}^2} \tag{2.5}$$

gdzie:

d_i i d_k to dwa wektory (dokumenty) porównywane,

d_{ij} i d_{kj} to j-te elementy wektorów d_i i d_k,

n to liczba elementów w każdym wektorze.

W kontekście analizy tekstowej, zbiory A i B mogą reprezentować zbiory słów występujących w dwóch różnych dokumentach.

Zalety współczynnika Dice'a:

- Odporność na różnice wielkości zbiorów: Współczynnik Dice'a uwzględnia różnice wielkości zbiorów, dzięki czemu może być stosowany do porównywania zbiorów o różnych rozmiarach.
- Skalowalność: Współczynnik Dice'a można łatwo obliczyć dla dużych zbiorów danych, co czyni go skalowalnym rozwiązaniem w przypadku analizy tekstu czy innych zastosowań.

Wady współczynnika Dice'a:

- Wrażliwość na małe różnice: Współczynnik Dice'a może być wrażliwy na małe różnice między zbiorami, zwłaszcza gdy zbiory mają niewiele wspólnych elementów. W takich przypadkach może przeszacować różnice między zbiorami.
- Nie uwzględnia kolejności elementów: Współczynnik Dice'a nie uwzględnia kolejności występowania elementów w zbiorach, co może prowadzić do błędów w ocenie podobieństwa, gdy kolejność elementów ma znaczenie.
- Może dawać wyniki nieintuicyjne: W niektórych przypadkach, współczynnik Dice'a może dawać wyniki, które są trudne do zinterpretowania lub nie odzwierciedlają rzeczywistego podobieństwa między zbiorami.

Współczynnik Jaccarda (ang. *Jaccard coefficient*) to miara podobieństwa, która oblicza stosunek liczby wspólnych elementów między dwoma zbiorami, do liczby elementów w ich sumie (suma zbiorów bez powtórzeń). Wzór na współczynnik Jaccarda dla zbiorów A i B jest następujący:

$$\text{Jaccard\_dist}(d_i, d_k) = \frac{\sum_{j=1}^{n} d_{ij} d_{kj}}{\sum_{j=1}^{n} d_{ij}^2 + \sum_{j=1}^{n} d_{kj}^2 - \sum_{j=1}^{n} d_{ij} d_{kj}} \tag{2.6}$$

Podobnie jak w przypadku współczynnika Dice'a, w analizie tekstowej zbiory A i B mogą reprezentować zbiory słów występujących w dwóch różnych dokumentach. Oba współczynniki umożliwiają obliczenie zgodności pomiędzy cechami wektora.

Zalety współczynnika Jaccarda:

- Odporność na różnice wielkości zbiorów: Współczynnik Jaccarda jest mniej wrażliwy na różnice wielkości zbiorów niż niektóre inne miary podobieństwa, dzięki czemu może być stosowany do porównywania zbiorów o różnych rozmiarach.
- Skalowalność: Współczynnik Jaccarda można łatwo obliczyć dla dużych zbiorów danych, co czyni go skalowalnym rozwiązaniem w przypadku analizy tekstu czy innych zastosowań.

Wady współczynnika Jaccarda:

- Wrażliwość na małe różnice: Współczynnik Jaccarda podobnie jak Dice’a, może być wrażliwy na małe różnice między zbiorami, zwłaszcza, gdy zbiory mają niewiele wspólnych elementów. W takich przypadkach może przeszacować różnice między zbiorami.
- Nie uwzględnia kolejności elementów: Współczynnik Jaccarda nie uwzględnia kolejności występowania elementów w zbiorach, co może prowadzić do błędów w ocenie podobieństwa, gdy kolejność elementów ma znaczenie.
- Może dawać wyniki nieintuicyjne: W niektórych przypadkach, współczynnik Jaccarda może dawać wyniki, które są trudne do zinterpretowania lub nie odzwierciedlają rzeczywistego podobieństwa między zbiorami.

W aspekcie odległości kosinusowej, współczynniki Dice'a i Jaccarda mogą być wykorzystane jako alternatywne metody oceny podobieństwa między dokumentami tekstowymi.

Odległość kosinusowa oblicza kąt między dwoma wektorami w przestrzeni wielowymiarowej (na przykład wektorami cech reprezentującymi dokumenty tekstowe), podczas gdy współczynniki Dice'a i Jaccarda porównują zbiory słów w tych dokumentach. Każda z tych miar ma swoje zalety i wady, a wybór odpowiedniej metody zależy od konkretnego problemu i rodzaju danych.

Wzór na najbardziej powszechną metodę w dziedzinie analizy podobieństwa dokumentów, czyli odległość kosinusową[27] jest następujący:

$$\text{Cosine\_dist}(d_i, d_k) = \frac{\sum_{j=1}^{n} d_{ij} d_{kj}}{\sqrt{\sum_{j=1}^{n} d_{ij}^2} \sqrt{\sum_{j=1}^{n} d_{kj}^2}} \tag{2.7}$$

Zalety odległości kosinusowej:

- Unormowana: Odległość kosinusowa uwzględnia kierunek wektorów, a nie ich długość. Dzięki temu, jest mniej wrażliwa na różnice w długości dokumentów czy liczbie wystąpień słów.
- Odporna na rzadkie dane: W przypadku analizy tekstu, macierze wektorów często są rzadkie (większość wartości to zera), co oznacza, że większość słów występuje tylko w niewielu dokumentach. Odległość kosinusowa dobrze radzi sobie z porównywaniem rzadkich wektorów.
- Szybka i skalowalna: Odległość kosinusowa jest stosunkowo szybkim algorytmem, co pozwala na efektywne porównywanie dużych zbiorów danych.

Wady odległości kosinusowej:

- Brak uwzględnienia częstości: Odległość kosinusowa może nie uwzględniać wystarczająco różnic w częstości występowania słów, co może prowadzić do przeszacowania podobieństwa między dokumentami o różnych długościach.
- Nie uwzględnia kolejności słów: Podobieństwo kosinusowe nie uwzględnia kolejności występowania słów w tekście, co może prowadzić do błędów w ocenie podobieństwa, gdy kolejność słów ma znaczenie.
- Ograniczona interpretowalność: Wartość podobieństwa kosinusowego może być trudna do zinterpretowania w kontekście rzeczywistym, ponieważ nie daje bezpośredniej informacji o liczbie wspólnych cech czy różnic między porównywanymi wektorami.

[27] http://nlp.stanford.edu/IR-book/pdf/06vect.pdf, s. 122

2.2.5. Odległość Levenshteina

Odległości Levenshteina (ang. *Levenshtein distance*) [9] *k* pomiędzy dwoma ciągami tekstowymi jest to najmniejsza liczba operacji: wstawiania znaku, usunięcia znaku lub wymiany znaku na inny w jednym z ciągów, umożliwiających zmianę jednego ciągu w drugi. Odległość Levenshteina jest uogólnieniem odległości Hamminga (ang. *Hamming distance*). W przeciwieństwie do odległości Hamminga, odległość Levenshteina umożliwia porównywanie ciągów o różnej długości, co bezpośrednio przekłada się na liczne zastosowania algorytmu, m.in. w korekcie pisowni, maszynowym tłumaczeniu tekstów czy eksploracji danych tekstowych, a szerzej w mechanizmach wyszukiwarek internetowych, procesorach tekstów itp.

Odległość Levenshteina *k* jest równa ostatniemu elementowi macierzy Levenshteina ***D[m,n]***, co można ująć w poniższą formułę:

$$k = \mathbf{D}[m, n] = \text{LevenshteinDistance}(\mathbf{Text1}, \mathbf{Text2}) \tag{2.8}$$

Główna idea algorytmu odległości edycyjnej Levenshtein (funkcji *LevenshteinDistance*) opisuje poniższa formuła:

$$\begin{cases} \prod_{i=1}^{n}\prod_{j=1}^{m} \mathbf{D}[i,j] = \min(\mathbf{D}[i-1,j]+1, \mathbf{D}[i,j-1]+1, \mathbf{D}[i-1,j-1]+cost) \\ cost = 0 : \mathbf{Text1}[i] \equiv \mathbf{Text2}[j] \\ cost = 1 : \mathbf{Text1}[i] \neq \mathbf{Text2}[j] \\ \mathbf{D}[i,0] = i \\ \mathbf{D}[0,j] = j \\ \mathbf{D}[0,0] = 0 \end{cases} \tag{2.9}$$

gdzie:

D – macierz Levenshteina o rozmiarach *n*+1, *m*+1, utworzona na bazie dwóch ciągów tekstowych: ***Text1*** i ***Text2***,

m, *n* – długości dwóch analizowanych ciągów (tj. liczby znaków w poszczególnych ciągach),

D*[i,j]* - (i,j) – element macierzy o współrzędnych *i* oraz *j*,

min – funkcja obliczająca wartość minimalnych z trzech podanych wartości całkowitych,

cost – zmienna przybierająca wartości 0 lub 1.

Algorytm odległości Levenshteina opisuje poniższy pseudo-kod (2.10):

(2.10)

```
input variables: char Text1[0..m-1], char Text2[0..n-1]
     declare: int D[0..m, 0..n]
     for i from 0 to m
      D[i, 0] := i
         for j from 0 to n
          D[0, j] := j

         for i from 1 to m
          for j from 1 to n

           if char of Text1 at (i – 1) = char of Text2 at (j – 1) then
                        cost := 0 else cost := 1
           end if
                        D[i, j] :=
                            min(D[i - 1, j] + 1,
                            D[i, j - 1] + 1,
                            D[i - 1, j - 1] + cost)
           end for (variable j)
          end for (variable i)

return D[m, n];
```

Tabela 2.2 oraz pseudo-kod 2.10 pokazują, że dany element *[i, j]* macierzy **D** jest obliczany dla każdej iteracji na podstawie elementów: *D[i-1,j]*, *D[i,j-1]* oraz *D[i-1,j-1]*. To oznacza, że każdy z tych elementów musi zostać obliczony we wcześniejszych iteracjach z wyjątkiem wartości początkowych (tj. uzupełnionych w komórkach pierwszego wiersza i pierwszej kolumny).

		y	e	s	t	e	R	d	a	y
	0	1	2	3	4	5	6	7	8	9
t	1	**1**	2	3	3	4	5	6	7	8
o	2	2	2	3	4	4	5	6	7	8
m	3	3	3	3	4	5	5	6	7	8
o	4	4	4	4	4	5	6	6	7	8
r	5	5	5	5	5	5	5	6	7	8
r	6	6	6	6	6	**6**	5	6	7	8
o	7	7	7	7	7	7	6	6	7	8
w	8	8	8	8	8	8	7	7	7	**8**

Tabela 2.2. Macierz Levenshteina D, skonstruowana na podstawie wyrazów: *yesterday* i *tomorrow*[28]

[28] Link do programu generującego macierz Levenshteina (autor: Artur Niewiarowski): https://cloud.mck.pk.edu.pl/index.php/s/7cSdPqLQMFmTwkZ

Kolejne przykłady zestawiono w tabeli poniżej (2.3).

Lp.	Ciąg nr 1	Ciąg nr 2	Odległość Levenshteina
1	Dog	Dogs	1
2	University	Universities	3
3	Tom is writing a letter	Tom is writin letters	4
4	Cat	Cat	0

Tabela 2.3. Przykłady analizy dwóch ciągów tekstowych algorytmem odległości Levenshteina

W pierwszym przypadku należy dodać jeden znak w pierwszym ciągu lub usunąć jeden znak w drugim, aby te dwa ciągi były identyczne. W drugim wierszu w pierwszym ciągu należy wymienić jeden znak na drugi i dodać dwa znaki lub w drugim ciągu zamienić jeden znak i usunąć dwa znaki, w celu osiągnięcia identyczności analizowanych ciągów. W trzecim wierszu tabeli należy usunąć znaki: „g", „a", „s" i spację w ciągu pierwszym lub dodać cztery znaki w drugim ciągu. W ostatnim przykładzie odległość Levenshteina jest równa 0, ponieważ ciągi są identyczne i nie trzeba wykonywać na nich żadnych operacji.

2.3. Popularne implementacje mechanizmów analizy danych tekstowych w bazach danych

Systemy zarządzania bazami danych są obecnie nieodzownym środowiskiem większych platform przechowujących i zarządzających danymi, tj. systemów bankowych, portali internetowych, portali społecznościowych, systemów typu wirtualny dziekanat, itp. Niezależnie od typu bazy danych[29] systemy te umożliwiają m.in.: łatwość składowania dużych ilości danych, przejrzystość danych dzięki odpowiednim strukturom (np. tabelom w relacyjnych baza danych), szybki dostęp do danych, współdzielenie danych przez wielu użytkowników, mechanizmy zapobiegające utracie danych w razie awarii (np. transakcyjność, replikacja danych), jak również dostarczają licznych specjalistycznych funkcji. Mowa tutaj o funkcjach zarówno matematycznych, jak również związanych z danymi tekstowymi (tudzież wspomniany algorytm Soundex oraz wiele innych funkcji umożliwiających modyfikację ciągów tekstowych, np. trim, replace, upper, substring, oraz wielu innych[30]). Natomiast najważniejszym pod względem funkcjonalności i zastosowania algorytmem operującym na danych tekstowych jest algorytm Full-Text Index[31]. Jego zastosowanie to poszukiwanie

[29] tj. relacyjnej, obiektowej, relacyjno-obiektowej, nierelacyjnej – tutaj popularne w ostatnich latach NoSQL, BigData,

[30] Manual szbd MariaDB: https://mariadb.com/kb/en/string-functions/

[31] https://mariadb.com/kb/en/full-text-index-overview/

dokumentów lub ciągów tekstowych na podstawie wprowadzonych fragmentów ciągów z dodatkowymi kryteriami. Na podstawie tego mechanizmu powstał osobny rodzaj baz danych - Full-Text Database[32] rozwijany już od lat siedemdziesiątych ubiegłego wieku m.in. przez IBM. Zadaniem tych baz danych jest przechowywanie pełnych tekstów: książek[33], czasopism[34], gazet[35], prac naukowych[36], które można przeszukiwać z odpowiednimi kryteriami.

Sposób działania algorytmu na przykładowych danych zostanie opisany na podstawie funkcjonalności popularnego systemu zarządzania bazą danych MariaDB[37]. W tym systemie rodzaje przeszukiwania zindeksowanych rekordów są następujące: *in natural language mode*, *in boolean mode, with query expansion*.

Pierwszy typ jest domyślny i jego główną cechą jest brak obsługi operatorów specjalnych, a fragmenty ciągów tekstowych są oddzielone przecinkami i muszą pokrywać się dokładnie z tekstem przeszukiwanym.

```
mariadb> select tytul_pracy from prace_dyplomowe_i where match(tytul_pracy)
against ('tabel, tabela, tabelka, tabeli');
    +-----------------------------------------------------------------------------------------+
    | tytul_pracy                                                                             |
    +-----------------------------------------------------------------------------------------+
    | Przegląd metod podziału tabeli na partycje                                              |
    | Optymalizacja w aspekcie partycjonowania wersjonowanych tabel w bazie danych MariaDB |
    +-----------------------------------------------------------------------------------------+
```

Tabela 2.4. Wynik użycia indeksowania Full-Text metodą przeszukiwania *in natural language mode*

Typ przeszukiwania *in boolean mode* umożliwia analizę danych tekstowych z użyciem operatorów takich jak: „+” – słowo musi istnieć w każdym zwróconym rekordzie, „-” – słowo nie może istnieć w zwróconym rekordzie, „>” – większa ważność słowa po znaku, „<” – mniejsza ważność słowa po znaku, „"” – ujęcie w cudzysłów „wielu słów” traktowane jest jako jeden termin, „*” – zastępuje zero lub więcej znaków (umieszczany na końcu terminu).

[32] Powiązane projekty: Apache Solr, ArangoSearch, BaseX, Elasticsearch, KinoSearch, Lemur/Indri, Lucene, mnoGoSearch, PostgreSQL, Searchdaimon, Sphinx, Swish-e, Terrier IR Platform, Xapian, Bsasearch, RediSearch

[33] Np. JSTOR (https://www.jstor.org/)

[34] Np. Academic Search Complete (https://www.ebsco.com/products/research-databases/academic-search-complete)

[35] Np. Lexis-Nexis (https://www.lexisnexis.com/en-us/home.page)

[36] Np. Google Scholar (https://scholar.google.com/)

[37] https://mariadb.org/

```
mariadb> select match(tytul_pracy) against
('dany* problem' in boolean mode) i, tytul_pracy from prace_dyplomowe_i order by i desc;
+----------------------+-----------------------------------------------------------------------------------------+
| i                    | tytul_pracy                                                                             |
+----------------------+-----------------------------------------------------------------------------------------+
|   0.23287785053253174 | Problem redundancji danych w bazie danych MariaDB                                      |
|  0.005233161151409149 | Replikacja danych w bazie danych MariaDB                                               |
| 0.0026165805757045746 | Optymalizacja w aspekcie partycjonowania wersjonowanych tabel w bazie danych MariaDB |
| 0.0026165805757045746 | Praktyczne zastosowania rekurencji w zapytaniach SQL w bazie danych                    |
|                     0 | Przegląd metod podziału tabeli na partycje                                             |
+----------------------+-----------------------------------------------------------------------------------------+
```

Tabela 2.5. Wynik użycia indeksowania Full-Text metodą przeszukiwania *in boolean mode* z opcją gwiazdki

```
mariadb> select match(tytul_pracy) against
('dany* -problem -sql' in boolean mode) i, tytul_pracy from prace_dyplomowe_i order by i desc;
+----------------------+-----------------------------------------------------------------------------------------+
| i                    | tytul_pracy                                                                             |
+----------------------+-----------------------------------------------------------------------------------------+
|  0.005233161151409149 | Replikacja danych w bazie danych MariaDB                                               |
| 0.0026165805757045746 | Optymalizacja w aspekcie partycjonowania wersjonowanych tabel w bazie danych MariaDB |
|                     0 | Problem redundancji danych w bazie danych MariaDB                                      |
|                     0 | Praktyczne zastosowania rekurencji w zapytaniach SQL w bazie danych                    |
|                     0 | Przegląd metod podziału tabeli na partycje                                             |
+----------------------+-----------------------------------------------------------------------------------------+
```

Tabela 2.6. Wynik użycia indeksowania Full-Text metodą przeszukiwania *in boolean mode* z opcją gwiazdki i znakiem minus

Powyższe przykłady przedstawiają działanie indeksowania Full-Text w trybie *in boolean mode* z dodatkowymi opcjami, które dla tego trybu są dostępne. Dodatkowo poprzez umieszczenie dyrektywy *match against* między słowami kluczowymi SQL *select* i *from* można zaobserwować ranking ciągów tekstowych, dodatkowo posortowany (polecenie *order by desc*) od najbardziej trafnego do najmniej.

Ostatnim trybem indeksowania jest tryb *with query expansion*, który jest rozszerzeniem *in language mode*. Składa się z dwóch kroków: w ramach pierwszego kroku wyszukiwanie przebiega zwyczajnie jak w trybie *in natural language mode*, natomiast terminy z najlepiej punktowanych rekordów są umieszczane ponownie w ciągu wyszukiwania i zwracany jest ostatecznie drugi wynik wyszukiwania – w ramach drugiego kroku. Poniższy przykład bazujący na danych przedstawia zasadę działania trybu.

```
mariadb> select match(tytul_pracy)
against ('MariaDB' with query expansion) i, tytul_pracy from prace_dyplomowe_i order by i desc;
+--------------------+-----------------------------------------------------------------------------------------+
| i                  | tytul_pracy                                                                             |
+--------------------+-----------------------------------------------------------------------------------------+
|  1.3300366401672363 | Optymalizacja w aspekcie partycjonowania wersjonowanych tabel w bazie danych MariaDB |
|  0.6497191190719604 | Problem redundancji danych w bazie danych MariaDB                                       |
| 0.42207443714141846 | Replikacja danych w bazie danych MariaDB                                                |
| 0.12664911150932312 | Praktyczne zastosowania rekurencji w zapytaniach SQL w bazie danych                     |
|                   0 | Przegląd metod podziału tabeli na partycje                                              |
+--------------------+-----------------------------------------------------------------------------------------+
```

Tabela 2.7. Wynik użycia indeksowania Full-Text metodą przeszukiwania *with query expansion*

Jak widać na powyższym przykładzie, czwarty rekord zwróconego wyniku nie zawiera terminu „MariaDB”, a pomimo tego, jego indeks nie jest zerowy. Wynika to ze skojarzenia słów „bazie danych” (krok nr 2) ze słowem „MariaDB” z kroku nr 1 algorytmu.

Powyższe przykłady związane z implementacją w bazach danych metod związanych z analizą danych tekstowych pokazują jakie istotne zastosowanie mogą znaleźć – np. przeszukiwanie zawartości stron internetowych, portali, baz zawierających teksty.

3. Implementacja algorytmu odległości edycyjnej w analizie podobieństwa zbiorów tekstowych

Koncepcja budowy mechanizmu analizującego podobieństwo pomiędzy ciągami tekstowymi będąca tematem pracy będzie wykorzystywała przedstawiony w poprzednim rozdziale algorytm odległości edycyjnej Levenshteina [9]. Wprowadzenie tego algorytmu nadaje pewien stopień uniwersalności językowej wobec badanych dokumentów dzięki zastosowaniu formuły zaproponowanej poniżej [10]. Meritum rozdziału poprzedza w podrozdziale 3.2. propozycja zastosowania formuły do analizy krótkich zdań składających się z wyrazów (zamiast znaków w rozdz. 3.1.) [13], w tym problemów i ograniczeń algorytmu, który był pierwszą próbą zbudowania metody uniwersalnej językowo do analizy ciągów tekstowych. Znalazła ona zastosowanie m.in. w mechanizmie identyfikacji i klasyfikacji treści, który został przedstawiony przez autora w publikacji [12]. W dalszej części tego rozdziału zaproponuję ujęcie korelacji podobieństwa dwóch dokumentów tekstowych w postaci macierzy zero-jedynkowej, odpowiednio bez zastosowania formuły bazującej na algorytmie Levenshteina (rozdz. 3.3.) oraz z jej implementacją (rozdz. 3.4.).

3.1. Miara podobieństwa ciągów na bazie algorytmu edycyjnej Levenshteina

Odległość Levenshteina można wykorzystać do określenia podobieństwa pomiędzy ciągami tekstowymi. Podobieństwo będzie zawierało się w przedziale domkniętym od 0 do 1. Miara podobieństwa (*p*) jest przedstawiona według poniższego wzoru:

$$p = 1 - \left(\frac{k}{k_{\max}}\right);\ k_{\max} = \max(n, m),\quad \begin{matrix} k \geq 0, m > 0, n > 0 \\ p \in \langle 0,1 \rangle \end{matrix} \tag{3.1}$$

gdzie:

ułamek w nawiasie – będący ilorazem liczby *k* operacji (odległości Levenshteina) do wartości k_{max} liczby elementów dłuższego ciągu (wartości pesymistycznej, czyli przypadku, w którym oba ciągi byłyby całkowicie różne) – wyznacza wartość oznaczającą różność ciągów w przedziale <0,1>,

n, m – długości badanych ciągów tekstowych (liczba znaków).

Poniższa tabela przedstawia przykłady porównania dwóch ciągów tekstowych z użyciem miary podobieństwa bazującej na algorytmie odległości edycyjnej.

Lp.	**Ciąg tekstowy nr 1**	**Ciąg tekstowy nr 2**	***k***	k_{max}	***p***
1	Dog	Dogs	1	4	0,75
2	University	Universitie**r**	3	12	0,75
3	Tom is writing a letter	Tom is **writin** letter**r**	4	23	0,82
4	World	World	0	5	1
5	Dog	Cat	3	3	0
6	XXXX	XX**YY**	2	4	0,5
7	Pięćdziesięciopięciogroszówka	Pi**ec**dziesi**e**ciopi**e**ciogrosz**o**wka	5	29	0,83
8	Dwudziestopięciolatek	Dwudziestopi**e**ciolatek	1	22	0,95
9	компьютер	Компьютеры	1	11	0,90

Tabela 3.1 . Przykłady porównania dwóch ciągów tekstowych

O ile odległość edycyjna (*k*) informuje nas o liczbie zmian jaką należy wykonać na jednym ciągu, aby był taki sam jak drugi, to nie daje informacji, jak bardzo te ciągi są do siebie podobne procentowo. Przez to nie nadaje się bezpośrednio do użycia w algorytmach analizujących podobieństwo ciągów tekstowych uwzględniających pochodzenie badanych wyrazów, czy też rdzeń. Przykładowo, odległość Levenshteina równa 3 nie daje wiedzy, czy ciągi są całkowicie różne jak w przykładzie 5 tabeli, czy też posiadają prawdopodobnie to same pochodzenie jak w wierszu 2 tabeli.

Taką informację uzyskujemy dopiero wprowadzając miarę podobieństwa (*p*). Tak więc, ciągi, które są całkowicie różne od siebie, czyli takie, w których należy wymienić/usunąć wszystkie litery w dłuższym z ciągów mają odległość edycyjną równą najdłuższemu ciągowi, czyli ich podobieństwo będzie wynosiło 0 (0%). Z kolei odległość edycyjna pomiędzy takimi samymi ciągami będzie równa 0, czyli podobieństwo będzie wynosiło 1 (100%). Metoda ta nadaje się z powodzeniem w większości przypadków do wykrywania podobieństw pomiędzy wyrazami, w tym m.in. zawierających błędy ortograficzne, tzw. literówki, czy też pomiędzy niektórymi formami odmiany, jak również oszacowaniem wspólnego rdzenia.

3.2. Koncepcja implementacji odległości edycyjnej Levenshteina w analizie podobieństwa zdań

W poszukiwaniu uniwersalności w analizie podobieństwa zdań będącej założonym celem niniejszej pracy, można podjąć próbę wykorzystania zasady działania algorytmu odległości edycyjnej Levenshteina do analizy zdań, gdzie rolę znaków przejmują wyrazy, a następnie wprowadzić miarę podobieństwa zaproponowaną w poprzednim rozdziale.
Tę koncepcję przedstawia poniższa reguła (3.2).

$$k_T = \mathop{\mathrm{I}}_{i_T=1}^{n_T} \mathop{\mathrm{I}}_{j_T=1}^{m_T} \mathbf{D_T}(i_T, j_T) = \min(\mathbf{D_T}(i_T - 1, j_T) + 1, \mathbf{D_T}(i_T, j_T - 1) + 1, \mathbf{D_T}(i_T - 1, j_T - 1) + \beta_T)$$

$$\begin{cases} \beta_T = 0 : \mathbf{a_T}(i_T) \equiv \mathbf{b_T}(j_T) \\ \beta_T = 1 : \mathbf{a_T}(i_T) \neq \mathbf{b_T}(j_T) \\ \mathbf{D_T}(i_T, 0) = i_T \\ \mathbf{D_T}(0, j_T) = j_T \\ \mathbf{D_T}(0,0) = 0 \end{cases} \tag{3.2}$$

gdzie:

I – znak oznaczający iterację za pomocą pętli for,

k_T - odległość Levenshteina dwóch badanych zdań,

$\mathbf{D_T}$ – macierz utworzona z dwóch porównywanych tekstów o rozmiarach wynikających z liczby wyrazów w poszczególnych tekstach,

$\mathbf{D_T}(i_T, j_T)$ – (i_T, j_T)-ty element macierzy $\mathbf{D_T}$,

β_T – zmienna przyjmująca odpowiednio wartości 0 lub 1,

$\boldsymbol{a_T}(i_T)$ – i_T-ty wyraz tekstu a_T,

$\boldsymbol{b_T}(j_T)$ – j_T-ty wyraz tekstu b_T,

n_T, m_T – długości badanych tekstów (tj. liczby wyrazów je tworzących).

W powyższym zapisie zauważalne jest podobieństwo pomiędzy bazową formułą (2.9) przedstawiającą zasadę działania algorytmu Levenshteina, a jego proponowaną modyfikacją (3.2). Wynika to z tego, że w tym ostatnim przypadku, macierz odległości edycyjnej dla algorytmu Levenshteina wypełniana jest odpowiednimi wartościami analizowanych wyrazów dla zdań, a nie znaków dla wyrazów.
Dla następującego przykładu, macierz Levenshteina według tej formuły (3.2) wyglądałaby następująco:

		Mateusz	**ma**	**dom**	**oraz**	**samochód**
	0	1	2	3	4	5
Mateusz	1	0	1	2	3	4
ma	2	1	0	1	2	3
domek	3	2	1	1	2	3
i	4	3	2	2	2	3
samochód	5	4	3	3	3	**2**

Tabela 3.2. Przykład porównania dwóch krótkich zdań niosących to samo przesłanie: *„Mateusz ma dom oraz samochód”* oraz *„Mateusz ma domek i samochód”*.

Odległość Levenshteina dla powyższych zdań wynosi: 2, w tym obliczona miara podobieństwa P=0,6.

Przedstawiony przykład uwzględniający algorytm Levenshteina jako bazę dla metody porównania zdań można dowolnie modyfikować w ramach wstępnej analizy uwzględniając omówione w poprzednim rozdziale algorytmy stemmingu i lematyzacji. Wtedy macierz Levenshteina na bazie tych samych zdań mogłaby wyglądać następująco (tabela 3.3).

		Mateusz	**posiadać**	**dom**	**i**	**pojazd**
	0	1	2	3	4	5
Mateusz	1	0	1	2	3	4
posiadać	2	1	0	1	2	3
dom	3	2	1	0	1	2
i	4	3	2	1	0	1
pojazd	5	4	3	2	1	**0**

Tabela 3.3. Przykład porównania dwóch krótkich zdań z przykładu poprzedniego sprowadzonych do formy podstawowej

Część wyrazów została sprowadzona do ich odpowiedników w formie podstawowej i poprzez ten zabieg będący efektem działania algorytmów stemmingu i lematyzacji – odległość Levenshteina wynosi 0. Oznacza to, że zdania są takie same.

Zaprezentowaną koncepcję analizy zdań algorytmem odległości Levenshteina można rozbudować o implementację opisanego podobieństwa ciągów (w tym przypadku wyrazów w zdaniach).

Jeżeli w formule (3.2) w miejscu ustalenia wartości zmiennej β_T dla poszczególnych iteracji, umieścimy zamiast znaków porównania i różności pomiędzy wartościami elementów ciągów a_T oraz b_T, formułę podobieństwa ciągów p opisaną w podrozdziale 3.1., to osiągniemy pewien stopień uniwersalności względem języka, w celu wykrywania podobnych ciągów tekstowych.

Taki zabieg umożliwi również podwyższenie stopnia odporności algorytmu na błędy ortograficzne pomiędzy badanymi wyrazami.

Po uwzględnieniu tej modyfikacji algorytm przyjmie następującą postać (3.3)[38]:

$$k_T = \prod_{i_T=1}^{n_T} \prod_{j_T=1}^{m_T} \mathbf{D_T}(i_T, j_T) = \min(\mathbf{D_T}(i_T - 1, j_T) + 1, \mathbf{D_T}(i_T, j_T - 1) + 1, \mathbf{D_T}(i_T - 1, j_T - 1) + \beta_T)$$

$$\begin{cases} \beta_T = 0 : \mathrm{e}(\mathbf{a_T}(i_T), \mathbf{b_T}(j_T)) >= q \\ \beta_T = 1 : \mathrm{e}(\mathbf{a_T}(i_T), \mathbf{b_T}(j_T)) < q \\ \mathbf{D_T}(i_T, 0) = i_T \\ \mathbf{D_T}(0, j_T) = j_T \\ \mathbf{D_T}(0,0) = 0 \end{cases} \tag{3.3}$$

gdzie:

e jest funkcja obliczająca podobieństwo *p* wyrazów (ciągów tekstowych) w badanych zdaniach, natomiast *bp* – (ang. *acceptable boundary value of similarity measure parameterized by the user*) reprezentuje wartość progową określaną przez użytkownika algorytmu (programu komputerowego analizującego ciągi). Jest to kwalifikator dla porównywanych ciągów tekstowych: $a_{T(iT)}$ oraz $b_{T(jT)}$ określający granicę uznawalności ich za ciągi tożsame – im wyższy tym bardziej restrykcyjny, tzn. ciągi muszą być bardziej podobne (liczba operacji Levenshteina bliższa 0).

Wprowadzenie zmiennej *bp* w algorytmie jest istotne i związane ze specyfiką tekstów pochodzących z różnych dziedzin wiedzy. Przykładowo, słownictwo z dziedziny nauk technicznych jest bardziej konkretne niż humanistycznych, ponieważ definicje/twierdzenia matematyczne mogą wymagać bardziej restrykcyjnych form, zwłaszcza, gdy zawierają sformułowania będące określeniem stałych lub zmiennych. Specyfika tekstów, to również długość wyrazów charakterystyczna dla różnych języków - np. język niemiecki jest znany z długich wyrazów[39], poprzez łączenie wielu członów w wyraz, przykładowo: „*der Schreibtisch*" (pol. *biurko*), powstało z wyrazów „*schreiben*" – (pol. *pisać*) i „*der Tisch*" – (pol. *stół*). Przy czym zarówno „*Schreibtisch"* nie oznacza to samo co „*schreiben"*, jak również „*Schreibtisch"* to

[38] Link do programu umożliwiającego obliczenie podobieństwa pomiędzy zdaniami według opisanej koncepcji (autor: Artur Niewiarowski) - https://cloud.mck.pk.edu.pl/index.php/s/veIaIGFt2n4HwER

[39] do niedawna najdłuższym niemieckim wyrazem był ciąg składający się z 63 liter – *„das Rindfleischetikettierungsüberwachungsaufgabenübertragungsgesetz"* (pol. *Ustawa o przekazywaniu zadań w zakresie nadzoru nad etykietowaniem wołowiny*)

samo co „*Tisch*”. Przykład w tabeli 3.4 pokazuje jak zmiana parametru *bp* wpływa na rezultat uznania słów za takie same (β_T).

Lp.	**Wyraz nr 1**	**Wyraz nr 2**	***bp***	β_T
1	*schreiben*	*Schreibtisch*	0-0,58	1
2	*schreiben*	*Schreibtisch*	0,59-1	0

Tabela 3.4. Przykład wpływu dopasowania wartości progowej *bp* na zmienną β_T

Innym przykładem dopasowania wartości *bp* względem języka może być zamiana liczby pojedynczej na liczbę mnogą w danym języku. Nieznacznie inna będzie dla języka polskiego, a inna dla języka angielskiego, czy niemieckiego.

Lp.	**Wyraz nr 1**	**Wyraz nr 2**	***bp***	β_T
1	*car*	*cars*	0-0,75	1
2	*cat*	*cats*	0-0,75	1
3	*samochód*	*samochody*	0-0,78	1
4	*pies*	*psy*	0-0,75	1
5	*Auto*	*Autos*	0-0,80	1
6	*Katze*	*Katzen*	0-0,83	1

Tabela 3.5. Przykład dopasowania zmiennej *bp* względem przykładów porównania krótkich fragmentów tekstów napisanych w różnych językach, w celu uzyskania zmiennej β_T =1

W większości języków europejskich wystarczy niewielka (z wyjątkami) modyfikacja formy pojedynczej, aby powstała forma mnoga, np.

- w języku angielskim jest to dodanie litery –s lub –es
- w języku niemieckim jest to dodanie litery –n/en, -r/er, -e, -s
- w języku włoskim to zamiana liter –e na –i , -a na –e, i, -o na –i
- w języku hiszpańskim to dodanie liter –s, -es.

Kolejnym powodem dopasowania wartości *bp* może być specyficzna wiedza o pochodzeniu jednego z badanych dokumentów tekstowych, np. taka, że dokument może zawierać liczne błędy ortograficzne.

Reasumując, badania wykazały, że dla zmiennej progowej wartością bliżej uniwersalnej dla większości języków europejskich jest *bp*=0,70.

Modyfikacja algorytmu Levenshteina poprzez dodanie analizy podobieństwa wyrazów umożliwia wprowadzenie pewnego stopnia uniwersalności względem języka dla tego algorytmu. Jak wykaże poniższy przykład, nie jest to metoda idealna, a jedynie sprawdza się w

przypadkach, w których podobne wyrazy występują odpowiednio po sobie w tekście. Jest to cecha algorytmu Levenshteina.

Dla poniższego przykładu, macierz Levenshteina według powyższej formuły i wartości progowej *bp=0,65*, przyjętej przez użytkownika aplikacji komputerowej analizującej dane, wyglądałaby następująco:

		Dóży	**kot**	**wszedł**	**na**	**dżewo**
	0	1	2	3	4	5
Duży	1	0	1	2	3	4
kot	2	1	0	1	2	3
wyszedł	3	2	1	0	1	2
na	4	3	2	1	0	1
drzewo	5	4	3	2	1	**0**

Tabela 3.6. Przykład porównania dwóch krótkich zdań niosących to samo przesłanie, posiadających błędy ortograficzne: *„Dóży kot wszedł na dżewo”* oraz *„Duży kot wyszedł na drzewo”*

Jednakże zamiana wyrazów sąsiadujących ze sobą wprowadza istotną korektę wyniku (tabela 3.7).

		Kot	**dóży**	**wszedł**	**na**	**dżewo**
	0	1	2	3	4	5
Duży	1	1	1	2	3	4
kot	2	1	2	2	3	4
wyszedł	3	2	2	2	3	4
na	4	3	3	3	2	3
drzewo	5	4	4	4	3	**2**

Tabela 3.7. Przykład porównania dwóch krótkich fragmentów zdań z analogicznym przesłaniem do poprzedniego przypadku, ale ze zmienioną kolejnością dwóch sąsiadujących ze sobą wyrazów

Do odrębnej dyskusji pozostaje kwestia tego, czy analizowane zdania według czytelnika są takie same, czy nie. Bezsporną kwestią jest to, że niosą to samo przesłanie.

Powyższą koncepcję analizy zdań bazującą na algorytmie odległości Levenshteina można rozszerzyć o słownik wyrazów bliskoznacznych (tzw. tezaurus, Tabela 3.8), dodatkowo uwzględniając korektę błędów ortograficznych w oparciu o wspomnianą miarę podobieństwa[40]. Metoda polega na znalezieniu w słowniku odpowiadającej grupy wyrazów

[40] program komputerowy dostępny jest pod adresem:
https://cloud.mck.pk.edu.pl/index.php/s/A0MmdvPr6Yizx9A (autor: Artur Niewiarowski)

do analizowanego terminu, a następnie sprowadzenie do wspólnej postaci (np. kodu liczbowego identyfikującego grupę terminów).

ID	Wspólna definicja	Terminy pokrewne
#1	imiona	Anna, Maria, Zbigniew
#2	owoce	owoce, jabłko, jabłka, pomarańcza, pomarańcze, banan, banany
#3	mieszkanie	dom, domek, mieszkanie, rezydencja
#4	oraz	oraz, i
#5	auto	auto, samochód, samochodzik
#6	duży	duży, ogromny, wielki
#7	drzewo	drzewo, drzewko, sosna, dąb
#8	wszedł	wszedł, wkroczył, wskoczył, wprowadził się

Tabela 3.8. Tabela ilustrująca prostą strukturę tezaurusa wraz z przykładową zawartością

Lp.	Tekst nr 1	Tekst nr 2	Wynik kodowania tekstu nr 1	Wynik kodowania tekstu nr 2	Wynik porównania (*p*)
1	Mateusz ma dom oraz samochód	Mateusz ma domek i samochód	mateusz ma #3 #4 #5	mateusz ma #3 #4 #5	1 (100%), *bp* dowolne
2	Anna lubi jabłka	Maria lubi owoce	#1 lubi #2	#1 lubi #2	1 (100%), *bp* dowolne
3	Anna lubi jabłka	Owoce lubi Maria	#1 lubi #2	#2 lubi #1	0,33 (33%), *bp* dowolne
4	Duży kot wskoczył na sosnę	Dóży kot wszedł na dżewo	#6 kot #8 na #7	#6 kot #8 na #7	1 (100%), dla *bp*=0,65

Tabela 3.9. Tabela z krótkimi prostymi zdaniami poddanymi analizie

Jak pokazują przykłady zawarte w tabeli (3.9), odnalezienie różnych, ale tożsamych wyrazów w słowniku wyrazów bliskoznacznych i sprowadzenie ich do wspólnego reprezentanta wystarczyło, aby metoda porównywania ciągów tekstowych oparta o algorytm odległości edycyjnej nabrała nowego charakteru i stała się prostym narzędziem do wykrywania tego typu nadużyć w tekstach. Dla przykładu nr 4 z tabeli, dla zdań: *„Dóży kot wszedł na dżewo”* oraz *„Duży kot wskoczył na sosnę”*, nastąpiło sprowadzenie wyrazów do wspólnej formy, uwzględniające błędy ortograficzne (np. dżewo → drzewo) oraz odmianę wyrazów w analizowanych zdaniach względem słownika wyrazów pochodnych (np. sosnę → sosna). Wartość progowa *bp* dla algorytmu podobieństwa wyrazów została ustalona przez użytkownika programu na 0,65. Umożliwiło to odnalezienie takich wyrazów jak: *„dóży”* oraz *„sosnę”* w słowniku wyrazów pochodnych i sprowadzenie ich do formy reprezentującej odpowiednią grupę wyrazową.

Metoda wykorzystująca tezaurus przedstawiona została w poniższych krokach:

Krok 1. Pobierz wszystkie kody (wyrazy podstawowe) dwóch analizowanych terminów (wyrazów) (a_S, b_S) z tezaurusa.

$$\Psi(a_s, \tau) \rightarrow \mathbf{C}_{\mathbf{a_S}}(i_S, t_{i_S}) \tag{3.4}$$

$$\Psi(b_s, \tau) \rightarrow \mathbf{C}_{b_S}(j_S, t_{j_S}) \tag{3.5}$$

gdzie:
Ψ - funkcja pobierająca kody (wyrazy podstawowe) ze zdań a_s i b_s,
τ - tezaurus,
t_{i_S} – liczba wariantów terminu (wyrazu),
i_s – numer wyrazu w zdaniu a_s
$\mathbf{C}_{\mathbf{a_S}}$ – tablica kodów (wyrazów podstawowych) terminów zdania a_S.

Krok 2. Obliczenie częstości wystąpienia kodów w zdaniach a_s i b_s.

$$\Gamma(\mathbf{C}_{\mathbf{a_S}}, \mathbf{C}_{\mathbf{b_S}}) \rightarrow \mathbf{C}_{\mathbf{ab_S}}(h) \tag{3.6}$$

gdzie:
Γ – funkcja obliczająca częstość wystąpienia kodów w zdaniach,
$\mathbf{C}_{\mathbf{ab_S}}$ – tablica najlepiej dopasowanych pod względem częstości wystąpienia terminów,
h – ID terminu (wyrazu).

Krok 3. Podmiana terminów (wyrazów) w zdaniach na odpowiedniki w postaci kodów (wyrazów podstawowych).

$$\Phi(\mathbf{C}_{\mathbf{ab_S}}, \mathbf{C}_{\mathbf{a_S}}) \rightarrow \mathbf{NC}_{\mathbf{a_S}}(i_S) \tag{3.7}$$

$$\Phi(\mathbf{C}_{\mathbf{ab_S}}, \mathbf{C}_{\mathbf{b_S}}) \rightarrow \mathbf{NC}_{\mathbf{b_S}}(j_S) \tag{3.8}$$

gdzie:
Φ – funkcja podmieniająca wyrazy (terminy) na odpowiedniki w postaci kodów (wyrazów podstawowych)
$\mathbf{NC}_{\mathbf{b_S}}$ – przekonwertowane zdania po działaniu funkcji podmiany wyrazów na kody.

Krok 4. Funkcja obliczająca podobieństwo zdań (formuła 3.3).

$$\Omega(\boldsymbol{NC_{a_S}}, \boldsymbol{NC_{b_S}}, bp, bp_\tau) \rightarrow P_S \tag{3.9}$$

gdzie:
Ω – funkcja obliczająca podobieństwo pomiędzy ciągami tekstowymi (Formuła 5),
bp – akceptowalna granica podobieństwa terminów porównywanych względem zdań,
bp_τ - akceptowalna granica podobieństwa terminów pomiędzy porównywanym terminów w zdaniu a terminem w tezaurusie.

W kolejnych przykładach poniżej przedstawiono wpływ jaki ma wprowadzenie miary podobieństwa pomiędzy ciągami tekstowymi bazującej na odległości Levenshteina oraz dodatkowego zastosowania tezaurusa dla jakości wyniku porównania dwóch krótkich zdań (formuła 3.3). Przykłady przedstawiono w oparciu o teksty w języku angielskim. Tezaurus dla testów reprezentuje tabela poniżej (tabela 3.10).

ID	**Wspólna definicja**	**Terminy pokrewne**
#1	names	Tom, Mary, John, Jimmy, Jane, Derek, Gina
#2	cars	car, auto, automobile, taxi, vehicle
#3	numbers	one, two three, four, five, six, seven, eight, nine, ten
#4	seasons	spring, summer, autumn, winter
#5	fruits	apple, pear, cherry, mango, kiwi, watermelon
#6	cities	Warsaw, Berlin, London
#7	phones	phone, telephone, iPhone, mobile phone
#8	very	very, extremely
#9	shortcuts1	is not, isn’t
#10	shortcuts2	are not, aren’t
#11	shortcuts3	don’t, do not
#12	fluid	milk, water
#13	my_friends	Tom, Jack, Ella, Olivia

Tabela 3.10. Tabela ilustrująca prostą strukturę tezaurusa wraz z przykładową zawartością

Lp.	Zdania (1)	Zdania (2)
z1	Tom is writing a letter	**Dere** is **writin** a **letters**
z2	We are waiting for a taxi	We are **waitin** for **car**
z3	Is Mary having breakfast?	Is **Jane hasing brekfest**?
z4	Tom is not writing a letter	**Jimm** isn't **writin leter**
z5	He isn't looking at the stars	He **is not look** at the **start**
z6	He drinks milk twice a day	He **is** drinks **water** twice a day
z7	We go to work six times a week	We **goes** to **works seven** times a week
z8	I always feel great in spring	I **alway** feel great in **summer**
z9	Do you like apples?	**Does** you **likes pear**?
z10	I don't like milk	I **do not** likes water
z11	Tom was writing the letter all day yesterday	**Jimmy writting** the **leter** all day **yestaredy**
z12	They met when they were studying in Berlin	They met when they were studying in **Warsaw**
z13	I was working in London this time last year	I was **work** in **Berlin** this **times** last **years**
z14	I have found his telephone number	I have found his **phone** number
z15	I was shocked when I found out that Derek and Gina had got divorced	I was **shock** when I found out that **John** and **Mary has gotten** divorced
z16	I have been working for five hours	I **has** been working for **six hour**
z17	It had been raining for days so when they finally left, the roads were very muddy	It **has** been raining for days so when they **finaly** left, the roads were **extremly** muddy

Tabela 3.11. Tabela z krótkimi zdaniami poddanymi analizie

Tekst pogrubiony w tabeli 3.11 w prawej kolumnie (2) oznacza błąd ortograficzny lub podmianę wyrazu względem zdania w kolumnie po lewej stronie (1). Ma to na celu sprawdzenie dokładności przedstawianych metod, tj. miary podobieństwa w połączeniu ze słownikiem wyrazów bliskoznacznych.

Lp.	Kolumna (1)	Kolumna (2)
z1	#1 is writing a letter	#1 is writin a letters
z2	we are waiting for a #2	we are waitin for #2
z3	is #1 having breakfast?	is #1 hasing brekfest?
z4	#1 #9 writing a letter	#1 #9 writin leter
z5	he #9 looking at the stars	he #9 look at the start
z6	he drinks #12 twice a day	he is drinks #12 twice a day
z7	we go to work #3 times a week	we goes to works #3 times a week
z8	i always feel great in #4	i alway feel great in #4
z9	do you like #5?	does you likes #5?
z10	i #11 like #12	i #11 likes #12
z11	#1 was writing the letter all day yesterday	#1 writting the leter all day yestaredy
z12	they met when they were studying in #6	they met when they were studying in #6
z13	i was working in #6 this time last year	i was work in #6 this times last years
z14	i have found his #7 number	i have found his #7 number
z15	i was shocked when i found out that #1 and #1 had got divorced	i was shock when i found out that #1 and #1 has gotten divorced
z16	i have been working for #3 hours	i has been working for #3 hour
z17	it had been raining for days so when they finally left, the roads were #8 muddy	it has been raining for days so when they finaly left, the roads were #8 muddy

Tabela 3.12. Tabela przedstawiająca zdania po konwersji

Tabela 3.12 zawiera przekonwertowane zdania z tabeli poprzedniej (3.11) względem reprezentantów wspólnych grup wyrazowych na bazie słownika (3.10). Tabela poniżej (3.13) zawiera zestawienie trzech serii analizy powyższych zdań:

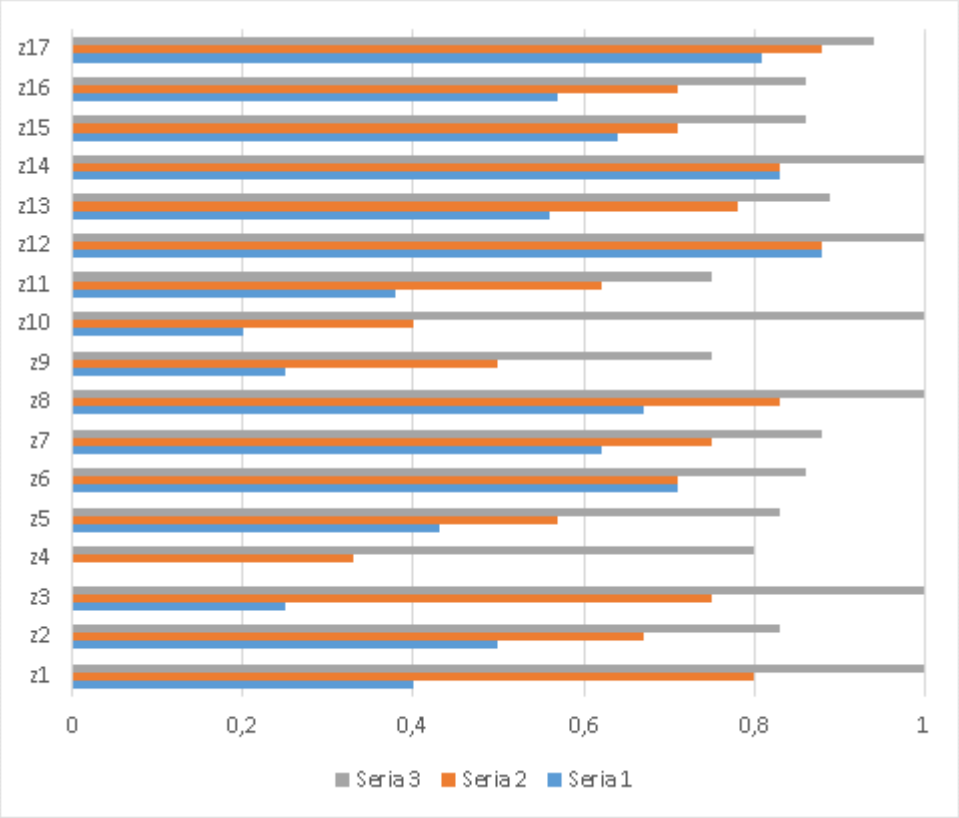
seria 1 – wynik miary podobieństwa *p* bazującej na algorytmie odległości Levenshteina dla analizy zdań zamiast wyrazów (formuła 3.2),

seria 2 – wynik miary podobieństwa *p* bazującej na algorytmie odległości Levenshteina dla analizy zdań zamiast wyrazów uwzględniająca podobieństwo terminów (formuła 3.3),

seria 3 – wynik miary podobieństwa *p* bazującej na algorytmie odległości Levenshteina dla analizy zdań zamiast wyrazów uwzględniająca podobieństwo terminów i tezaurus (formuła 3.9).

Lp.	Seria 1	Seria 2	Seria 3
z1	0,4	0,8	1
z2	0,5	0,67	0,83
z3	0,25	0,75	1
z4	0	0,33	0,8
z5	0,43	0,57	0,83
z6	0,71	0,71	0,86
z7	0,62	0,75	0,88
z8	0,67	0,83	1
z9	0,25	0,5	0,75
z10	0,2	0,4	1
z11	0,38	0,62	0,75
z12	0,88	0,88	1
z13	0,56	0,78	0,89
z14	0,83	0,83	1
z15	0,64	0,71	0,86
z16	0,57	0,71	0,86
z17	0,81	0,88	0,94

Tabela 3.13. Tabela przedstawiająca miary *p* podobieństwa uwzględniające omówione podejścia do analizy ciągów tekstowych w powyższym rozdziale



Rysunek 3.1. Wykres przedstawiający zależności pomiędzy wartościami poszczególnych serii analizy ciągów tekstowych ujętymi w tabeli powyżej (3.13)

Porównanie przedstawionych badań (tabela 3.13, rysunek 3.1) pokazuje, że zastosowanie miary podobieństwa bazującej na odległości Levenshteina w połączeniu z tezaurusem podnosi jakość analizy podobieństwa zdań od 20% nawet do 75-100%.

Lp.	**Seria 1**	**Seria 2**	**Seria 3**	**Seria 4**
z1	0,80	0,40	0,08	0,25
z2	0,67	0,55	0,10	0,37
z3	0,75	0,25	0,06	0,14
z4	0,33	0,00	0,00	0,00
z5	0,57	0,46	0,07	0,30
z6	0,71	0,77	0,12	0,62
z7	0,75	0,63	0,07	0,45
z8	0,83	0,66	0,11	0,50
z9	0,50	0,25	0,62	0,14
z10	0,40	0,22	0,50	0,12
z11	0,62	0,40	0,53	0,25
z12	0,88	0,90	0,00	0,81
z13	0,78	0,56	0,06	0,38
z14	0,83	0,83	0,13	0,71
z15	0,71	0,69	0,04	0,52
z16	0,71	0,57	0,08	0,40
z17	0,88	0,81	0,05	0,68

Tabela 3.14. Tabela przedstawiająca wartości porównania ciągów tekstowych za pomocą znanych popularnych metod

Tabela 3.14 przedstawia wyniki podobieństwa ciągów tekstowych za pomocą następujących metod:

seria 1 – algorytm odległości Levenshteina dla analizy zdań zamiast wyrazów uwzględniająca podobieństwo terminów (formuła 3.3),

seria 2 – odległość kosinusowa (ang. *Cosine distance*) pomiędzy wektorami wag frekwencji terminów (ang. *term frequency weight method - tf*) analizowanych ciągów tekstowych,

seria 3 – odległość Dice'a (ang. *Dice distance*) pomiędzy wektorami wag frekwencji terminów analizowanych ciągów tekstowych,

seria 4 – odległość Jaccarda (ang. *Jaccard distance*) pomiędzy wektorami wag frekwencji terminów analizowanych ciągów tekstowych.

Pomimo tego, że wspólnym mianownikiem przedstawionych metod jest brak implementacji słownika wyrazów bliskoznacznych, metoda analizy podobieństwa tekstów bazująca na odległości Levenshteina pod względem działania wyraźnie różni się od popularnych metod bazujących na wadze terminów. Wynika to z tego, że nie potrzebuje ona innych dokumentów w celu oszacowania wag, czy też prostego przeliczenia ilości wystąpień terminów pomiędzy

dokumentami. Sama metoda wyraźnie przoduje pod względem jakości analizy szacowania podobieństwa ciągów tekstowych, jak również ze względu na charakter algorytmu edycyjnego uwzględnia występowanie po sobie wyrazów w dokumencie (czy też w zdaniu) w przeciwieństwie do algorytmów z serii 2-3.

Analiza przedstawiona w kolejnej części rozdziału kontynuuje tematykę analizy zdań bazującą wyłącznie na algorytmie odległości edycyjnej Levenshteina, ale w innym kontekście. W tabeli poniżej przedstawiono wybrane wyniki analizy skuteczności wykrywania podobieństwa pomiędzy krótkimi fragmentami tekstów po translacji na wybrane języki obce. Przeanalizowane zostały znane cytaty Alberta Einsteina napisane w języku polskim, przetłumaczone za pomocą narzędzia *Google Translate* (opartego m.in. na algorytmach sztucznej inteligencji [35], uznawanego przez znaczną część użytkowników translatorów z całego świata za najlepszy program tłumaczący teksty na języki obce[41] [42]) odpowiednio na języki: angielski, rosyjski i niemiecki, a następnie przetłumaczone z powrotem na język polski – wersje w języku polskim zostały poddane analizie. Celem zastosowania wspomnianego narzędzia do tłumaczenia tekstów jest chęć uzyskania wyników możliwie obiektywnych – czyli bazujących na tłumaczeniach maszyny, nie człowieka, jednocześnie zachowujących sens przesłania.

Lp.	Cytat w języku polskim (oryginalny)	Cytat w języku angielskim (przetłumaczony)	Cytat w języku polskim (przetłumaczony ponownie)	Wynik analizy
1	Wyobraźnia znaczy więcej niż wiedza	Imagination is more than knowledge	Wyobraźnia jest czymś więcej niż wiedzą	66,7%
2	Nigdy nie myślę o przyszłości. Nadchodzi ona wystarczająco szybko	I never think about the future. It comes soon enough	Nigdy nie myślę o przyszłości. Chodzi wystarczająco szybko	77,8%
3	Uczony jest człowiekiem, który wie o rzeczach nieznanych innym i nie ma pojęcia o tym, co znają wszyscy	The scholar is a man who knows about things unknown to others and have no idea about what they know everyone	Uczony jest człowiekiem, który wie o rzeczach nieznanych innym i nie masz pojęcia o tym, co znają wszyscy.	94,4%
4	Wszyscy wiedzą, że czegoś nie da się zrobić, i przychodzi taki jeden, który nie wie, że się nie da, i on właśnie to robi	Everyone knows that something can not be done, and there comes a one who does not know	Każdy wie, że coś nie da się zrobić, i przychodzi taki jeden, który nie wie, że nie da, i on właśnie to robi.	83,3%

[41] "*I traveled the world for 6 months, and here's the single best app I couldn't live without*": https://www.businessinsider.nl/best-app-for-traveling-the-world-google-translate-2018-10/?international=true&r=US

[42] *"A British court was forced to rely on Google Translate because it had no interpreter"*: http://www.businessinsider.com/teesside-magistrates-court-forced-to-rely-on-google-translate-because-it-had-no-interpreter-2017-8

		that it is impossible, and he does just that		
5	Prawdą jest to, co wytrzyma próbę doświadczenia	Truth is what stands the test of experience	Prawdą jest to, co przetrwa próbę doświadczenia	85,7%
6	Nie wiem, na co będzie trzecia wojna światowa, ale czwarta będzie na pewno na maczugi	I do not know what will be a third world war, but the fourth will surely be on the club.	Nie wiem co będzie jedna trzecia wojna światowa, ale czwarta będzie na pewno na klubie.	80%
7	Najpiękniejszym, co możemy odkryć, jest tajemniczość	The most beautiful thing we can discover, is a mystery	Najpiękniejszym, co możemy odkryć, jest tajemnicą	83,3%
8	Tylko życie poświęcone innym warte jest przeżycia	Only a life devoted to others is worth the experience	Tylko życie poświęcone innym jest warte doświadczenia	57,1%

Tabela 3.15. Tabela zawierająca przetłumaczone w kierunkach: polski → angielski → polski. Wartość progowa dla miary podobieństwa wyrazów w analizowanych zdaniach została ustalona na bp=0,7

Ze względu na niedoskonałość tłumaczenia tekstów przez automaty, powyższa analiza (tabela 3.15) zachowała wcześniej założony cel, gdyż teksty w języku polskim różnią się pomiędzy sobą, ale nie na tyle, aby całkowicie straciły sens. Zarówno w tabeli powyżej (3.15), jak również w tabelach poniżej (3.16, 3.17) wynik podobieństwa zależy od tego, czy w porównywanych tekstach znajdują się te same wyrazy, ewentualnie odmienione, w jakiej kolejności po sobie ułożone są wyrazy, czy tak samo w obu tekstach – jeżeli tak, wtedy wynik porównania jest wysoki. Jeżeli w badanych tekstach kolejność wyrazów nie jest taka sama oraz część wyrazów została zastąpiona przez ich odpowiedniki, które nie są do siebie podobne – to wynik porównania będzie niski. Widać to zarówno w języku angielskim (tabela 3.15), rosyjskim (tabela 3.16), jak również niemieckim (tabela 3.17).

Lp.	**Cytat w języku polskim (oryginalny)**	**Cytat w języku rosyjskim (przetłumaczony)**	**Cytat w języku polskim (przetłumaczony ponownie)**	**Wynik analizy**
1	Wyobraźnia znaczy więcej niż wiedza	Воображение более чем знания	Wyobraźnia jest czymś więcej niż wiedzą	66,7%
2	Nigdy nie myślę o przyszłości. Nadchodzi ona wystarczająco szybko	Я никогда не думаю о будущем. Он идет достаточно скоро	Nigdy nie myślę o przyszłości. Chodzi wystarczająco szybko	77,8%
3	Uczony jest człowiekiem, który wie o rzeczach nieznanych innym i nie ma pojęcia o tym, co znają wszyscy	Ученый человек, который знает о том, неизвестные другим, и понятия не имею о том, что они знают все	Nauczył człowiek, który wie o nieznanych innym i nie mają pojęcia o tym, co wiedzą wszystko	66,7%
4	Wszyscy wiedzą, że czegoś nie da się zrobić, i przychodzi taki jeden, który nie wie, że się nie da, i on właśnie to robi	Все знают, что что-то не может быть сделано, и наступает тот, кто не знает, что это невозможно, и он	Każdy wie, że coś nie da się zrobić, i przychodzi człowiek, który nie wie, że jest to niemożliwe, a on właśnie to robi	62,5%

		делает именно это		
5	Prawdą jest to, co wytrzyma próbę doświadczenia	Истина это то, что выдерживает испытание опытом	Prawdą jest to, co przetrwa próbę doświadczenia	85,7%
6	Nie wiem, na co będzie trzecia wojna światowa, ale czwarta będzie na pewno na maczugi	Я не знаю, что будет третья мировая война, но четвертая, безусловно, будет в клубе	Nie wiem co będzie jedna trzecia wojna światowa, ale czwarta będzie na pewno być w klubie	68,8%
7	Najpiękniejszym, co możemy odkryć, jest tajemniczość	Самое красивое, что мы можем узнать, остается загадкой	Najpiękniejsza rzecz, jaką możemy znaleźć pozostaje tajemnicą	28,6%
8	Tylko życie poświecone innym warte jest przeżycia	Только жизнь посвящена другим стоит опыт	Tylko życie poświęcone innym jest warte doświadczenia	57,1%

Tabela 3.16. Tabela zawierająca przetłumaczone w kierunkach: polski → rosyjki → polski. Wartość progowa dla miary podobieństwa wyrazów w analizowanych zdaniach została ustalona na bp=0,7

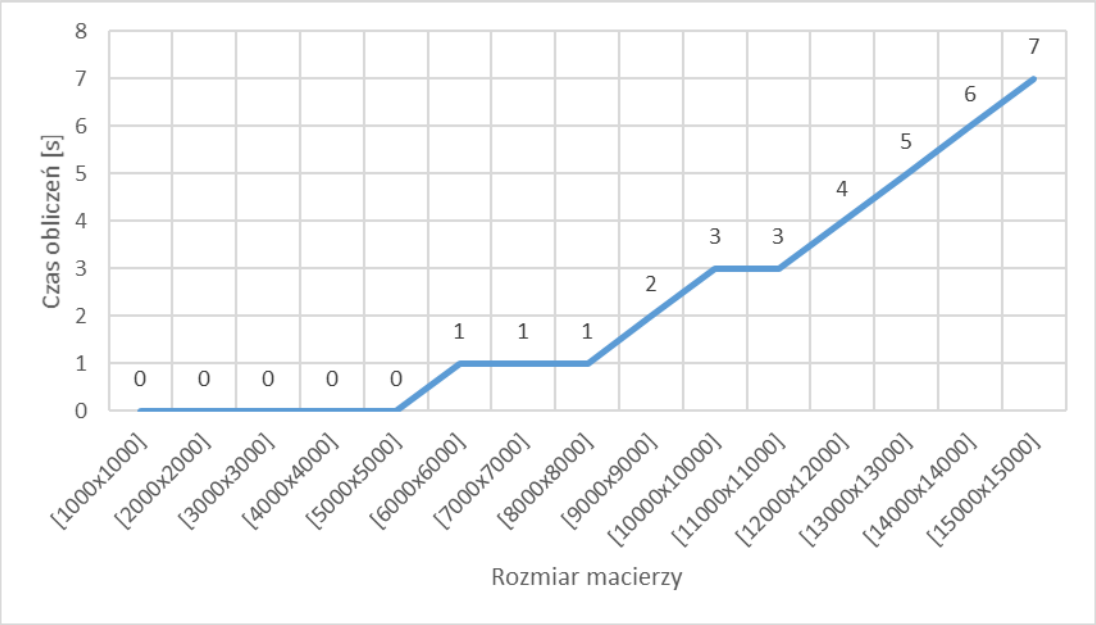
Jak widać na przykładach przedstawionych w tabelach, mechanizm porównania krótkich fragmentów tekstów, bazujący na algorytmie odległości edycyjnej Levenshteina i mierze podobieństwa, może znaleźć zastosowanie do przeprowadzania prostych i szybkich testów skuteczności tłumaczenia tzw. translatorów.

Lp.	**Cytat w języku polskim (oryginalny)**	**Cytat w języku niemieckim (przetłumaczony)**	**Cytat w języku polskim (przetłumaczony ponownie)**	**Wynik analizy**
1	Wyobraźnia znaczy więcej niż wiedza	Phantasie ist mehr als Wissen	Wyobraźnia jest czymś więcej niż wiedzą	66,7%
2	Nigdy nie myślę o przyszłości. Nadchodzi ona wystarczająco szybko	Ich denke niemals an die Zukunft. Sie kommt früh genug	Nigdy nie myślę o przyszłości. Chodzi wystarczająco szybko	77,8%
3	Uczony jest człowiekiem, który wie o rzeczach nieznanych innym i nie ma pojęcia o tym, co znają wszyscy	Der Gelehrte ist ein Mann, über Dinge unbekannt, andere weiß und haben keine Ahnung, was sie wissen alle	Uczony jest człowiekiem o rzeczach nieznanych, a inne białe i nie mają pojęcia, co wszyscy wiedzą	44,4%
4	Wszyscy wiedzą, że czegoś nie da się zrobić, i przychodzi taki jeden, który nie wie, że się nie da, i on właśnie to robi	Jeder weiß, dass etwas nicht getan werden kann, und es kommt einer, der nicht weiß, dass es unmöglich ist, und er tut genau dies	Każdy wie, że coś można zrobić, a tam jest jeden, który nie wie, że nie da, i on właśnie to robi	58,3%
5	Prawdą jest to, co wytrzyma próbę doświadczenia	Wahr ist, was steht den Test der Erfahrung	Prawdą jest to, co przetrwa próbę doświadczenia	85,7%
6	Nie wiem, na co będzie trzecia wojna światowa, ale czwarta będzie na pewno na maczugi	Ich weiß nicht, was ein dritter Weltkrieg sein, aber der vierte wird sicherlich auf der Club sein	Nie wiem co jest jedna trzecia wojna światowa, ale czwarta będzie na pewno na klubie	73,3%

7	Najpiękniejszym, co możemy odkryć, jest tajemniczość	Das Schönste, was wir entdecken können, ist ein Rätsel	Najpiękniejszym, co możemy odkryć, jest tajemnicą	83,3%
8	Tylko życie poświecone innym warte jest przeżycia	Nur ein Leben für die anderen ist die Erfahrung wert	Tylko życie przeżyte dla innych jest warte doświadczenia.	37,5%

Tabela 3.17. Tabela zawierająca przetłumaczone w kierunkach: polski → niemiecki → polski. Wartość progowa dla miary podobieństwa wyrazów w analizowanych zdaniach została ustalona na bp=0,7

Opisana powyżej metoda jest stosunkowo dokładna. Algorytm analizuje zdania z uwzględnieniem sekwencyjności występowania po sobie wyrazów, co w pewnych przypadkach jest rzeczą pożądaną. Natomiast nadaje się do analizy niedużych ciągów tekstowych. Przy analizie większych danych (np. wypracowań, książek), metoda staje się nieoptymalna czasowo, co można zaobserwować na poniższych wykresach podsumowujących przeprowadzone analizy ilościowe[43].



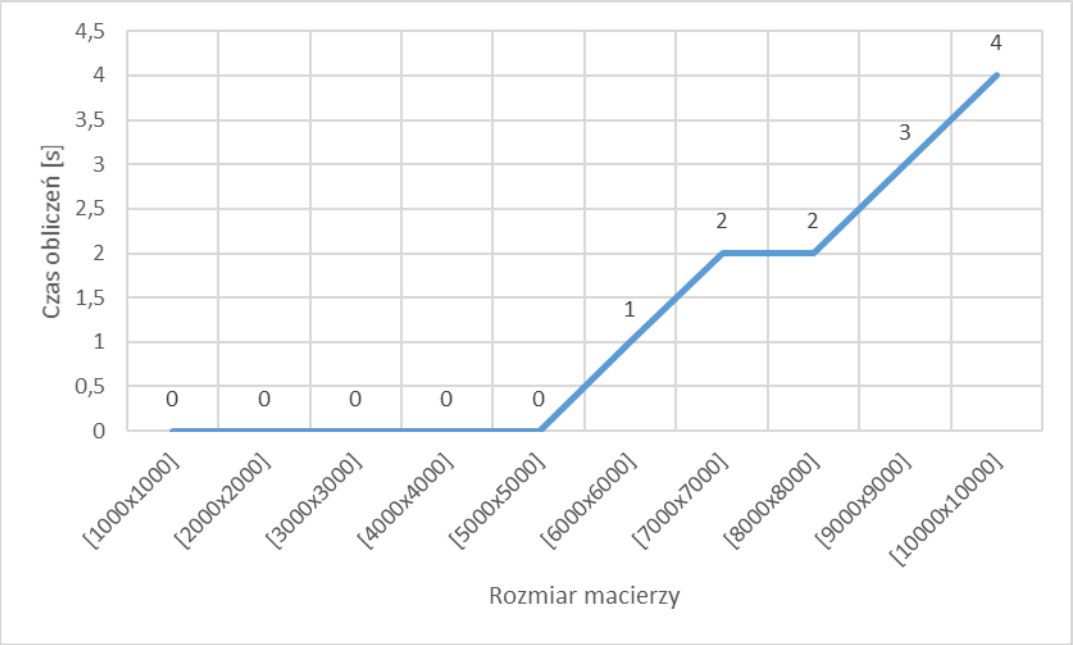
Rysunek 3.2. Wykres zależności czasu obliczeń dla rosnącego rozmiaru macierzy analizowanych tekstów (ciągów tekstowych składających się ze znaków) za pomocą algorytmu odległości Levenshteina[44]

Powyższy wykres przedstawia zależność czasu obliczeń do rozmiaru macierzy (będącej produktem działania algorytmu Levenshteina) powstającej w wyniku analizy dwóch ciągów

[43] Testy zostały wykonane na komputerze o standardowych parametrach obliczeniowych: Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz, 16 GB RAM

[44] autor programu wykonującego obliczenia: Artur Niewiarowski, https://cloud.mck.pk.edu.pl/index.php/s/MEmYufR3ZDV6YVs

tekstowych (dla czytelności prezentacji – macierzy kwadratowych). Jak pokazują dalej prowadzone badania których wyniki zamieszczono poniżej, pomimo zmiany analizowanych znaków na wyrazy i zastosowaniu zmodyfikowanego algorytmu odległości edycyjnej Levenshteina (3.2), zależność przyrostu czasu obliczeń od ilości analizowanych danych nie zmienia się znacząco. Wynik ten jest efektem podobnej złożoności obliczeniowej obu algorytmów, która ze względu na algorytm składający się głównie z dwóch pętli iterujących po elementach macierzy wynosi w obu przypadkach $O(n)^2$.

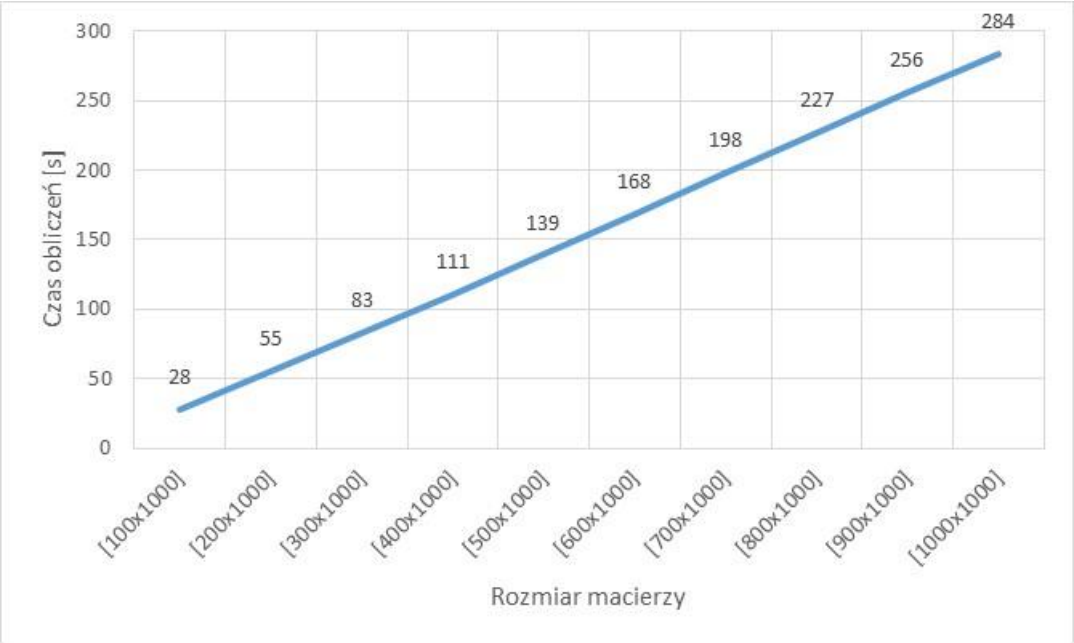


Rysunek 3.3. Wykres zależności czasu obliczeń do rosnącego rozmiaru macierzy badanych zdań (ciągów tekstowych składających się z wyrazów) za pomocą algorytmu odległości Levenshteina[45]

Istnieje wiele metod dotyczących sposobów przyspieszenia obliczeń algorytmu odległości Levenshteina. Jedną z nich jest praca autora [11], w której zaproponowana została metoda dekompozycji macierzy na mniejsze podmacierze. Badane są w niej ciągi tekstowe, dla których macierze są rzędu nawet 9 x 10^{12} elementów [46].

[45] autor programu wykonującego obliczenia: Artur Niewiarowski, https://cloud.mck.pk.edu.pl/index.php/s/N7puy2ziFAro3xT

[46] W ramach analizy metody wykorzystana została technologia firmy Microsoft dla wielowątkowości ujęta w środowisku .NET. Intel(R) Xeon(R) CPU E5-2620 0 @ 2.00GHz, 24 threads, 24 GB RAM



Rysunek 3.4. Wykres zależności czasu obliczeń do rosnącego rozmiaru macierzy badanych zdań za pomocą algorytmu odległości Levenshteina z użyciem miary podobieństwa

Wykres powyżej (rys. 3.4) pokazuje, że po dodatkowej implementacji algorytmu Levenshteina w celu analizy podobieństwa wyrazów (tj. miary podobieństwa[47]) w głównym algorytmie, wydajność algorytmu spada drastycznie względem poprzednich przykładów. O ile w algorytmie bez modyfikacji, czas obliczeń dla macierzy wielkości 1000x1000 wynosił poniżej 1 sek., to w powyższym przypadku jest to aż ok. 5 min. Związane jest to ze złożonością obliczeniową całości mechanizmu, która wynosi $O(n)^4$. Dla celów testowych przyjęto, że jedno zdanie posiada 11 wyrazów składających się w sumie ze 110 znaków, co jest równie ok. 10 znaków na jeden wyraz.

Poniżej znajduje się wykres (rys. 3.5), który przedstawia analizę danych tej samej wielkości, ale z użyciem technologii wielowątkowości. Obliczenia znacząco przyspieszyły (od pewnego stopnia wielkości danych ponad dwukrotnie), ale nadal jest to wynik niesatysfakcjonujący dla dużych zbiorów danych tekstowych.

[47] autor programu: Artur Niewiarowski, https://cloud.mck.pk.edu.pl/index.php/s/JhW7DUGOnSUSg1N

Rysunek 3.5. Wykres zależności czasu obliczeń do rosnącego rozmiaru macierzy badanych zdań za pomocą algorytmu odległości Levenshteina z użyciem miary podobieństwa i wielowątkowości[48]

Dla celów testowych przyjęto, że jedno zdanie posiada 11 wyrazów składających się w sumie ze 110 znaków, co jest równie ok. 10 znaków na jeden wyraz; natomiast na jeden wątek przypada fragment macierzy o wielkości 500 x 500 zdań.

W następnym podrozdziale zaproponowana została metoda będąca alternatywą dla powyższych rozważań, którego głównym przesłaniem jest nadal uniwersalność względem języka badanych danych tekstowych. Jest natomiast o wiele wydajniejsza oraz posiada inne cechy ją charakteryzujące.

[48] autor programu: Artur Niewiarowski, https://cloud.mck.pk.edu.pl/index.php/s/GlvYvIV4gT4TQ92

3.3. Koncepcja metody analizy macierzowej danych tekstowych

Koncepcja analizy macierzowej dwóch dokumentów tekstowych [47] polega na zbudowaniu na bazie dokumentów macierzy, o rozmiarze wynikającym z wielkości badanych dokumentów. Macierz jest uzupełniana wartościami 0 lub 1 (prawda lub fałsz) w zależności od podobieństwa porównywanych wyrazów w danej iteracji, która opisana jest poniższą formułą:

$$\begin{aligned} &\mathop{\mathrm{I}}_{ti=1}^{tm}\mathop{\mathrm{I}}_{tj=1}^{tn}\mathbf{M}[ti,tj]=\beta \\ &\begin{cases}\beta=\text{true}:\mathbf{Doc1}[ti]\equiv\mathbf{Doc2}[tj]\\ \beta=\text{false}:\mathbf{Doc1}[ti]\neq\mathbf{Doc2}[tj]\end{cases} \end{aligned} \tag{3.10}$$

gdzie:

M – macierz dokumentów o rozmiarach *tn*, *tm*, stworzona na podstawie dokumentów: *Doc1* oraz *Doc2*,

tm, tn – liczba wyrazów tworzących dokumenty Doc1 i Doc2 (rozmiary dokumentów),

ß – zmienna przyjmująca odpowiednio wartości 0 lub 1 (prawda lub fałsz),

M[ti,tj] - (ti,tj) – element macierzy **M**,

Doc1[ti] – *ti* - element dokumentu *Doc1*, oddzielony spacją od sąsiednich elementów dokumentu.

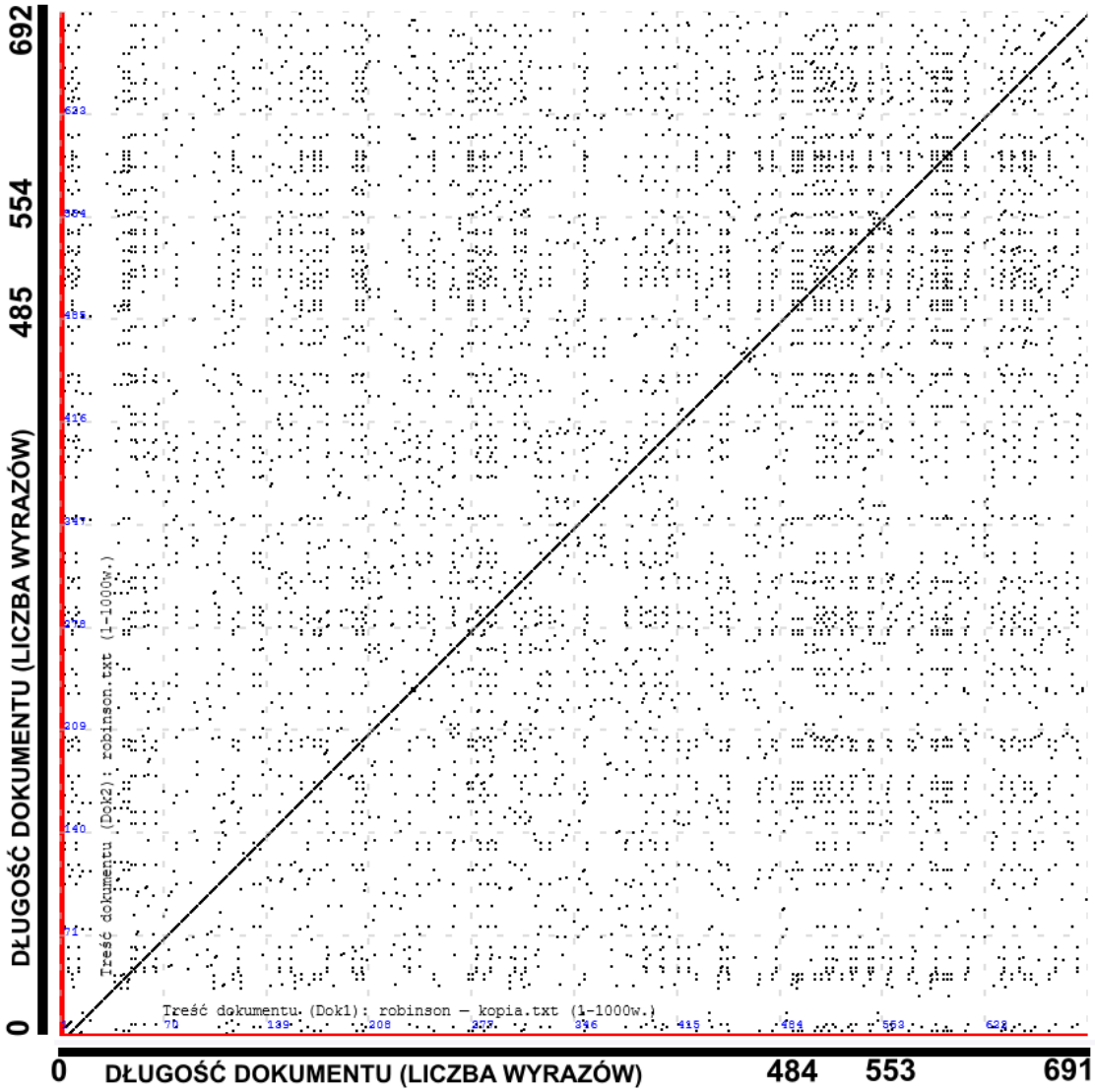
Poniżej znajduje się pseudo-kod (3.11) opisujący główną ideę wypełnienia macierzy (bez uwzględnienie miary podobieństwa wyrazów):

(3.11)

```
input variables: term Doc1[0..tm-1], term Doc2[0..tn-1]
   declare: boolean M[0..tm-1, 0..tn-1]
   for ti from 0 to tm-1
        for tj from 0 to tn-1

         if term of Doc1 at (ti) = term of Doc2 at (tj) then
           M[ti,tj] := true
         else
           M[ti,tj] := false
         end if

        end for (variable tj)
      end for (variable ti)
return M;
```

W powyższym pseudokodzie wprowadzono nazewnictwo zmiennych jak w formule przedstawiającej koncepcję metody. Różnica pomiędzy kodem a formułą (3.10) ma miejsce w indeksowaniu tablic reprezentujących dokumenty tekstowe. W formule indeksowanie zaczyna się od wartości 1, natomiast w kodzie od wartości 0. Wynika to z tego, że w większości środowisk programistycznych tablice indeksowane są od zera, co też w tym miejscu chciano zachować, jako ogólnie przyjętą regułę.

Poniżej na rysunku przedstawiono formę graficzną wyniku analizy macierzowej dwóch dokumentów tekstowych (fragmentu książki). W celu ułatwienia interpretacji algorytmu, oba dokumenty są takie same.



Rysunek 3.6. Graficzna interpretacja zaproponowanej metody macierzowej analizy dwóch tych samych fragmentów książki Daniela Defoe pt. „*The Life and Adventures of Robinson Crusoe*”

Osie pionowa i pozioma reprezentują analizowane dokumenty tekstowe. Liczby na osiach opisują numery porządkowe wyrazów, z których składa się dokument tekstowy. Punkty na rysunku to wyrazy wspólne dla obu dokumentów (w dalszej części rozdziału wyznaczane za pomocą miary podobieństwa bazującej na algorytmie odległości edycyjnej Levenshteina). Fragmenty tekstu podobne pomiędzy dokumentami, to te miejsca na rysunku, gdzie punkty występujące po sobie tworzą sekwencję. Ze względu na to, że powyższe dokumenty są identyczne, to na rysunku widoczna jest sekwencja punktów wyznaczająca diagonalę przecinającą macierz.

Na jakość wyniku analizy danych tekstowych, ma wpływ określenie dodatkowych parametrów, takich jak.:

- wybór metody wykrywania sekwencji punktów,
- określenie dopuszczalnych przerw pomiędzy punktami,
- określenie wartości progowej dla miary podobieństwa pomiędzy wyrazami,

Powyższe elementy zostaną szczegółowo opisane w kolejnych podrozdziałach.

3.4. Uwzględnienie algorytmu odległości edycyjnej w analizie macierzowej

W niniejszym rozdziale, algorytm odległości edycyjnej Levenshteina używany jest (tak jak to zostało szczegółowo opisane w poprzednich rozdziałach) jako alternatywa wobec popularnych algorytmów stemmingu i lematyzacji.

Poniższa formuła (3.12) przedstawia miejsce umieszczenia funkcji obliczającej miarę podobieństwa pomiędzy wyrazami, na podstawie której przypisywana jest odpowiednio wartość 0 lub 1 w elemencie macierzy w ramach danego kroku iteracji.

$$\underset{ti=1}{\overset{tm}{I}}\ \underset{tj=1}{\overset{tn}{I}}\ \boldsymbol{M}[ti,tj]=\beta \qquad (3.12)$$

$$\begin{cases}\beta = \text{true}: e(\boldsymbol{Doc1}[ti], \boldsymbol{Doc2}[tj]) \geq bp \\ \beta = \text{false}: e(\boldsymbol{Doc1}[ti], \boldsymbol{Doc2}[tj]) < bp\end{cases}$$

gdzie:

e – funkcja zwracająca wartość miary podobieństwa,

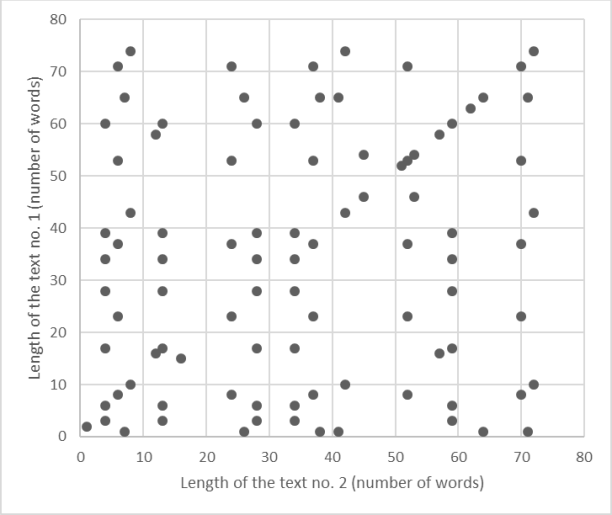
bp – wartość progowa określana przez użytkownika, będąca wyznacznikiem dla porównywanych wyrazów.

Przyjęte elementy konstrukcji macierzy dokumentów możemy zapisać w postaci pseudokodu jak poniżej (3.13). Nazewnictwo zmiennych i funkcji jest adekwatne do formuły przedstawionej powyżej, a zasada indeksowania tablic jest taka jak w poprzednim przykładzie.

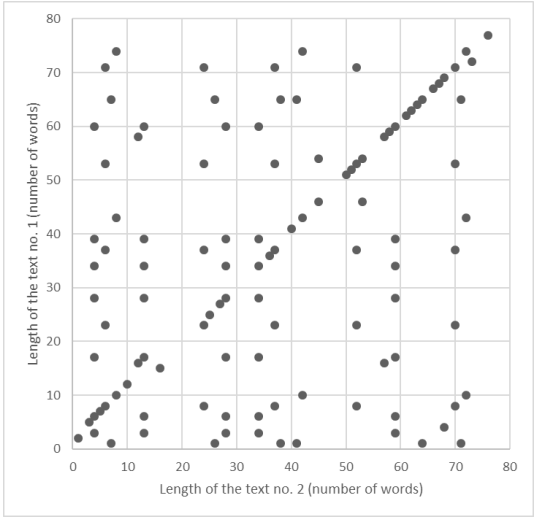
(3.13)

```
input variables: term Doc1[0..TM-1], term Doc2[0..TN-1],
                 decimal bp
   declare: boolean m[0..TM-1, 0..TN-1]
   for ti from 0 to TM-1
        for tj from 0 to TN-1

          if e(term of Doc1 at (ti), term of Doc2 at (tj)) >= bp then
            m[ti,tj] := true
          else
            m[ti,tj] := false
          end if

        end for (variable tj)
      end for (variable ti)
return m;
```

Efekt użycia miary podobieństwa w metodzie macierzowej analizy tekstu obrazuje prosty przykład poniżej. Zastosowano w nim do analizy dwa teksty napisane w dwóch językach: portugalskim i hiszpańskim. Dla tego przykładu wybrano te dwa języki, ponieważ oba należą do grupy języków romańskich i posiadają wspólne pochodzenie, które przejawia się istotnie w podobieństwie pomiędzy znaczną częścią wyrazów. Wyraźną różnicą pomiędzy tymi języki są natomiast reguły gramatyczne oraz wymowa, gdzie język hiszpański jest językiem melodyjnym, natomiast portugalski posiada wyraźny akcent zaczerpnięty z dialektu francuskiego i jest językiem trudniejszym do nauki.



Rysunek 3.7. Forma graficzna wyniku analizy podobieństwa pomiędzy krótkimi tekstami w językah portugalskim i hiszpańskim



Rysunek 3.8. Forma graficzna wyniku analizy podobieństwa pomiędzy krótkimi tekstami w językah portugalskim i hiszpańskim

Na rysunku 3.7 nie zastosowano wprowadzonej miary podobieństwa, wynik porównania tekstów wyniósł 30,77% podobieństwa. Na rysunku 3.8 użyto miarę podobieństwa bazującą na odległości Levenshteina, z wartością progową *bp*=0,75 i wynik wzrósł do 53,85% podobieństwa – tj. pomiędzy tekstami napisanymi w różnych językach, niosącymi to samo przesłanie.

Tekst poddany analizie w języku hiszpańskim jest następującej treści: 7.1(a). Tekst poddany analizie w języku portugalskim jest następującej treści: 7.1(b).

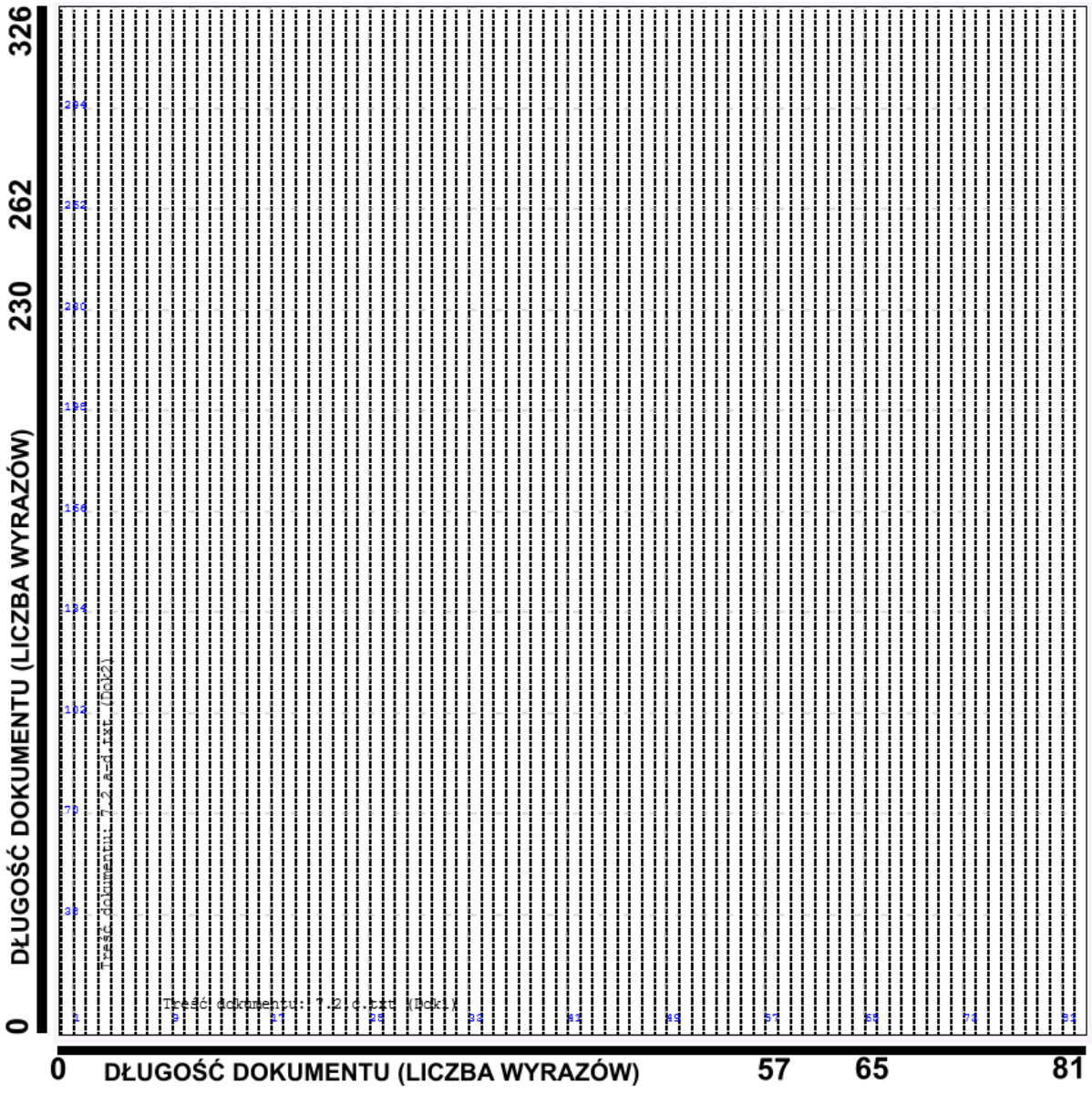
Jak widać na powyższych rysunkach (3.7-3.8) przedstawiających uzupełnione macierze punktami będącymi wyrazami uznanymi za tożsame w badanych tekstach, wykorzystanie miary podobieństwa daje możliwość pełniejszej analizy porównywanych tekstów. Podobieństwo na rysunku po prawej stronie (3.8) jest widoczne i to pomimo zauważalnych różnic pomiędzy wyrazami dla tych dwóch języków.

3.5. Parametryzacja analizy

W tej części pracy zaproponowane zostały dwie metody wykrywania sekwencji punktów, tj. grupowania punktów (skupień) oraz zachowania ciągłości zdań. Ich zastosowanie wymaga zbudowania algorytmu generowania raportu w postaci przedziałów fragmentów uznanych za podobne (w tym wydruku tych fragmentów), co stanowi dopełnienie całości mechanizmu macierzowej analizy dokumentów tekstowych.

3.5.1. Metody wykrywania zależności pomiędzy punktami

Metoda grupowania punktów jest najprostszą metodą umożliwiającą zobrazowanie podobieństwa pomiędzy dokumentami. Dla poszczególnych przypadków opisanych w dalszej części rozdziału, ustawiano różne parametry obliczeniowe, co bezpośrednio miało wpływ na wyniki (co widać na rysunkach). Teksty, które poddane zostały analizie to fragmenty: 7.2(a-d) i 7.2(c) oraz 7.3(a) i 7.3(b) (rozdział Załączniki).

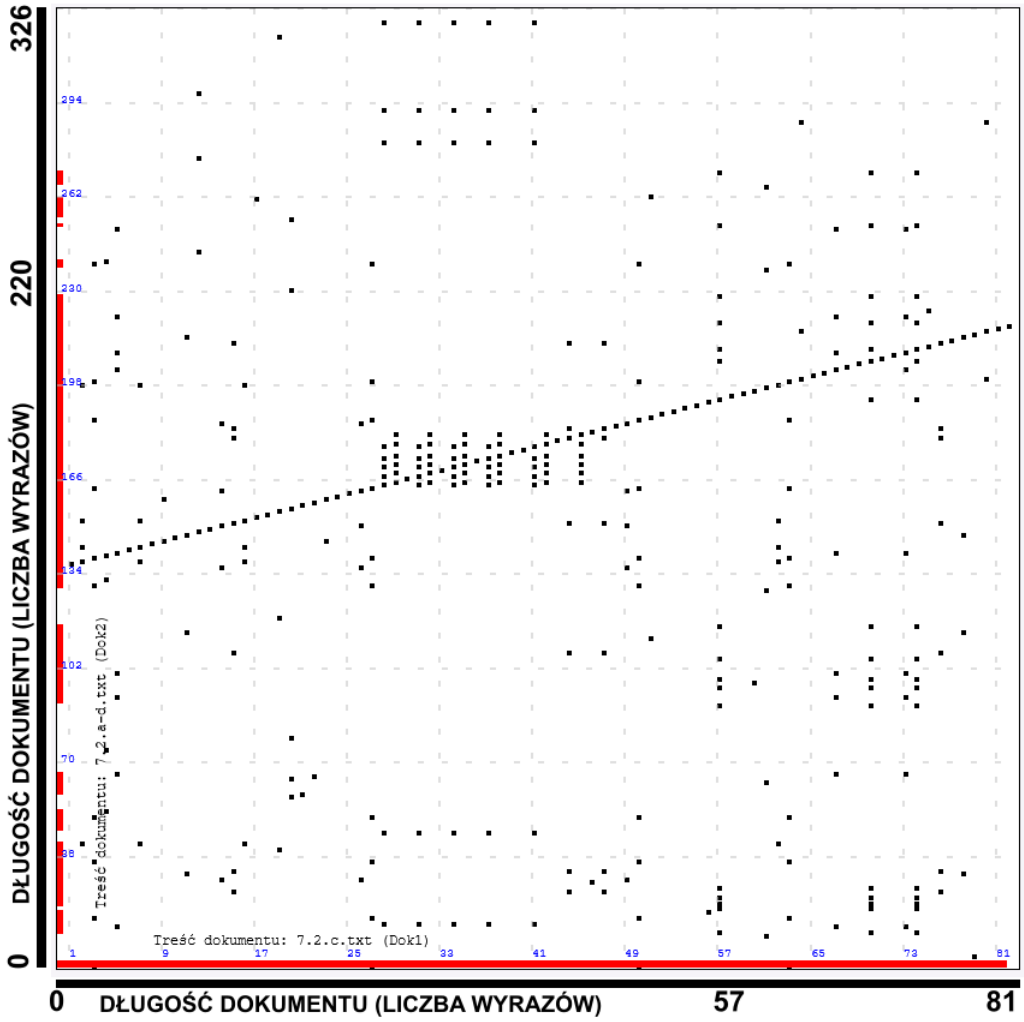


Rysunek 3.9. Wykres podobieństwa pomiędzy dwoma dokumentami tekstowymi 7.2(a-d) oraz 7.2(c) z załącznika. Wynik podobieństwa: 100% (dla dokumentu 7.2.c.txt – oś pozioma) wobec 100% (dla dokumentu 7.2.a-d.txt – oś pionowa)

Na rysunku powyżej (3.9) przedstawiono podobieństwo dokumentów dla parametru podobieństwa wyrazów *bp*=0%. Oznacza to, że badane wyrazy nie muszą być w ogóle do siebie

podobne, aby zostały uznane za takie same. Efektem takiej parametryzacji jest powyższy wykres, gdzie każdy wyraz uznany został za podobny do każdego innego. Parametry dotyczące wykrywania sekwencyjności punktów i grupowania zostały pominięte, aby uwidocznione zostały wszystkie punkty, tj. bez filtracji. Numeryczny wynik podobieństwa dokumentów to 100%.

W dalszej analizie badano podobieństwo dokumentów dla parametru podobieństwa wyrazów *bp*=100% (oznaczającego, że badane wyrazy muszą być identyczne wobec siebie, aby zostały uznane za takie same).



Rysunek 3.10. Wykres podobieństwa pomiędzy dwoma dokumentami tekstowymi 7.2(a-d) oraz 7.2(c) z załącznika. Wynik podobieństwa: 100% (dla dokumentu 7.2.c.txt – oś pozioma) wobec 46,93% (dla dokumentu 7.2.a-d.txt – oś pionowa)

Algorytm celowo pominął wyrazy składające się z jednego znaku oraz cyfr, co ma bezpośredni wpływ na liczby na wykresie (rys. 3.10) oznaczające kolejność wyrazów w dokumentach (dzięki zastosowaniu tego typu popularnej metody dokumenty są mniejsze od oryginalnych i dzięki

temu ich analiza przebiega szybciej). Punkty, które odzwierciedlają położenie takich samych wyrazów pomiędzy dokumentami nie zostały przefiltrowane, aby wykres mógł uwzględnić również pojedyncze występowanie terminów oraz aby na tle pozornie chaotycznie rozmieszonych punktów uwidocznić te, które tworzą sekwencję. Kolor czerwony na osiach oznacza pokrycie dokumentów podobnym lub identycznym tekstem.

Parametry dla obliczenia pokrycia dokumentów, to:

- maksymalna dopuszczalna przerwa pomiędzy wyrazami w celu zachowania ciągłości tekstu (oznaczenie używane w dalszej części pracy: *gw*[49]): 5 terminów w macierzy oraz
- minimalna wymagana liczba wyrazów w celu zbudowania wektora ciągłości tekstu (wzór 3.14) – oznaczenie używane w dalszej części pracy: *wv*[50]: 15.

Parametryzacja ta wpływa na brak koloru czerwonego dla wszystkich wspólnych punktów na wykresie. Wynik podobieństwa to 100% jednego dokumentu (uwzględnionego na osi poziomej) do 46,93% drugiego dokumentu (na osi pionowej), ponieważ jak widać bezpośrednio z wykresu, mniejszy dokument jest wycinkiem większego.

W ramach powyższej parametryzacji została wprowadzona koncepcja ciągłości tekstu. Ciągłość tekstu można zdefiniować za pomocą następującego wzoru (3.14):

$$\mathbf{T_j} = \left(t_{j,i}, t_{j,i+1}, \dots, t_{j,n-1}, t_{j,n}\right), i \in \langle 1, n_j \rangle \tag{3.14}$$

gdzie:
T – wektor ciągłości wyrazów (podobny fragment dokumentu do innego dokumentu),
j – numer wektora w wynikach porównania tekstów,
i – liczba naturalna oznaczająca kolejny wyraz w wektorze,
n_j – liczba naturalna oznaczająca liczbę wyrazów podobnych i niepodobnych w j-tym wektorze,
$t_{j,i}$ –wartość logiczna będąca wyznacznikiem podobieństwa konkretnych terminów pomiędzy dokumentami.

Maksymalna długość wektora **T** nie jest parametryzowana przez użytkownika. Maksymalna jego wielkość może być np. równa wielkości dokumentu, czyli liczbie wyrazów go tworzących. Jest to przypadek, gdy badane dokumenty są względem siebie identyczne. Na wektor **T** składają się wartości logiczne (1 – wyraz podobny lub 0 – wyraz niepodobny, czyli *true* lub *false*). Wartości te pochodzą z analizowanej macierzy przedstawionej we wzorze 3.12.

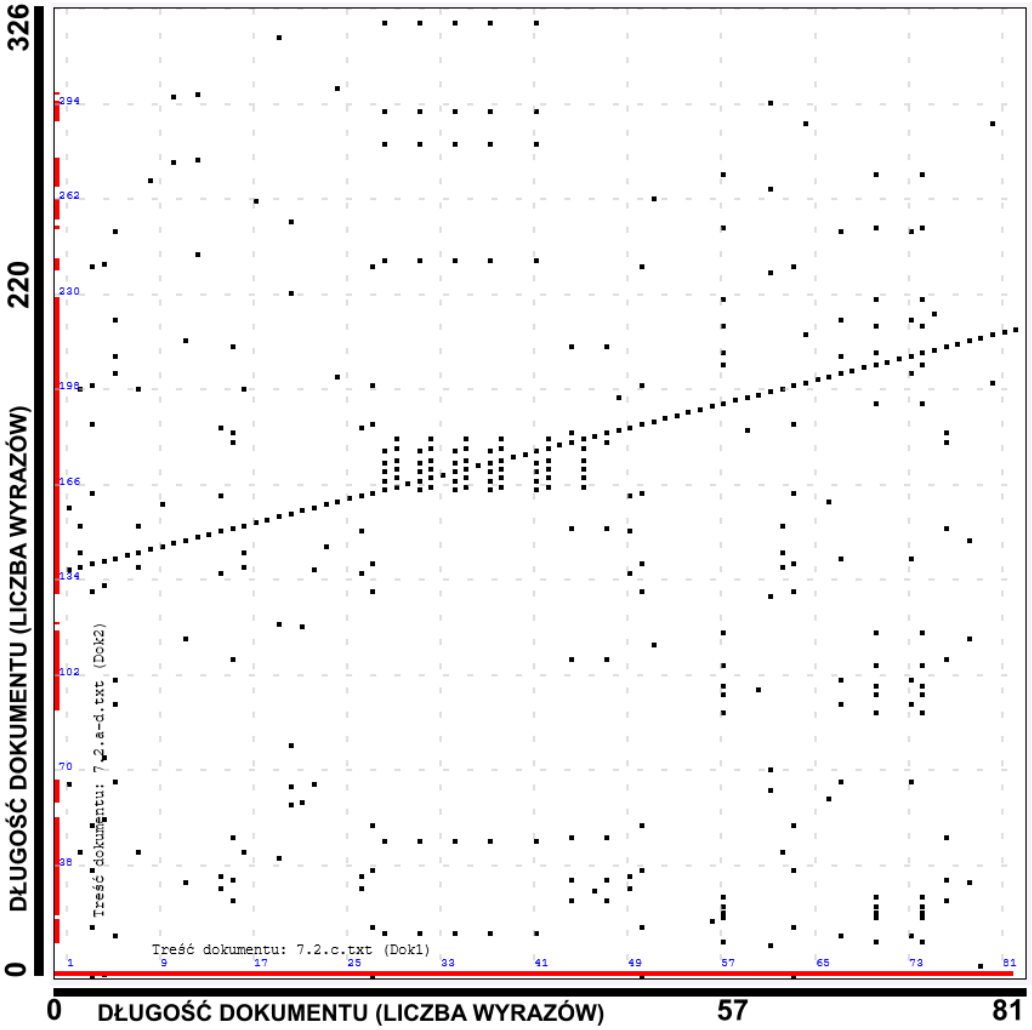
[49] maksymalna dopuszczalna przerwa pomiędzy wyrazami (ang. *maximum acceptable gap between words*)
[50] minimalna liczba wyrazów w wektorze sekwencji – (ang. *minimum number of words in sequence vector*)

Minimalna wielkość wektora wynika z parametru *wv* i jest określana (dostosowywana) przez użytkownika względem cech badanego tekstu (np. języka, w którym został napisany). Na liczbę *wv* składają się te wyrazy, które są do siebie podobne i występują zaraz po sobie lub z uwzględnieniem maksymalnej dopuszczalnej przerwy (tj. wyrazów niepodobnych) pomiędzy nimi określonej parametrem *gw*. Im mniejsza jest przerwa określona parametrem *gw* i im większa wielkość parametru *wv*, tym większe restrykcje nakładane są na analizę porównawczą tekstów – tzn. ciągi tekstowe muszą być bardziej dokładne, aby uznane zostały za tożsame. Metoda analizy macierzy bazująca na wektorze ciągłości wyrazów będzie wykorzystywana w dalszej części pracy do filtracji macierzy (m.in. w rozdz. 4), ponieważ jest najbardziej optymalna względem czasu wykonania analizy.

Wektory ciągłości wyrazów mają bezpośrednie przełożenie na oszacowanie fragmentów ciągów tekstowych ostatecznie uznanych za tożsame (pomiędzy dokumentami), które zawierają zarówno wyrazy identyczne, podobne, jak również różne (linie czerwone - pozioma i pionowa na wykresie).

Poniższy rysunek (3.11) przedstawia podobieństwo dokumentów dla parametru podobieństwa wyrazów 75%. Oznacza to, że badane wyrazy muszą spełniać kryterium podobieństwa *bp*=0.75, bazujące na formule opartej na algorytmie odległości edycyjnej opisanej we wcześniejszym rozdziale, aby zostały uznane za takie same – jest to różnica względem poprzedniego przykładu, gdzie wyrazy musiały być identyczne.

Efektem takiej parametryzacji jest poniższy wykres, który jest podobny (ale nie taki sam) do prezentowanego wcześniej. Oba wykresy różnią się niewielką liczbą punktów. Po obniżeniu progu podobieństwa *bp* do wartości 0.75 pojawiły się nowe punkty, ponieważ więcej wyrazów uznanych zostało za podobne do siebie.



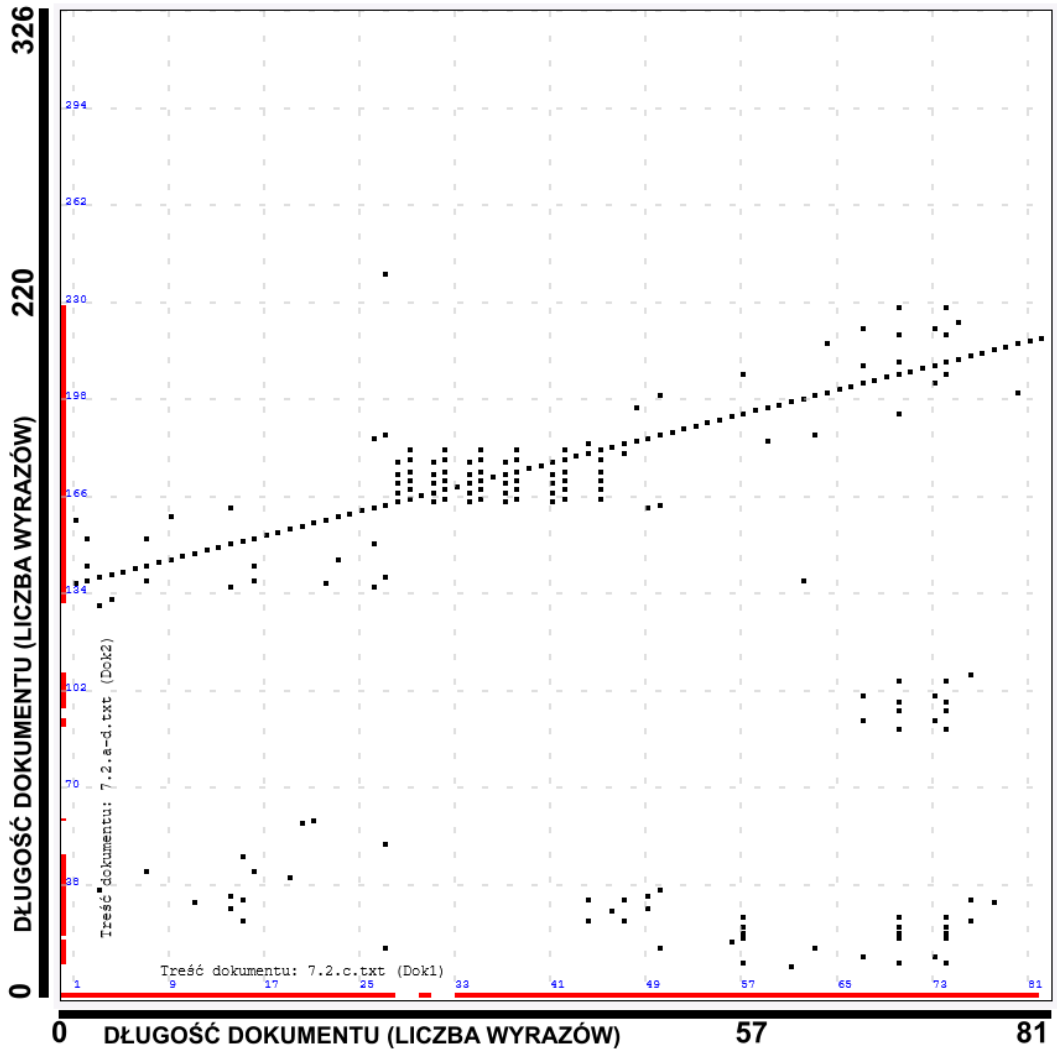
Rysunek 3.11. Wykres podobieństwa pomiędzy dwoma dokumentami tekstowymi 7.2(a-d) oraz 7.2(c) z załącznika. Wynik podobieństwa: 100% (dla dokumentu 7.2.c.txt – oś pozioma) wobec 50,31% (dla dokumentu 7.2.a-d.txt – oś pionowa)

Parametry dla obliczenia pokrycia dokumentów zastosowane w tym przykładzie, to:

- maksymalna dopuszczalna przerwa pomiędzy wyrazami w celu zachowania ciągłości tekstu: 5 terminów w macierzy oraz
- minimalna wymagana liczba wyrazów w celu zbudowania ciągłości tekstu: 15.

Wynik podobieństwa dokumentów zwiększył się dla drugiego i wynosi 50,31% (oś pionowa) do 100% (oś pozioma), co niekończenie jest efektem pożądanym. Przykłady w dalszej części rozdziału pokażą, że obok wartości progowej *bp*, równie ważnymi parametrami są: *gw* i *wv* oraz *P* i *R* – użyte w zależności od dobranej metody analizy punktów w macierzy.

Przeprowadzone badania na dokumentach wskazują, że najlepiej dobraną uniwersalną wartością progową *bp* jest 0.70-0.75, o czym będzie mowa w dalszej części pracy. Dlatego też, w większości przedstawionych dalej analiz, ta wartość będzie wynosiła 70% lub 75%.

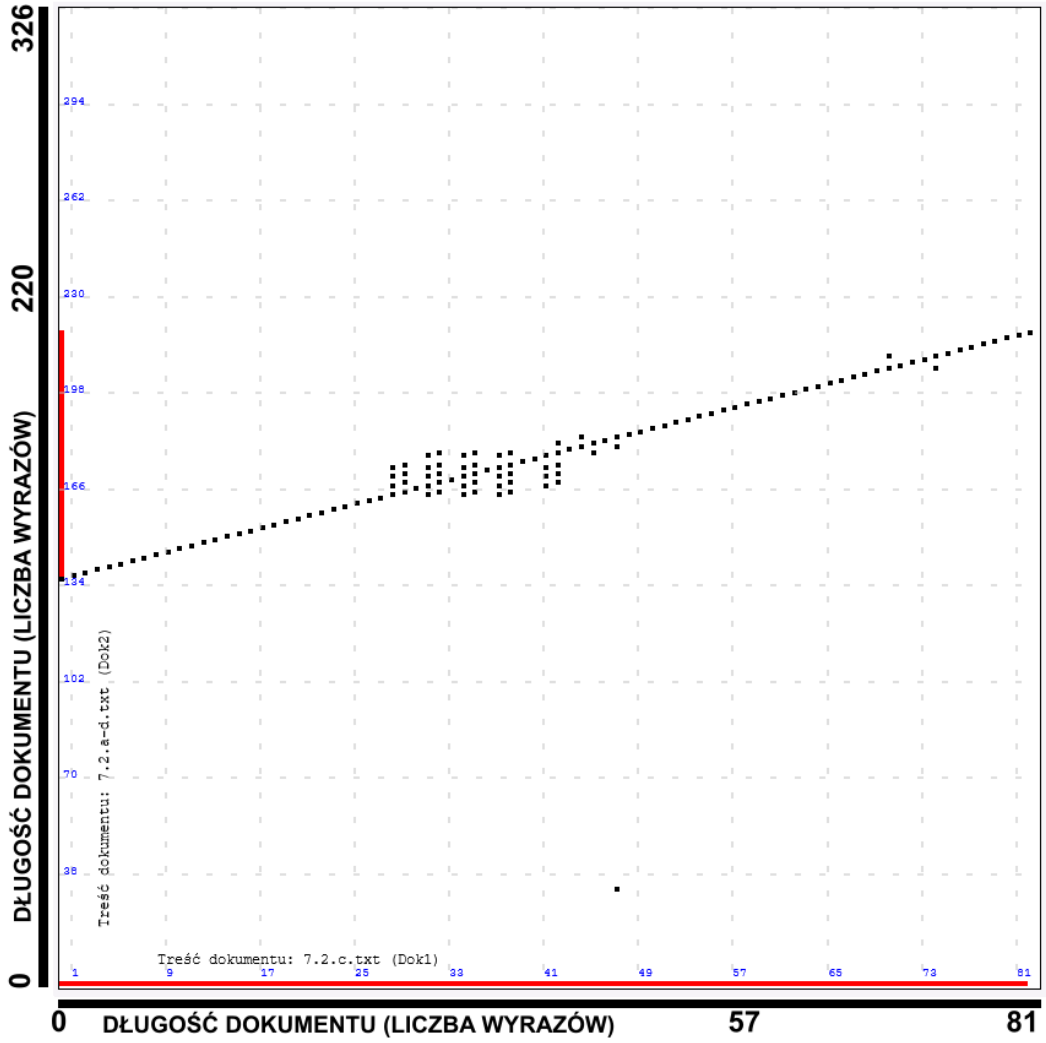


Rysunek 3.12. Wykres podobieństwa pomiędzy dwoma dokumentami tekstowymi 7.2(a-d) oraz 7.2(c) z załącznika. Wynik podobieństwa: 100% (dla dokumentu 7.2.c.txt – oś pozioma) wobec 36,5% (dla dokumentu 7.2.a-d.txt – oś pionowa)

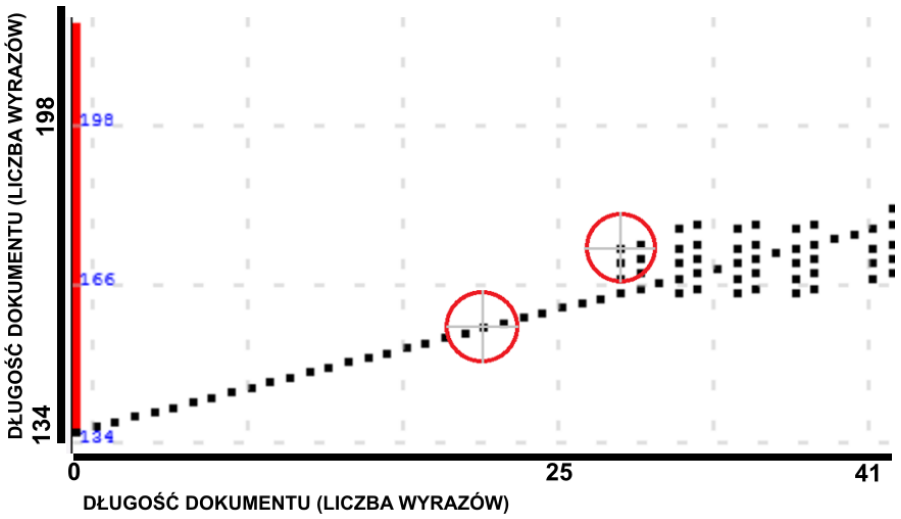
Powyższy rysunek (3.12) przedstawia wynik podobieństwa dokumentów dla parametru *bp* równego 75% w ramach drugiej fazy analizy polegającej na grupowaniu punktów. Grupowanie punktów jest realizowane za pomocą metody, która polega na zadaniu promienia analizowanego otoczenia dla danego punktu i wymaganej (minimalnej) liczbie punktów w tym promieniu, aby ten punkt mógł pozostać widoczny. Dla powyższego przykładu są to odpowiednio wartości: 8 i 3, które wybrane zostały tak, aby pokazać stopniowe oczyszczenie się rysunku z szumu (grupy pojedynczych wyrazów - punktów na rysunku), względem przykładu na rys. 3.11. Dalsza korekta tych wartości sprawi, że wynik będzie bardziej czytelny i poprawny.Jak widać na powyższym wykresie (3.12), gdzie liczba punktów jest znacznie

mniejsza względem poprzedniego przykładu, pozostały te punkty, które nie są skrajnie osamotnione, czyli tworzą grupę kilku punktów. Ostatecznie wykres jest bardziej przejrzysty, wyszczególnione zostały części na macierzy, które przedstawiają wspólne fragmenty dokumentów.

Poniższy rysunek (3.13) przedstawia podobieństwo dokumentów dla parametru podobieństwa wyrazów *bp*=75% oraz opisanej powyżej metody grupowania z wartościami: promień R=3 oraz minimalna liczba punktów w promieniu *P*=3. Efektem analizy jest powyższy wykres, na którym wyszczególniona została linia prosta reprezentująca podobieństwo całości tekstu krótszego wobec tekstu dłuższego. Niewielkie fragmenty składającego się z pojedynczych punktów, które można było zaobserwować w poprzednim przykładzie zostały usunięte.



Rysunek 3.13. Wykres podobieństwa pomiędzy dwoma dokumentami tekstowymi 7.2(a-d) oraz 7.2(c) z załącznika. Wynik podobieństwa: 100% (dla dokumentu 7.2.c.txt – oś pozioma) wobec 25,77% (dla dokumentu 7.2.a-d.txt – oś pionowa)



Rysunek 3.14. Wykres podobieństwa pomiędzy dwoma dokumentami tekstowymi (powiększenie skali rys. 3.13). Wynik podobieństwa: 100% wobec 25,77%

Dodatkowo na rysunku powyżej przedstawiono powiększenie środkowej części wykresu analizy z poprzedniego przykładu, na którym widać pionowe sekwencje punktów. Są one wynikiem powtarzającego się kilkukrotnie fragmentu tekstu: „his”. Czerwone okręgi uwidaczniają działanie algorytmu z uwzględnieniem zadanych parametrów. Linie przecinające się, to analizowany punkt, w jednym z okręgów widać pionowe punkty stanowiące promień *R*=3, w każdym z okręgów jest wymagana minimalna liczba punktów *P*=3.

Innym podejściem do analizy punktów na macierzy jest metoda opisana w daszlej części rozdziału. Polega ona na poszukiwaniu sekwencji występujących po sobie wyrazów z uwzględnieniem możliwych przerw pomiędzy terminami. Tutaj obok parametru określającego granicę podobieństwa (bp) pomiędzy wyrazami są dodatkowo zasttosowane: maksymalna dopuszczalna przerwa pomiędzy wyrazami (gw) oraz minimalna dopuszczalna liczba wyrazów w celu utworzenia wektora sekwencji wyrazów (wv).

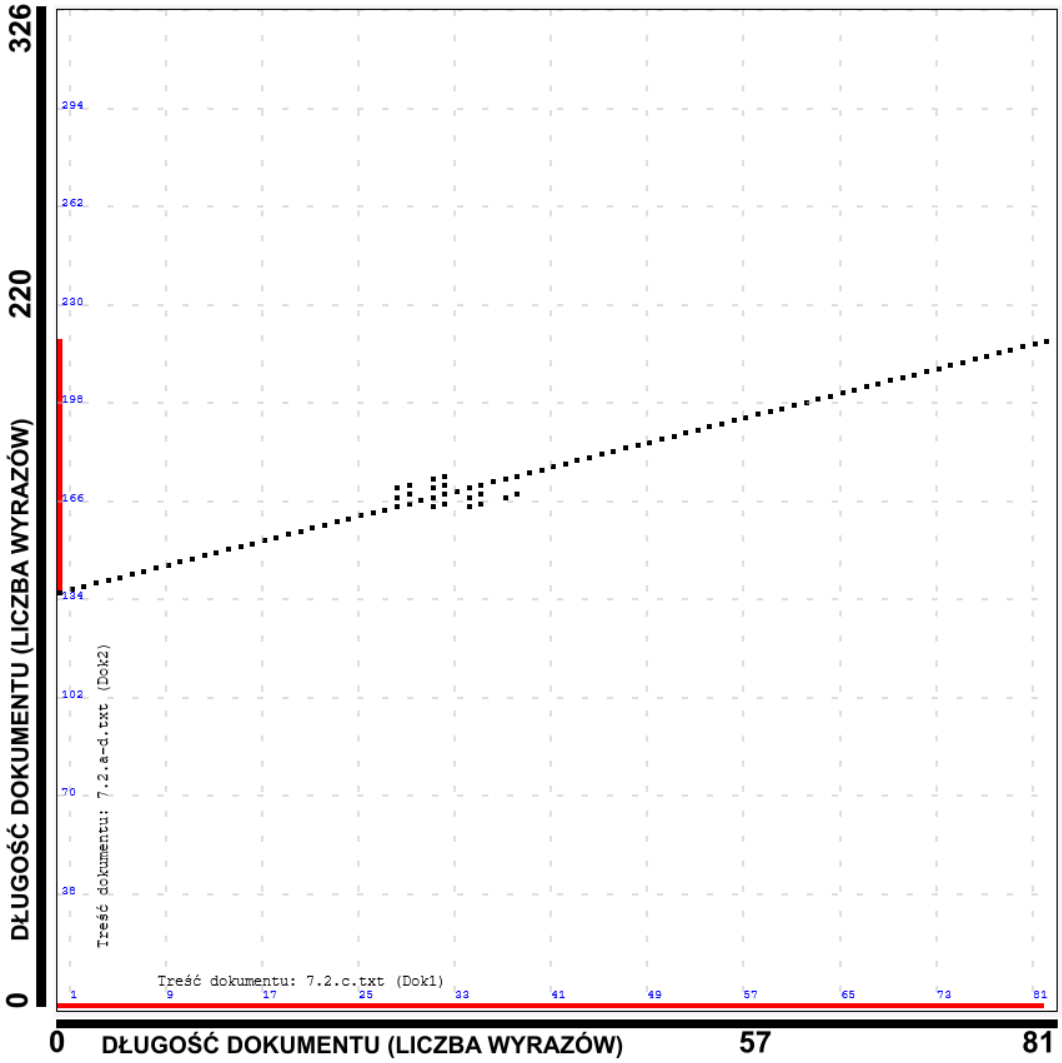
Poniższy rysunek (3.15) przedstawia wynik analizy podobieństwa dwóch dokumentów tekstowych, w której została użyta metoda poszukiwania sekwencji wyrazów. Teksty zostały dołączone do pracy w ramach rozdziału 7 (Załączniki).

Parametry analizy to:

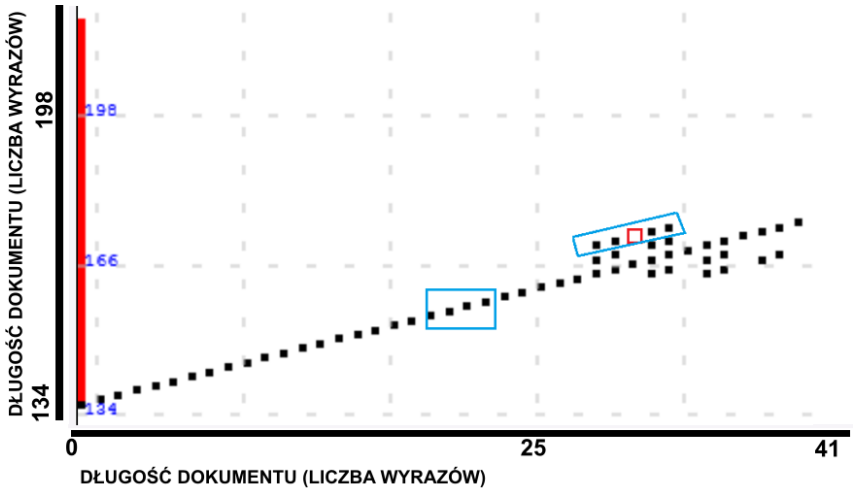
- maksymalna dopuszczalna przerwa pomiędzy wyrazami: *gw*=1

- minimalna dopuszczalna liczba wyrazów w celu utworzenia wektora sekwencji wyrazów: *wv*=4

- podobieństwo wyrazów: *bp*=75%.



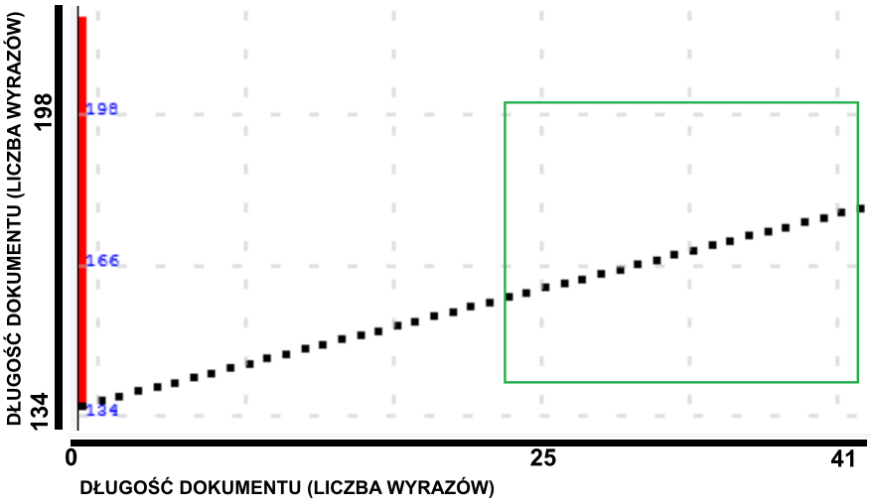
Rysunek 3.15. Wykres podobieństwa pomiędzy dwoma dokumentami tekstowymi 7.2(a-d) oraz 7.2(c) z załącznika. Wynik podobieństwa: 100% (dla dokumentu 7.2.c.txt – oś pozioma) wobec 25,46% (dla dokumentu 7.2.a-d.txt – oś pionowa)



Rysunek 3.16. Wykres podobieństwa pomiędzy dwoma dokumentami tekstowymi (powiększenie skali rys. 3.15). Wynik podobieństwa: 100% wobec 25,46%

Powyższy rysunek (3.16) przedstawia powiększenie środkowej części wykresu z rysunku 3.15, na którym widać pionowe sekwencje punktów. Podobnie jak w poprzednich przykładach (ze względu na te same analizowane teksty), są one wynikiem powtarzającego się kilkukrotnie fragmentu tekstu: „his”. Kolor niebieski oznacza utworzony wektor ciągłości wyrazów (wspólny fragment analizowanego tekstu uznany za podobny) z minimalnej dopuszczalnej liczby wyrazów, a kolor czerwony oznacza dopuszczalną przerwę pomiędzy punktami (czyli niepasujący wyraz pomiędzy wyrazami podobnymi), w celu utworzenia wektora. Mniejsze wektory składają się na większe lub w szczególnych przypadkach na jeden duży wektor.

Jeżeli dla powyższego przypadku parametr *wv* zostałby zwiększony do liczby większej od 4, np. *wv*=6, wtedy z wykresu zostałyby usunięcie mniejsze wektory, a wykres stałby się bardziej czytelny, aczkolwiek z wyniku zostałyby usunięte mniejsze powtarzające się fragmenty tekstu uznane za podobne (rys. 3.17).

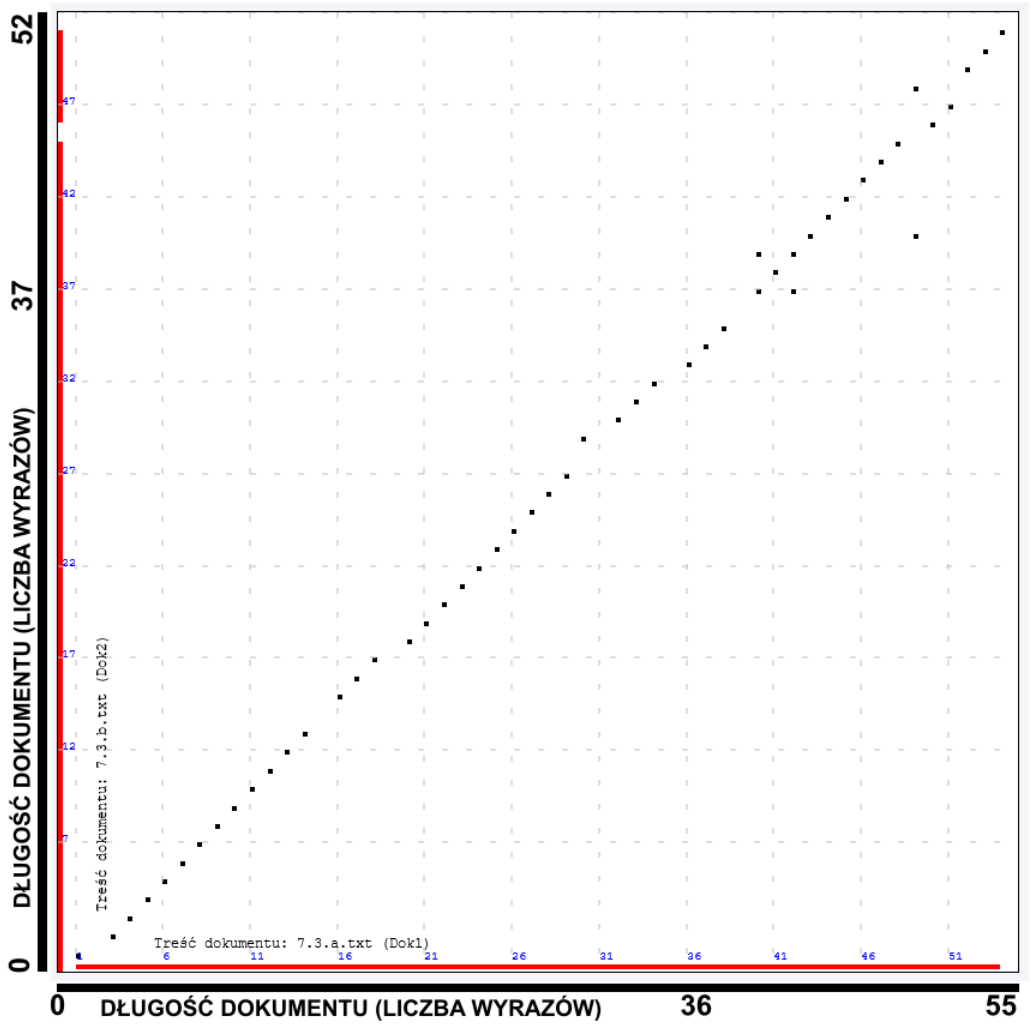


Rysunek 3.17. Wykres podobieństwa pomiędzy dwoma dokumentami tekstowymi (powiększenie skali). Wynik podobieństwa: 100% wobec 25,46%

W ramach analizy została użyta metoda poszukiwania sekwencji wyrazów. Parametry analizy to:

- maksymalna dopuszczalna przerwa pomiędzy wyrazami: *gw*=1
- minimalna dopuszczalna liczba wyrazów w celu utworzenia wektora sekwencji wyrazów: *wv*=6
- podobieństwo wyrazów: *bp*=75%.

Zielony kolor oznacza miejsce, w którym znajdowały się punkty będące wynikiem poprzedniej analizy, gdzie parametr *wv*=4. Żaden z poprzednich punktów (rys. 3.16), zgodnie z parametrem *wv=6*, nie mógł współtworzyć wektora ciągłości wyrazów podobnych.

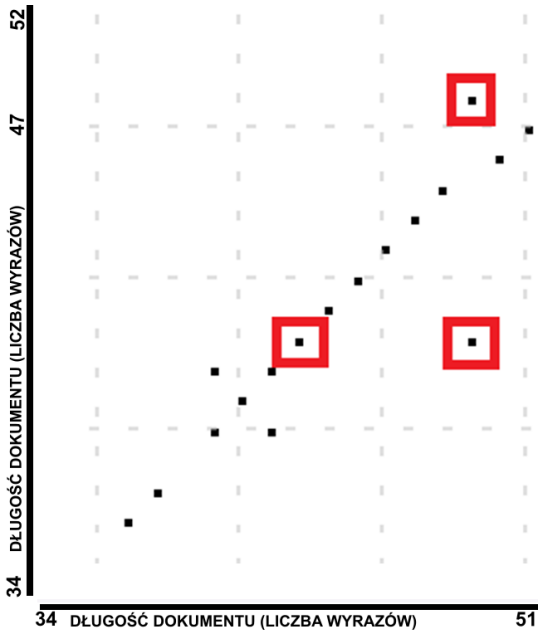


Rysunek 3.18. Wykres podobieństwa pomiędzy dwoma dokumentami tekstowymi 7.3(a) oraz 7.3(b) z załącznika. Wynik podobieństwa: 89,09% (dla dokumentu 7.3.a.txt – oś pozioma) wobec 94,23% (dla dokumentu 7.3.b.txt – oś pionowa)

Wynikiem graficznym analizy tekstów metodą grupowania punktów jest rysunek 3.18. Na wykresie widać punkty oddalone od głównej linii. Jest to wynikiem parametryzacji oraz samej metody. W analizie poszukiwania sekwencji punktów, takie elementy zostałyby uznane jako szum (czyli terminy niepotrzbne) i usunięte z macierzy.

Parametry analizy, której wynikiem graficznym jest rysunek 3.18:

- prób podobieństwa wyrazów *bp*=70%
- promień R=4
- minimalna liczba punktów w promieniu *P*=3.



Rysunek 3.19. Wykres zależności pomiędzy dwoma dokumentami tekstowymi (powiększenie skali rys. 3.18). Wynik podobieństwa: 89,09% wobec 94,23%

Odchylenie punktów od głównej diagonali jest wynikiem przestawienia miejscami odpowiednich wyrazów w tekście 7.3(b) względem oryginału, co przedstawiono na rysunku (3.19). Ze względu na wprowadzone parametry analizy, punkty te nie zostały usunięte i wliczają się do wyniku podobieństwa pomiędzy tekstami.

4. Weryfikacja przedstawionego mechanizmu analizy danych tekstowych

Ten rozdział ma na celu przedstawienie szczegółowej weryfikacji opracowanego mechanizmu analizy danych tekstowych. W ramach licznych testów udowodniona zostanie teza, że za pomocą opracowanej metody macierzowej analizy danych tekstowych bazującej na algorytmie odległości edycyjnej można porównywać teksty bez konieczności używania algorytmów lematyzacji, stemmingu oraz słownika wyrazów bliskoznacznych. Przedstawione zostaną wyniki eksperymentów, które dostarczą wglądu w skuteczność i precyzję zaproponowanego rozwiązania. Rozdział zakończy się dyskusją na temat dalszych kierunków badań i rozwoju analizowanego mechanizmu, aby zapewnić jego optymalizację pod kątem potrzeb różnych dziedzin i kontekstów. Wyniki zostały wygenerowane przez komercyjny program antyplagiatowy o nazwie N-DMS Antyplagius[51] firmy New Data Mining Systems sp. z.o.o.[52], który powstał na bazie zaprezentowanych w pracy rozwiązań algorytmicznych. Dla niektórych eksperymentów zostały udostępnione filmy zamieszczonych na platformie YouTube, do których podane zostały linki[53]. Filmy nie są integralną częścią niniejszej pracy, jednakże zostały dołączone w celu uzupełnienia informacji. Mogą one okazać się przydatne dla osób zainteresowanych konkretną analizą przedstawioną w pracy lub tematyką dotyczącą badanych języków. Niektóre filmy celowo bazują na innym zestawie danych, niż ten w danym podrozdziale. Celem takiego podejścia jest celowe poszerzenie zakresu analizowanego materiału i dostarczenie szerszego kontekstu na temat omawianych języków.

Porównane zostały teksty pod kątem podobieństwa w ramach tych samych języków, jak również teksty napisane w językach pokrewnych, wywodzących się z tych samych grup językowych. Ma to na celu wykazanie skuteczności, w tym przede wszystkim czułości i adaptacyjności mechanizmu do różnych języków, bez rzeczywistej konieczności implementacji reguł gramatycznych.

Dane tekstowe poddane analizie pochodzą z różnych źródeł, w tym z popularnych encyklopedii internetowych oraz opracowanych przez sztuczną inteligencję. Różne wersje językowe tych samych artykułów są efektem tłumaczenia przez translatory języków obcych lub wynikiem opracowania artykułów przez chatGPT. Translatory automatyczne bazują na

[51] Adres internetowy programu Antyplagius: https://antyplagius.n-dms.com
[52] Adres internetowy firmy: https://n-dms.com
[53] Adres internetowy kanału programu Antyplagius na platformie YouTube: https://www.youtube.com/playlist?list=PLPFeTDhxdQPawnjGhPytgFJeJb-YOXmHC

technologii, która analizuje tekst źródłowy i generuje tłumaczenie w języku docelowym, zwykle za pomocą zaawansowanych algorytmów i sztucznej inteligencji.

Translatory języka obcego użyte w eksperymentach to:

- Google Translate (Tłumacz Google) - jest to najbardziej znany i powszechnie używany translator języka obcego. Oferuje tłumaczenia między wieloma językami, zarówno w formie tekstowej, jak i mówionej. Działa na podstawie zaawansowanych algorytmów opartych na uczeniu maszynowym i sztucznej inteligencji, co pozwala na generowanie tłumaczeń o coraz wyższej jakości.
- DeepL - to inny popularny translator języka obcego, który także korzysta z zaawansowanych technologii uczenia maszynowego i sztucznej inteligencji. DeepL oferuje tłumaczenia między wybranymi językami, ze szczególnym naciskiem na jakość tłumaczeń.
- Microsoft Translator - jest to usługa tłumaczenia języka obcego opracowana przez Microsoft, która oferuje tłumaczenia między wieloma językami w formie tekstowej, mówionej oraz obsługuje tłumaczenia na żywo w rozmowach. Microsoft Translator jest zintegrowany z wieloma innymi produktami Microsoft, takimi jak Office, Skype czy system operacyjny Windows.
- chatGPT OpenAI - dzięki zaawansowanym modelom językowym, oferuje wysokiej jakości tłumaczenia języków obcych. Model ten jest w stanie tłumaczyć teksty pomiędzy wieloma językami, zachowując przy tym kontekst i znaczenie oryginału. Dzięki głębokiemu uczeniu maszynowemu, ChatGPT rozumie niuanse językowe, idiomy oraz specyficzne struktury gramatyczne, co pozwala na tworzenie tłumaczeń, które są nie tylko poprawne, ale także naturalne i płynne.

Wszystkie przedstawione w rozdziale obliczenia w ramach pojedynczych analiz trwały mniej niż 0,2 sek. dlatego czas obliczeń nie został uwzględniony w tabelach.

4.1. Analiza dokumentów tekstowych napisanych w językach: hiszpańskim i portugalskim[54]

Języki hiszpański i portugalski należą do wspólnej rodziny języków romańskich i wywodzą się z łaciny. Mają swoje korzenie na Półwyspie Iberyjskim, gdzie ewoluowały z języków używanych przez różne plemiona i narody. Język hiszpański, jest najbardziej rozpowszechnionym językiem romańskim na świecie. Jest oficjalnym językiem w Hiszpanii oraz większości krajów Ameryki Łacińskiej. Pochodzi z dialektów łacińskich używanych przez Celtów i Iberów na Półwyspie Iberyjskim, a następnie przez Wizygotów, których język również wpłynął na kształtowanie się hiszpańskiego. W wyniku rekonkwisty, język ten rozprzestrzenił się na południowe obszary półwyspu, podbijane od Maurów, co także przyczyniło się do wzbogacenia słownictwa i gramatyki hiszpańskiego.

Język portugalski, drugi pod względem liczby użytkowników język romański na świecie, jest oficjalnym językiem Portugalii, Brazylii, a także niektórych krajów Afryki i Azji. Język portugalski wyodrębnił się z języka galicyjskiego, innego języka romańskiego, który miał swoje korzenie w północno-zachodniej Hiszpanii i północnej Portugalii. W miarę jak Portugalczycy odkrywali i kolonizowali nowe terytoria, ich język rozprzestrzeniał się na inne kontynenty. Hiszpański i Portugalski są dzisiaj ważnymi językami na świecie, zarówno pod względem liczby użytkowników, jak i wpływów kulturowych.

Język hiszpański i portugalski mają wiele wspólnych słów i wyrażeń, ponieważ oba te języki mają swoje korzenie w językach romańskich i łacinie. Poniżej znajduje się kilka przykładów słów, które są podobne w obu językach:

- "Amigo" (hiszpański) / "Amigo" (portugalski) - Przyjaciel
- "Casa" (hiszpański) / "Casa" (portugalski) - Dom
- "Buenos días" (hiszpański) / "Bom dia" (portugalski) - Dzień dobry
- "Hermano" (hiszpański) / "Irmão" (portugalski) - Brat
- "Hablar" (hiszpański) / "Falar" (portugalski) - Mówić
- "Comer" (hiszpański) / "Comer" (portugalski) - Jeść
- "Libro" (hiszpański) / "Livro" (portugalski) - Książka

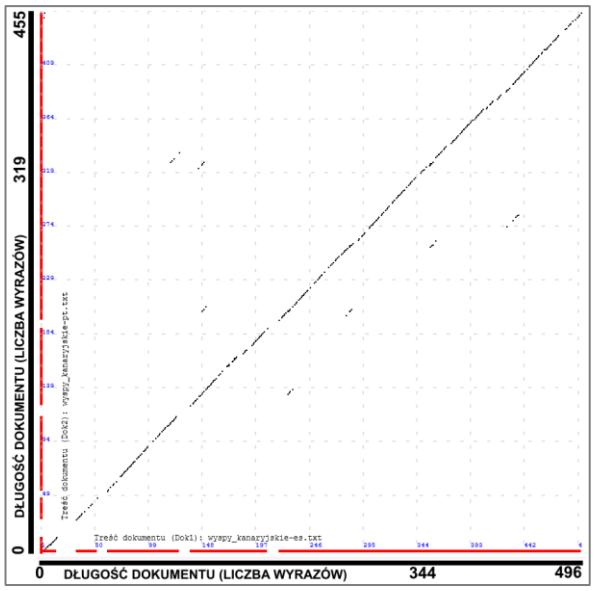
[54] Więcej przykładów analizy tych języków znajduje się na stronie oficjalnego kanału programu Antyplagius: https://www.youtube.com/watch?v=TjmUBKAARkM&list=PLPFeTDhxdQPawnjGhPytgFJeJb-YOXmHC&index=3

- "Ciudad" (hiszpański) / "Cidade" (portugalski) - Miasto
- "Mujer" (hiszpański) / "Mulher" (portugalski) - Kobieta

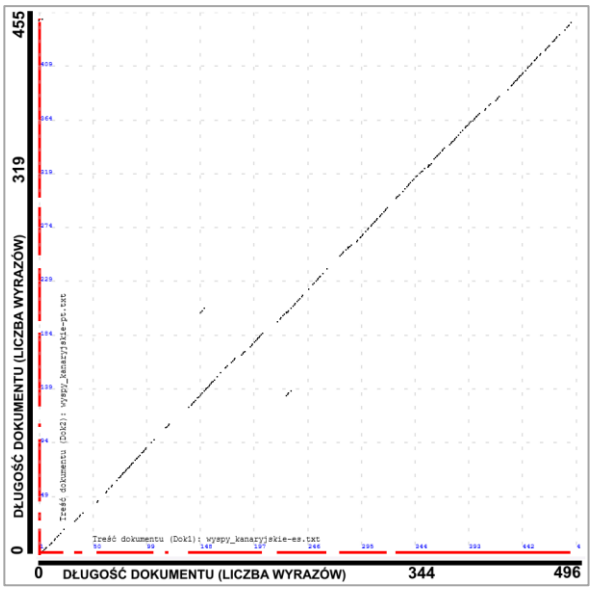
TEST. Artykuł o Wyspach Kanaryjskich

Artykuł przedstawia opis Wysp Kanaryjskich – archipelagu należącym do Hiszpanii. Znajdują się tam informację o geografii, klimacie, historii, kulturze, gospodarce, a także przyrodzie i działań na rzecz ochrony środowiska. Tekst w języku polskim (rozdz. 7.5.1)[55] przetłumaczony został na dwa języki: hiszpański (rozdz. 7.5.3)[56] oraz portugalski (rozdz. 7.5.2)[57] za pomocą jednego z najpopularniejszych zaawansowanych modeli językowych opracowanych przez firmę OpenAI – ChatGPT[58] w wersji 4. Graficzna interpretacja porównanych tekstów znajduje się poniżej. Stała *gw* będzie w większości przypadków podobna dla wszystkich testów w rozdziale, ponieważ specyfika problemu nie wymusza jej ciągłego dostosowywania do tekstu. Stała *gw* jest przewidziana do analizy tekstów, gdzie istnieje znaczne podobieństwo próby przekłamania treści poprzez celową zmianę struktury zdań, w tym zmianę kolejności terminów, usuwanie wyrazów i wstawiania odpowiedników w postaci synonimów. Wartość została dobrana w ramach wcześniejszych badań nad tekstami napisanymi w różnych językach, ale w ramach tych samych grup językowych [39,47].

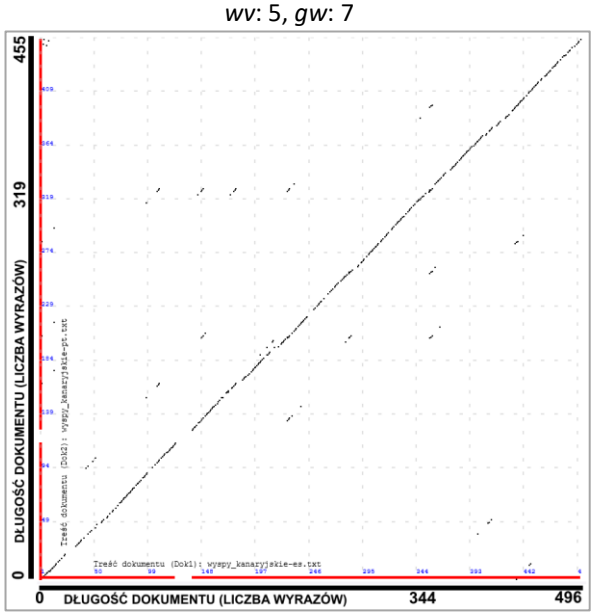
[55] Treść dostępna pod adresem: https://antyplagius.n-dms.com/tests/Spanish-Portuguese/wyspy_kanaryjskie-pl.txt
[56] Treść dostępna pod adresem: https://antyplagius.n-dms.com/tests/Spanish-Portuguese/wyspy_kanaryjskie-es.txt
[57] Treść dostępna pod adresem: https://antyplagius.n-dms.com/tests/Spanish-Portuguese/wyspy_kanaryjskie-pt.txt
[58] Strona projektu: https://chat.openai.com/



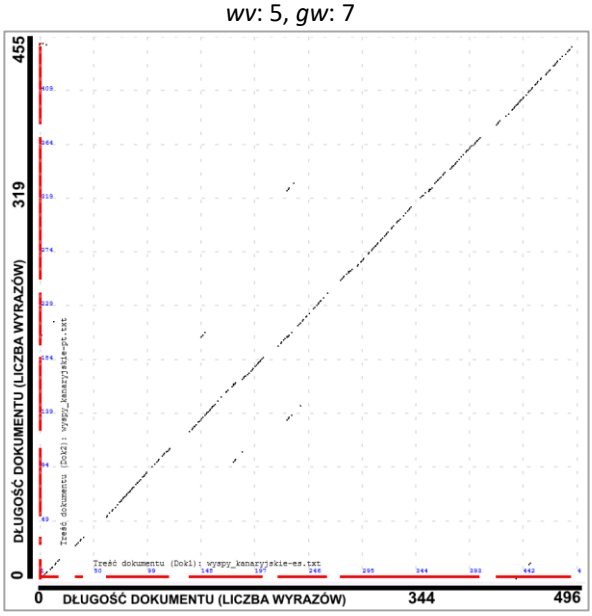
Rysunek 4.1. Parametry analizy – *bp*: 50%, *wv*: 5, *gw*: 7



Rysunek 4.2. Parametry analizy – *bp*: 60%, *wv*: 5, *gw*: 7



Rysunek 4.3. Parametry analizy – *bp*: 50%, *wv*: 5, *gw*: 10



Rysunek 4.4. Parametry analizy – bp: 60%, *wv*: 5,*gw*: 10

ID	Język dokumentu (1)	Język dokumentu (2)	*Bp*	*wv*	*gw*	*Liczba wyrazów (liczba znaków) (1) x (2)*	WYNIK (1)	WYNIK (2)
1	Hiszpański	Portugalski	50%	5	7	496 X 455 (2791 X 2602)	65,66%	72,03%
2	Hiszpański	Portugalski	60%	5	7	496 X 455 (2791 X 2602)	54,14%	59,47%
3	Hiszpański	Portugalski	50%	5	10	496 X 455 (2791 X 2602)	71,11%	77,09%
4	Hiszpański	Portugalski	60%	5	10	496 X 455 (2791 X 2602)	54,55%	59,69%

Tabela 4.1. Tabela przedstawiająca wyniki kilku analiz porównania tekstów, z różnymi parametrami analizy

Powyższe wyniki (tabela 4.1 oraz rysunki 4.1-4.4) należy rozpatrywać przede wszystkim pod kątem tezy, jaka została postawiona w rozdziale. Z powyższych wykresów widać, że teksty wykazują znaczne podobieństwo, a w niektórych przypadkach niemalże identyczność – świadczy o tym linia przebiegająca po ukosie macierzy. Na podstawie rysunków (4.1-4.4) i tabeli (4.1) widać, że najlepszy rezultat wykazujący zachodzące podobieństwo pomiędzy tekstami w językach hiszpańskim i portugalskim uzyskuje się poprzez ustawienie granicy podobieństwa wyrazów na poziomie wyższym od 60% (*bw*) i minimalna liczba wyrazów w wektorze sekwencji (*wv*) oraz maksymalna dopuszczalna przerwa pomiędzy wyrazami (*gw*) pozostają na niezmienionym poziomie i wynoszą odpowiednio 5 i 7-10 (rys. 4.2,4.4). Ustawienie wartości podobieństwa wyrazów (*bw*) poniżej tego progu (rys. 4.3) sprawia, że graficzny rezultat porównania jest mniej czytelny, pojawia się szum. Zmniejszenie stałej *bw* do 0% będzie generowało wynik 100% podobieństwa pomiędzy dokumentami – co będzie oczywistym błędem i wynika z zasady działania algorytmu – tzn. każdy badany wyraz będzie uznawany za tożsamy z innym. Przerwy należy rozumieć, jako umieszczone pomiędzy terminami inne wyrazy lub zmiana szyku wyrazów, wynikająca przede wszystkim z różnic pomiędzy językami. Dlatego też, ze względu na to, że są to różne języki, zmienne *wv* i *gw* w większości analiz były ustawione według podanych wartości i nie zmieniane.

ID	Język hiszpański	Język portugalski
1	[...] economía del archipiélago. Cultura La cultura contemporánea de las Islas Canarias es una mezcla de influencias españolas y tradiciones nativas guanches. La música local, la danza como el tajaraste, un baile y música rítmicos y los festivales reflejan este patrimonio. El español es el idioma oficial, [...]	[...] economia do arquipélago. Cultura A cultura contemporânea das Ilhas Canárias é uma mistura de influências espanholas e tradições nativas guanches. A música local, a dança como o tajaraste, uma dança e música rítmicas e os festivais refletem esse patrimônio. O espanhol é a língua oficial, [...]
2	[...] las autoridades locales, especialmente en el contexto de la creciente presión derivada del desarrollo turístico intensivo. Conclusión Las Islas Canarias, con sus diversos paisajes, rica historia y cultura, así como su clima constante, representan un destino fascinante. Para muchos, son sinónimo de escape de la rutina diaria, ofreciendo tanto aventura como relajación. Su importancia como centro de investigación científica y conservación de la naturaleza subraya su valor no solo para [...]	[...] das autoridades locais, especialmente no contexto da crescente pressão derivada do desenvolvimento turístico intensivo. Conclusão As Ilhas Canárias, com suas diversas paisagens, rica história e cultura, bem como seu clima constante, representam um destino fascinante. Para muitos, são sinônimo de fuga da rotina diária, oferecendo tanto aventura quanto relaxamento. Sua importância como centro de pesquisa científica e conservação da natureza destaca seu valor não apenas para [...]

Tabela 4.2. Fragmenty tekstów uznanych za podobne

W tabeli 4.2 przedstawiono wybrane fragmenty uznane za podobne będące wynikiem analizy ID 4 (tab. 4.1) porównania omawianych tekstów. Każdy z powyższych wyrazów uznany za

podobny do swojego odpowiednika w drugim tekście ma swoją interpretację graficzną w postaci punktu na macierzy (rys. 4.4). Natomiast wyrazy, które nie są podobne, ale pojawiły w ciągu tekstowym poprzez odpowiednie ustawienie parametrów *wv* i *gw* są uwzględnione na rysunkach w ramach czerwonych linii obok osi poziomej i pionowej wraz z wyrazami podobnymi – razem stanowią wspomniany wcześniej wektor ciągłości wyrazów. Innymi słowy, przykładowe fragmenty tekstów umieszczonych w tabeli 4.2., składające się z wyrazów podobnych i różnych, mają swoje pokrycie w czerwonych liniach na rysunkach.

4.2. Analiza dokumentów tekstowych napisanych w językach: czeskim i słowackim[59]

Język czeski i słowacki to dwa blisko spokrewnione języki słowiańskie, które mają wspólne korzenie w prasłowiańskim, języku używanym przez Słowian we wczesnym średniowieczu. Język czeski jest oficjalnym językiem Czech i jest używany przez około 10 milionów osób. Wywodzi się z języka staroczeskiego, który był używany na terenie Czech od IX do XI wieku. W trakcie swojego rozwoju, język czeski uległ wpływom innych języków, takich jak łacina, niemiecki i rosyjski. Dzisiejszy czeski jest zróżnicowany dialektalnie, ale większość ludności posługuje się standardową formą tego języka.

Język słowacki, z kolei, jest oficjalnym językiem Słowacji i jest używany przez około 5 milionów osób. Podobnie jak czeski, słowacki ma swoje korzenie w językach słowiańskich używanych na terenie Słowacji od IX wieku. W trakcie swojego rozwoju, słowacki również uległ wpływom innych języków, zwłaszcza łaciny i niemieckiego. Choć słowacki posiada kilka dialektów, standardowy język słowacki, oparty na środkowo-słowackim dialekcie, jest powszechnie używany przez mówiących tym językiem.

Mimo że język czeski i słowacki mają wiele wspólnych cech gramatycznych i leksykalnych, różnią się nieco pod względem fonetyki, słownictwa i wymowy. W praktyce, osoby mówiące po czesku i słowacku zazwyczaj mogą się wzajemnie zrozumieć, chociaż istnieją pewne różnice i tzw. „fałszywi przyjaciele", które mogą utrudnić komunikację. „Fałszywy przyjaciel" to termin używany w językoznawstwie, który odnosi się do słów w dwóch różnych językach, które wyglądają lub brzmią podobnie, ale mają różne znaczenia. Fałszywe

[59] Więcej przykładów analizy tych języków znajduje się na stronie oficjalnego kanału programu Antyplagius: https://www.youtube.com/watch?v=e5jVSQ_4lHo&list=PLPFeTDhxdQPawnjGhPytgFJeJb-YOXmHC&index=5

przyjaciółki mogą występować na różnych poziomach językowych, takich jak gramatyka, słownictwo czy wymowa. Przykładem fałszywego przyjaciela między czeskim a słowackim może być słowo "konečně" (czeski) i "konečne" (słowacki), które mają podobną wymowę i pisownię, ale różne znaczenia. W języku czeskim "konečně" oznacza "wreszcie" lub "w końcu", podczas gdy w języku słowackim "konečne" oznacza "ostatecznie" lub "definitywnie".

Poniżej znajduje się kilka przykładów słów, które są podobne w obu językach:

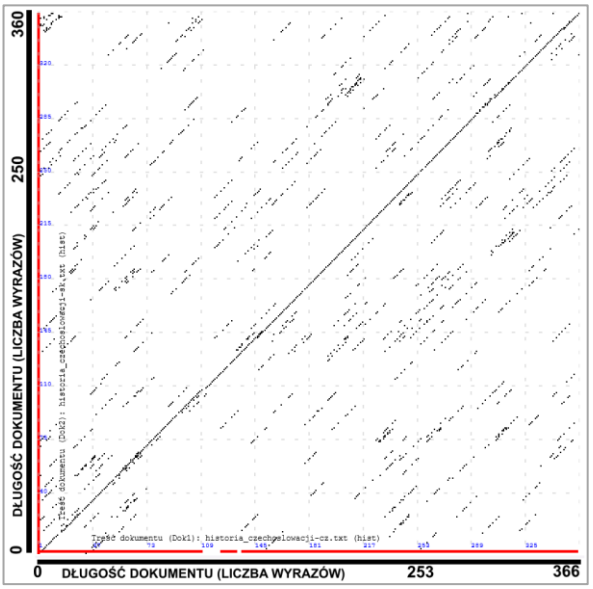
- "Dobrý den" (czeski) / "Dobrý deň" (słowacki) - Dzień dobry
- "Děkuji" (czeski) / "Ďakujem" (słowacki) - Dziękuję
- "Přítel" (czeski) / "Priateľ" (słowacki) - Przyjaciel
- "Dům" (czeski) / "Dom" (słowacki) - Dom
- "Bratr" (czeski) / "Brat" (słowacki) - Brat
- "Mluvit" (czeski) / "Hovoriť" (słowacki) - Mówić
- "Jíst" (czeski) / "Jesť" (słowacki) - Jeść
- "Kniha" (czeski) / "Kniha" (słowacki) - Książka
- "Město" (czeski) / "Mesto" (słowacki) - Miasto
- "Žena" (czeski) / "Žena" (słowacki) – Kobieta

TEST. Analiza artykułu o Czechosłowacji

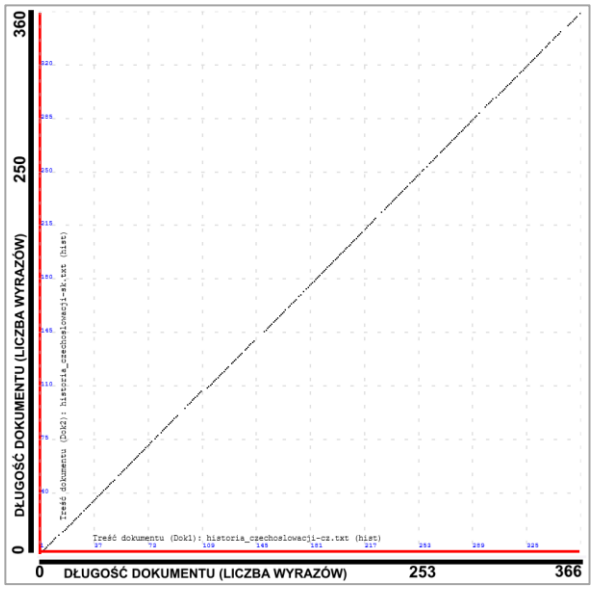
Artykuł przedstawia historię Czechosłowacji, od jej powstania w 1918 roku po rozpad w 1993 roku. Tekst w języku polskim (rozdz. 7.6.1)[60] przetłumaczony został na dwa języki: czeski (7.6.2)[61] oraz słowacki (rozdz. 7.6.3)[62] za pomocą zaawansowanego modelu językowego opracowanego przez firmę OpenAI – ChatGPT w wersji 4. Następnie teksty zostały między sobą porównane, a graficzną interpretację porównanych tekstów przedstawiono poniżej. Parametr *gw* przyjęto na takim samym poziomie dla wszystkich testów w rozdziale, ponieważ specyfika tekstów nie wymusza jego zmiany. Wartość *gw*=8 została również dobrana na tym poziomie

[60] Treść dostępna pod adresem: https://antyplagius.n-dms.com/tests/Czech-Slovak/historia_czechoslowacji-pl.txt
[61] Treść dostępna pod adresem: https://antyplagius.n-dms.com/tests/Czech-Slovak/historia_czechoslowacji-cz.txt
[62] Treść dostępna pod adresem: https://antyplagius.n-dms.com/tests/Czech-Slovak/historia_czechoslowacji-sk.txt

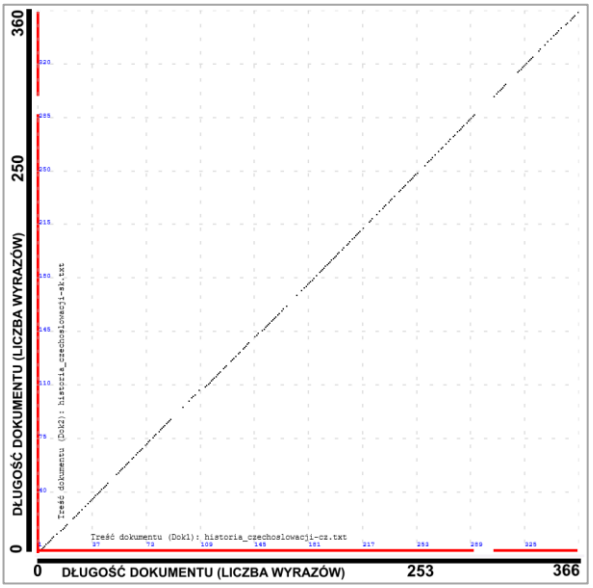
ze względu na wczesniejsze badania nad tekstami napisanymi w różnych językach, ale w ramach tych samych grup językowych [39,47].



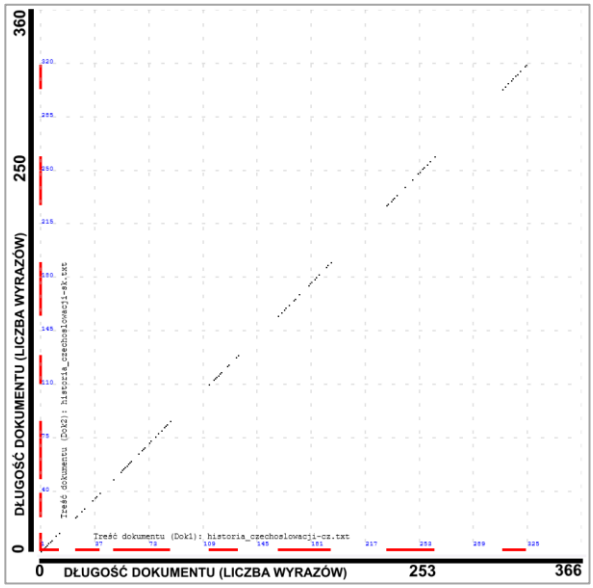
Rysunek 4.5. Parametry analizy – *bp*: 20%, *wv*: 7, *gw*: 8



Rysunek 4.6. Parametry analizy – *bp*: 50%, *wv*: 7, *gw*: 8



Rysunek 4.7. Parametry analizy – *bp*: 70%, *wv*: 7, *gw*: 8



Rysunek 4.8. Parametry analizy – bp: 100%, *wv*: 7,*gw*: 8

ID	Język dokumentu (1)	Język dokumentu (2)	*bp*	*wv*	*gw*	*Liczba wyrazów (liczba znaków) (1) x (2)*	WYNIK (1)	WYNIK (2)
1	Czeski	Słowacki	20%	7	8	366 X 360 (2302 X 2337)	100%	100%
2	Czeski	Słowacki	50%	7	8	366 X 360 (2302 X 2337)	85,60%	87,04%
3	Czeski	Słowacki	70%	7	8	366 X 360 (2302 X 2337)	66,76%	67,89%
4	Czeski	Słowacki	100%	7	8	366 X 360 (2302 X 2337)	25,76%	26,20%

Tabela 4.3. Tabela przedstawiająca wyniki analiz przy różnych wartościach parametru *bp*

ID	Język czeski	Język słowacki
1	[...] Historie Československa Od vzniku po rozpad Vznik Československa Československo, stát existující v letech –, bylo založeno po pádu RakouskoUherska na konci první světové války. Nová republika byla vyhlášena . října a zahrnovala historická území Čech, Moravy, Slezska, Slovenska a Podkarpatské Rusi.[...]	[...] História Československa Od vzniku po rozpad Vznik Československa Československo, štát existujúci v rokoch –, vznikol po páde RakúskoUhorska na konci prvej svetovej vojny. Nová republika bola vyhlásená . októbra a zahŕňala historické územia Čiech, Moravy, Sliezska, Slovenska a Podkarpatskej Rusi.[...]
2	[...] Tento stát, přes mnohé výzvy, byl většinu své existence místem relativní stability a rozvoje, ačkoliv nakonec nedokázal uniknout rozpadu, který se stal závěrečným aktem jeho bouřlivé historie. [...]	[...] Tento štát, napriek mnohým výzvam, bol väčšinu svojej existencie miestom relatívnej stability a rozvoja, hoci nakoniec neunikol rozpadu, ktorý sa stal záverečným aktom jeho búrlivej histórie. [...]

Tabela 4.4. Przykłady fragmentów tekstu uznanych za podobne dla analizy ID 2 z tabeli 4.3 i rys. 4.6

Przedstawione wyżej rezultaty (tabele 4.3 i 4.4 oraz rysunki 4.5-4.8) pokazują, że teksty wykazują znaczne podobieństwo, a w niektórych przypadkach wręcz identyczność – niezależnie od różnicy w wielkości zmiennej *bw*. Na podstawie rysunków i tabeli widać, że najlepszy rezultat wykazujący zachodzące podobieństwo pomiędzy tekstami w językach czeskim i słowackim uzyskuje się poprzez ustawienie podobieństwa wyrazów (*bw*) na poziomie 50%-70% (rys. 4.6-4.7). Jednakże ustawienie wartości podobieństwa wyrazów (*bw*) poniżej tego progu sprawia, że graficzny rezultat porównania jest mniej czytelny, pojawia się szum, a linia ukośna, która odpowiada za wizualne potwierdzenie podobieństwa tekstów, jest mniej widoczna (rys. 4.5). W tym przypadku podwyższenie wyniku podobieństwa pomiędzy dokumentami nie jest efektem satysfakcjonującym, ale mylącym. Analogicznie do poprzednich badanych przypadków, przerwy na rysunku należy rozumieć, jako umieszczone pomiędzy terminami inne wyrazy lub zmianę szyku wyrazów, wynikające z różnic pomiędzy językami. Z rysunku 4.8 można dodatkowo wywnioskować, że badane języki są do siebie bardzo podobne – powodem jest ukształtowana dosyć wyraźnie linia pomimo parametru *bp*=100%.

4.3. Analiza dokumentów tekstowych napisanych w językach: białoruskim i ukraińskim[63]

W państwach europejskich używa się następujących języków pisanych cyrylicą: Rosyjski - Federacja Rosyjska (Rosja europejska), Ukraiński - Ukraina, Białoruski - Białoruś, Bułgarski - Bułgaria, Serbski - Serbia, Macedoński - Macedonia Północna, Czarnogórski – Czarnogóra. Ten podrozdział dotyczy porównania dwóch wybranych z powyższych – białoruskiego i ukraińskiego, przede wszystkim ze względu na bliskie sąsiedztwo, liczbę ludności, jak również sąsiedztwo z Rosją, której język ma silny wpływ na te państwa. Język białoruski (znany także jako *беларуская мова*, *bielaruskaja mova*), jest językiem indoeuropejskim z grupy wschodniosłowiańskiej, do której należą również rosyjski i ukraiński [34,36]. Język ukraiński i białoruski to dwa wschodniosłowiańskie języki, które mają wspólne korzenie w językach prasłowiańskich używanych przez Słowian we wczesnym średniowieczu.

Język ukraiński jest oficjalnym językiem na Ukrainie i jest używany przez około 40 milionów osób. Ukraiński wywodzi się z języka staroruskiego, który był używany na terenie Kijowskiej Rusi od IX do XIV wieku. W trakcie swojego rozwoju, język ukraiński uległ wpływom innych języków, takich jak polski, rosyjski i turecki.

Język białoruski jest oficjalnym językiem na Białorusi i jest używany przez około 6-7 milionów osób. Podobnie jak ukraiński, białoruski ma swoje korzenie w językach staroruskich używanych na terenie Wielkiego Księstwa Litewskiego. Białoruski również uległ wpływom innych języków, takich jak polski, rosyjski i litewski.

Mimo że język ukraiński i białoruski mają wspólne korzenie, różnią się pod względem gramatyki, słownictwa i wymowy. Niemniej jednak istnieją wspólne elementy, które wskazują na ich bliskie pokrewieństwo:

- Analizowane języki używają alfabetu cyrylicy, choć z nieco różnymi wariantami liter.
- Mają zbliżone systemy gramatyczne, takie jak deklinacje rzeczowników, odmiany czasowników, przysłówki, przymiotniki itp.

[63] Więcej przykładów analizy tych języków znajduje się na stronie oficjalnego kanału programu Antyplagius: https://www.youtube.com/watch?v=d6o3QAQDWPk&list=PLPFeTDhxdQPawnjGhPytgFJeJb-YOXmHC&index=2

- Często dzielą podobne słownictwo, choć z pewnymi różnicami, np. "хліб" (ukraiński) / "хлеб" (białoruski) - chleb, "добрий день" (ukraiński) / "добры дзень" (białoruski) - dzień dobry, "дякую" (ukraiński) / "дзякуй" (białoruski) - dziękuję.

Różnice pomiędzy językami białoruskim i ukraińskim występują zarówno na poziomie fonetyki, gramatyki, jak i leksyki. Oto kilka z nich:

- fonetyka - jednym z kluczowych rozróżnień między białoruskim a ukraińskim jest różnica w akcencie. Na przykład, w języku białoruskim zachowano tak zwane "akanie". Tego typu zjawisko nie występują w języku ukraińskim.
- gramatyka - choć oba języki mają siedem przypadków gramatycznych, istnieją pewne różnice w ich użyciu i formach. Ukraiński, podobnie jak polski, posiada formę dopełniacza liczby mnogiej dla rzeczowników męskich, której nie ma w białoruskim.
- leksyka - wiele słów jest różnych w obu językach, choć zwykle są one wzajemnie zrozumiałe dla mówców obu języków. Białoruski otrzymał silne wpływy z języka polskiego w wyniku historycznych kontaktów, podczas gdy ukraiński ma więcej wspólnego z językami zachodniosłowiańskimi, takimi jak czeski i słowacki, choć również odcisnęła na nim swoje piętno bliska współpraca z Polską.
- alfabet - oba języki używają cyrylicy, ale ukraiński ma kilka unikalnych liter, które nie występują w białoruskim, takich jak ґ, ї, є i й . Natomiast język białoruski posiada literę: Ў (u z kreską).
- dialekty - język ukraiński ma wiele dialektów, które różnią się znacznie między różnymi regionami Ukrainy. Białoruski, z drugiej strony, jest bardziej jednolity, choć istnieją pewne różnice między różnymi regionami Białorusi.

Oba języki są blisko spokrewnione i wzajemnie zrozumiałe do pewnego stopnia, ale mają swoje unikalne cechy i różnice. Poniższe analiza odpowie na następujące pytanie – czy za pomocą algorytmu komputerowego niezawierającego dedykowanych reguł gramatycznych, dla języków wschodniosłowiańskich, w tym braku zaimplementowanych metod stemmingu i lematyzacji, da się przeprowadzić skuteczną analizę porównawczą pomiędzy tekstami napisanymi w językach: białoruskim i ukraińskim, przy okazji w ramach wspólnej słowiańskiej grupy językowej. Zwłaszcza w kontekście istotnych różnic jakie zostały przedstawione.

Ciągi tekstowe poddane analizie pochodzą ze źródeł internetowych w postaci artykułów encyklopedycznych z dwóch różnych znanych encyklopedii. Poddane zostały tłumaczeniu przez translatory bazujące na rozwiązaniach z dziedziny sztucznej inteligencji, które obecnie uznawane są za wyjątkowo skuteczne. Poniżej opisane zostaną trzy testy. Pierwszy test to dostosowanie jednego z języków do drugiego poprzez przetłumaczenie tego pierwszego, a następnie analiza ich podobieństwa. Drugi test rozpoczyna się przetłumaczeniem wersji artykułu napisanego w języku angielskim na dwa badane języki. Powyższe podejścia są nawiązaniem do badań jakie mają miejsce w kwestiach związanych z plagiatami typu *cross-language*, popełnianymi coraz częściej na świecie w szkołach oraz na uczelniach[37,38]. Trzeci test polega na analizie podobieństwa tekstów napisanych w dwóch językach będących tematem publikacji (przetłumaczonych z języka angielskiego) z kilkoma wybranymi językami posługującymi się cyrylicą.

Wizualizacja tego typu analizy zamieszczona jest w postaci filmu na kanale YouTube[64], gdzie przedstawione zostały kroki wykonywane w programie dla zidentyfikowania wyniku podobieństwa tekstów. Jest to analiza dwóch artykułów encyklopedycznych o Polskiej Akademii Nauk napisanych w badanych językach, bez tłumaczenia ich przez translatory.

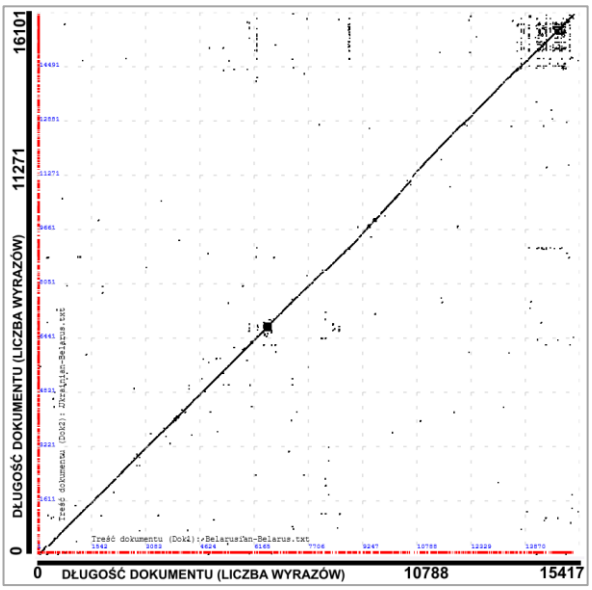
W ramach poniższych testów wszystkie badane teksty zostały zamieszczone w ramach zasobów zenętrznych, do których prowadzą linki w odnośnikach, ze względu na zbyt dużą objętność.

TEST 1. Artykuł encyklopedyczny o Białorusi

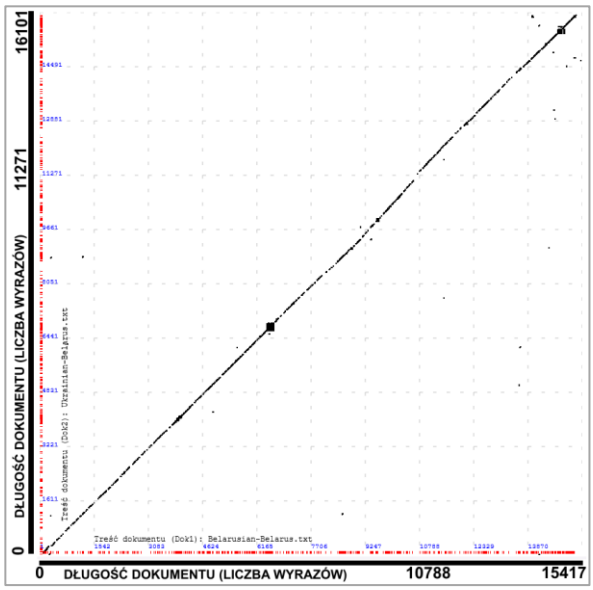
W tej analizie użyty został artykuł encyklopedyczny o Białorusi[65] (adres WWW strony encyklopedii dostępny jest w odnośniku[66]) w języku białoruskim[67] przetłumaczony przez translator na język ukraiński[68]. Teksty zostały między sobą porównane. Graficzna interpretacja porównanych tekstów znajduje się poniżej (rys. 4.9-4.14). Stała *gw* będzie w większości

[64] https://youtu.be/d6o3QAQDWPk
[65] https://antyplagius.n-dms.com/tests/Belarusian-Ukrainian/Belarus-Wikipedia.pdf
[66] https://be.wikipedia.org/wiki/%D0%91%D0%B5%D0%BB%D0%B0%D1%80%D1%83%D1%81%D1%8C
[67] Treść dostępna pod adresem: https://antyplagius.n-dms.com/tests/Belarusian-Ukrainian/Belarusian-Belarus.txt
[68] Treść dostępna pod adresem: https://antyplagius.n-dms.com/tests/Belarusian-Ukrainian/Ukrainian-Belarus.txt

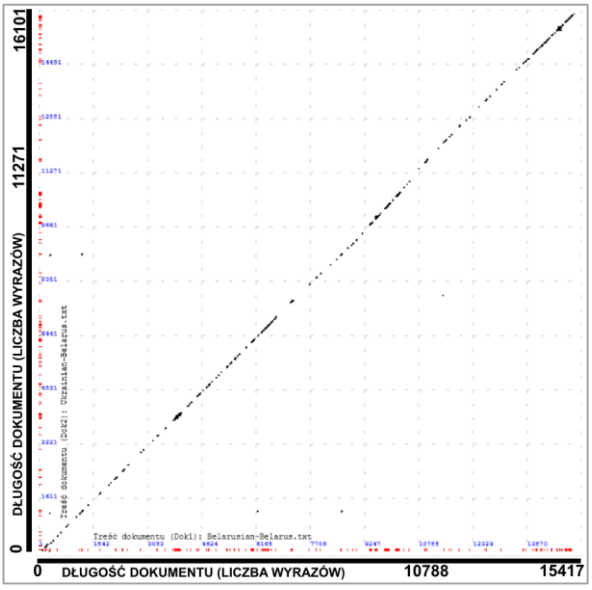
przypadków taka sama dla wszystkich testów w rozdziale (*gw*=8), ponieważ dla tych języków nie trzeba jej dostosowywć każdorazowo do tekstu. Wynika to z wcześniejszych badań nad tekstami napisanymi w różnych językach, ale w ramach tych samych grup językowych [39,47].



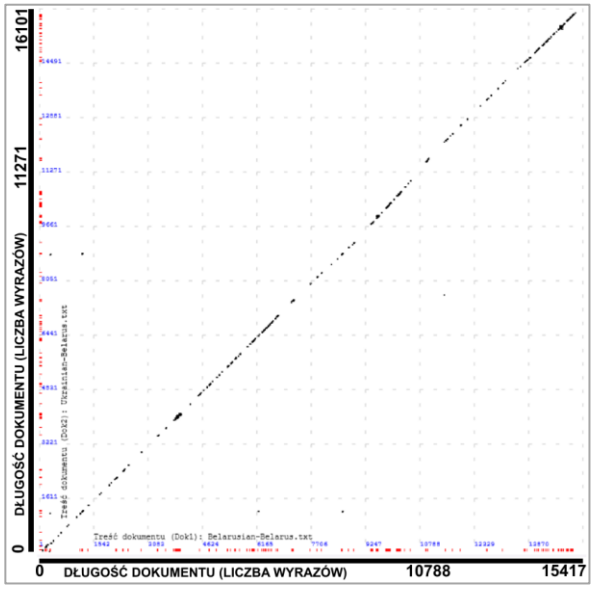
Rysunek 4.9. Parametry analizy – *bp*: 50%, *wv*: 5, *gw*: 8



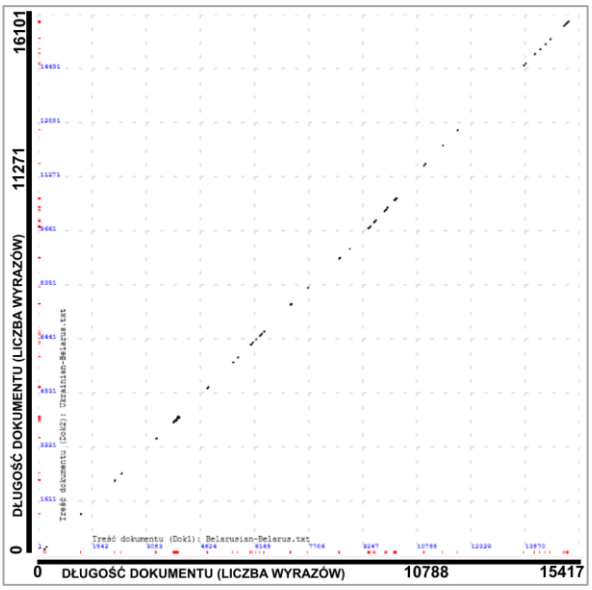
Rysunek 4.10. Parametry analizy – *bp*: 70%, *wv*: 5, *gw*: 8



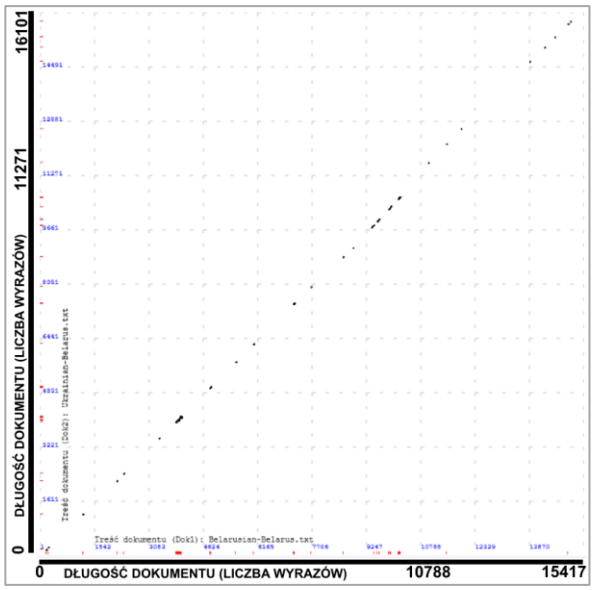
Rysunek 4.11. Parametry analizy – *bp*: 90%, *wv*: 5, *gw*: 8



Rysunek 4.12. Parametry analizy – bp: 100%, *wv*: 5, *gw*: 8



Rysunek 4.13. Parametry analizy – bp: 100%, *wv*: 8,*gw*: 8



Rysunek 4.14. Parametry analizy – bp: 100%, *wv*: 8,*gw*: 2

ID	Język dokumentu (1)	Język dokumentu (2)	*Bp*	*wv*	*gw*	*Liczba wyrazów (liczba znaków) (1) x (2)*	WYNIK (1)	WYNIK (2)
1	Białoruski	Ukraiński	42%	5	8	15417 X 16101 (117333 X 115464)	59,71%	56,49%
2	Białoruski	Ukraiński	45%	5	8	15417 X 16101 (117333 X 115464)	57,92%	54,51%
3	Białoruski	Ukraiński	50%	5	8	15417 X 16101 (117333 X 115464)	55,82%	52,62%
4	Białoruski	Ukraiński	70%	5	8	15417 X 16101 (117333 X 115464)	29,89%	26,87%
5	Białoruski	Ukraiński	90%	5	8	15417 X 16101 (117333 X 115464)	12,59%	9,9%
6	Białoruski	Ukraiński	100%	5	8	15417 X 16101 (117333 X 115464)	11,79%	9,03%
7	Białoruski	Ukraiński	100%	8	8	15417 X 16101 (117333 X 115464)	6,08%	4,51%
8	Białoruski	Ukraiński	100%	8	5	15417 X 16101 (117333 X 115464)	5,38%	3,94%
9	Białoruski	Ukraiński	100%	8	2	15417 X 16101 (117333 X 115464)	4,4%	3,11%
10	Białoruski	Ukraiński	100%	8	1	15417 X 16101 (117333 X 115464)	4,03%	2,73%

Tabela 4.5. Wyniki analiz porównania tekstów, z uwzględnieniem różnych parametrów

Przedstawione powyżej wyniki (tabela 4.5 oraz rysunki 4.9-4.14) należy, podobnie jak w poprzednich przypadkach, rozpatrywać pod kątem tezy, jaka została postawiona w rozdziale. Z powyższych wykresów widać, że teksty wykazują znaczne podobieństwo, a w niektórych przypadkach prawie identyczność – linia przebiegająca po ukosie macierzy. Na podstawie rysunków i tabeli widać, że najlepszy rezultat wykazujący zachodzące podobieństwo pomiędzy tekstami w językach białoruskim i ukraińskim uzyskuje się poprzez ustawienie podobieństwa wyrazów na poziomie wyższym jak 70% (*bw*) i minimalna liczba wyrazów w wektorze sekwencji (*wv*) oraz maksymalna dopuszczalna przerwa pomiędzy wyrazami (*gw*) pozostają na

niezmienionym poziomie i wynoszą odpowiednio 5 i 8 (rys. 4.10-4.12). Ustawienie wartości podobieństwa wyrazów (*bw*) poniżej tego progu sprawia, że graficzny rezultat porównania jest mniej czytelny, pojawia się szum, a linia ukośna, która odpowiada za wizualne potwierdzenie podobieństwa tekstów, jest mniej widoczna (rys. 4.9). Zwiększenie zmiennej *wv* i zmniejszenie *gw* obniża wynik podobieństwa, ale nie można tutaj mówić o zakłamaniu wyniku (rys. 4.13-4.14). Wynik jest bardziej dokładny, przerwy w występowaniu po sobie terminów tworzących wektor ciągłości, nie są uwzględniane lub są uwzględniane, ale w mniejszym stopniu. Przerwy należy rozumieć, jako umieszczone pomiędzy terminami inne wyrazy lub zmiana szyku wyrazów, wynikająca przede wszystkim z różnic pomiędzy językami. Dlatego też, ze względu na to, że są to różne języki, zmienne *wv* i *gw* w większości analiz były ustawione według podanych wartości i nie zmieniane.

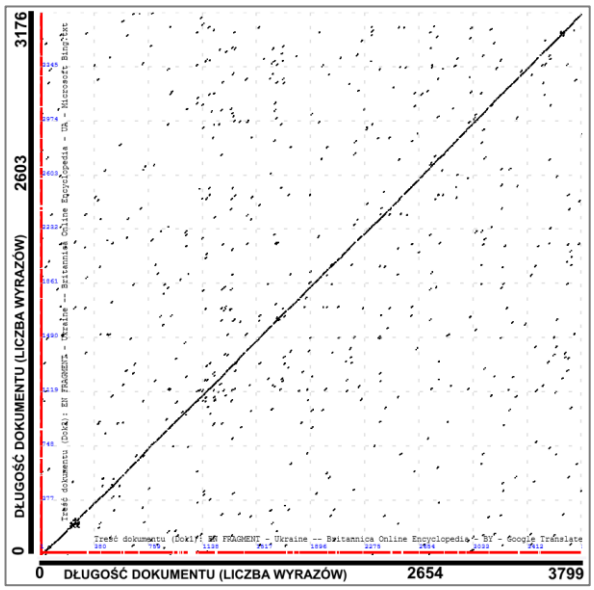
ID	Język białoruski	Język ukraiński
1	[...] 19 верасня 1991 гады краіна стала называцца Рэспубліка Беларусь, у гэты ж час былі прынятыя новыя герб і сцяг заменены на сучасныя герб і сцяг 7 чэрвеня 1995 [...]	[...] 19 вересня 1991 року країна стала називатися Республіка Білорусь, тоді ж були прийняті нові герб і прапор замінений на сучасні герб і прапор 7 червня 1995 [...]
2	[...] XVI ст. З 1620х гадоў назва замацавалася за ўсходнімі землямі Вялікага Княства Літоўскага — падзвінскімі і падняпроўскімі паветамі. На думку [...]	[...] XVI ст. З 1620х років назва закріпилася за східними землями Великого князівства Литовського — Подвинським і Наддніпрянським повітами. На думку [...]
3	[...] і мае тытул «Масква сталіца ўсёй Белай Русіі» Moscovia urbs metropolis tutius Russiæ Albæ. План горада павёрнуты на 90 градусаў поўнач — справа, зверху — захад Карта «Вялікае Княства Маскоўскае ці Царства Белай Русі паводле апошніх паведамленняў» Estats du Grandduc de Moscovie ou de l'Empereur de la Russie Blanche suivant les derniers relations, каля 1749 г. Картограф Гендрык дэ Лет Нідэрланды Карта [...]	[...] і має назву «Москва, столиця всієї Білої Русі» Moscovia urbs metropolis tutius Russiæ Albæ . План міста повернуто на 90 градусів справа – північ, зверху – захід Карта «Вялікае Княства Маскоўскае ці Царства Белай Русі паводле апошніх паведамленняў» Estats du Grandduc de Moscovie ou de l'Empereur de la Russie Blanche suivant les derniers relations, каля 1749 г. Картограф Гендрык дэ Лет Нідэрланды Карта [...]

Tabela 4.6. Fragmenty tekstu uznanego za podobne dla analizy ID 5 z tabeli 4.5

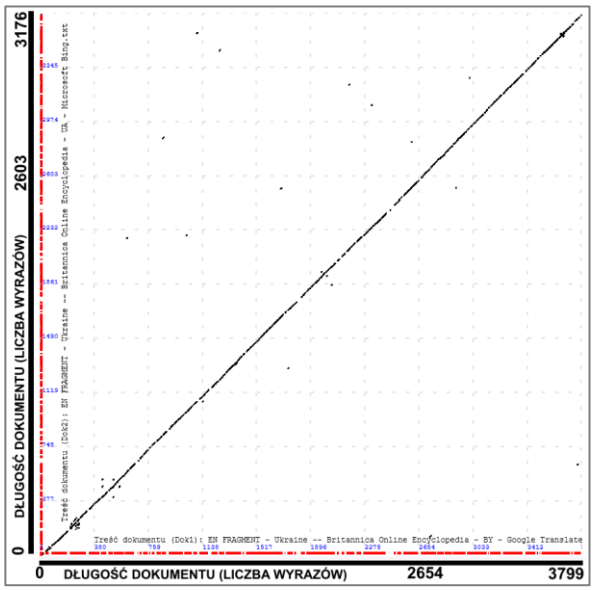
Tabela 4.6 zawiera wybrane fragmenty uznane za podobne będące wynikiem analizy porównania omawianych tekstów. Każdy z powyższych wyrazów uznany za podobny do swojego odpowiednika w drugim tekście ma swoją interpretację graficzną w postaci punktu na macierzy (rys. 4.11).

TEST 2. Artykuł encyklopedyczny o Ukrainie

W powyższej analizie zestawione zostały dwa fragmenty artykułów o Ukrainie[69] pochodzące z innej encyklopedii internetowej[70] (adres WWW strony encyklopedii dostępny jest w odnośniku[71]) niż w poprzednich analizach. Tekst w języku angielskim został przetłumaczony przez translatory dwóch różnych firm na języki białoruski[72] i ukraiński[73]. Parametry użyte w analizie porównania są analogiczne do tych z poprzednich testów. Poniżej znajduje się interpretacja graficzna przeprowadzonego porównania.

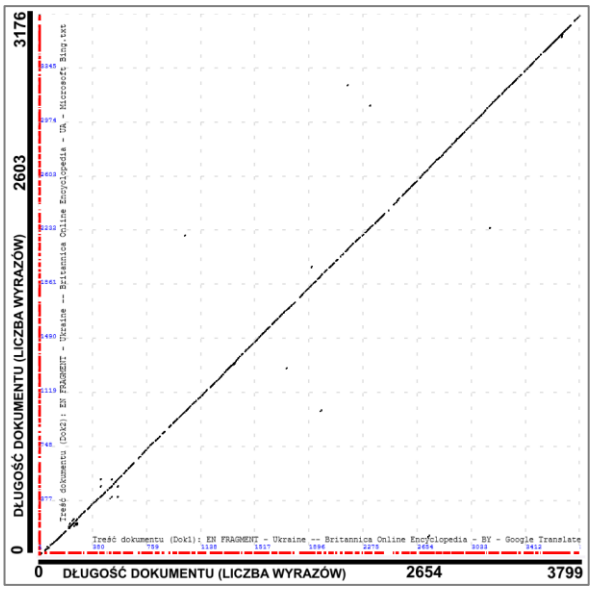


Rysunek 4.15. Parametry analizy – *bp*: 30%, *wv*: 5, *gw*: 8

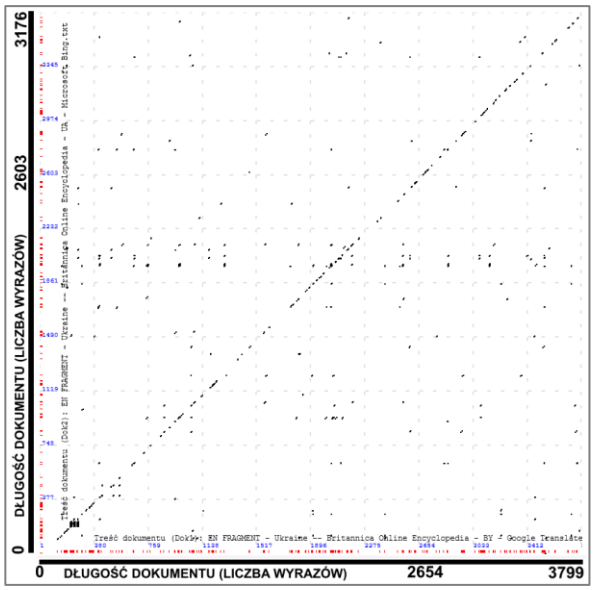


Rysunek 4.16. Parametry analizy – *bp*: 42%, *wv*: 5, *gw*: 8

[69] https://antyplagius.n-dms.com/tests/Belarusian-Ukrainian/EN%20FRAGMENT%20-%20Ukraine%20--%20Britannica%20Online%20Encyclopedia.txt
[70] https://antyplagius.n-dms.com/tests/Belarusian-Ukrainian/Ukraine%20--%20Britannica%20Online%20Encyclopedia.pdf
[71] https://www.britannica.com/place/Ukraine
[72] https://translate.google.pl
[73] https://www.bing.com/translator



Rysunek 4.17. Parametry analizy – *bp*:50%, *wv*:5, *gw*: 8



Rysunek 4.18. Parametry analizy–bp: 100%, *wv*:3,*gw*: 8

Podobnie jak w poprzednim teście, najlepiej dopasowane parametry dla analizy tych dwóch języków, to podobieństwo wyrazów *bp* powyżej 50%, ale nie więcej jak 70% - zmniejsza się wtedy szum i kształtuje wyraźniej pełniejsza linia prosta.

ID	Język dokumentu (1)	Język dokumentu (2)	*bp*	*wv*	*gw*	*Liczba wyrazów (liczba znaków) (1) x (2)*	WYNIK (1)	WYNIK (2)
1	Białoruski	Ukraiński	42%	5	8	3799 X 3716 (28616 X 28633)	55,41%	56,38%
2	Białoruski	Ukraiński	45%	5	8	3799 X 3716 (28616 X 28633)	52,88%	53,90%
3	Białoruski	Ukraiński	50%	5	8	3799 X 3716 (28616 X 28633)	51,22%	52,10%
4	Białoruski	Ukraiński	70%	5	8	3799 X 3716 (28616 X 28633)	21,32%	21,61%
5	Białoruski	Ukraiński	90%	5	8	3799 X 3716 (28616 X 28633)	3,05%	3,12%
6	Białoruski	Ukraiński	100%	5	8	3799 X 3716 (28616 X 28633)	2,45%	2,50%
7	Białoruski	Ukraiński	100%	8	8	3799 X 3716 (28616 X 28633)	0,47%	0,48%
8	Białoruski	Ukraiński	100%	8	5	3799 X 3716 (28616 X 28633)	0%	0%
9	Białoruski	Ukraiński	100%	8	2	3799 X 3716 (28616 X 28633)	0%	0%
10	Białoruski	Ukraiński	100%	8	1	3799 X 3716 (28616 X 28633)	0%	0%

Tabela 4.7. Wyniki analiz porównania tekstów, z uwzględnieniem różnych parametrów analizy

Podniesienie wymogu podobieństwa wyrazów bp do 100%, czyli wymuszenie konieczności ich identyczności sprawia, że wyniki analizy wynoszą 0% podobieństwa, a teksty uznane są za niepodobne do siebie.

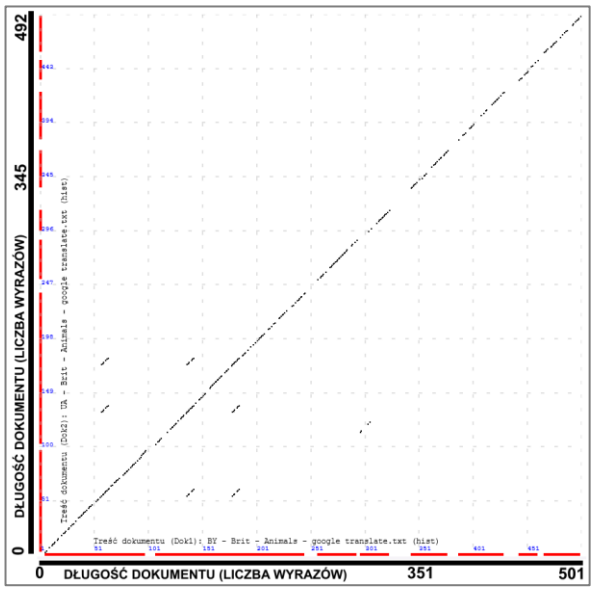
ID	Język białoruski	Język ukraiński
1	[...] Украіна, краіна, размешчаная ва ўсходняй Еўропе, другая па велічыні на кантыненце пасля Расіі. Сталіца — Кіеў, размешчаны на рацэ Днепр у паўночнацэнтральнай [...]	[...] Україна, країна розташована на сході Європи, друга за величиною на континенті після Росії. Столицею є Київ, розташований на річці Дніпро в північноцентральній [...]
2	[...] Саюза як Украінская Савецкая Сацыялістычная Рэспубліка С. С. Р. . Калі Савецкі Саюз пачаў распадацца ў 1990–91 гадах, заканадаўчая ўлада Украінскай ССР. абвясцілі суверэнітэт 16 ліпеня 1990 г. , а затым поўную незалежнасць 24 жніўня 1991 г. , крок, які быў пацверджаны народным адабрэннем на плебісцыце 1 снежня 1991 г. . Пасля распаду СССР у снежні 1991 года Украіна атрымала поўную незалежнасць. Краіна змяніла сваю афіцыйную назву на Украіна, і гэта дапамагло заснаваць Садружнасць Незалежных Дзяржаў СНД, абяднанне краін, якія раней былі рэспублікамі Савецкага Саюза. Клімат Украіна знаходзіцца ва ўмераным кліматычным поясе, на які паступае ўмерана цёплае вільготнае паветра з Атлантычнага акіяна. Зімы на захадзе значна мякчэй, чым на ўсходзе. [...]	[...] Союзу як Українська Радянська Соціалістична Республіка СРСР. Коли Радянський Союз почав розпадатися в 199091 роках, законодавчий орган УРСР проголосив суверенітет 16 липня 1990, а потім повну незалежність 24 серпня 1991, крок, який був підтверджений народним схваленням на плебісциті 1 грудня 1991. З розпадом СРСР у грудні 1991 року Україна отримала повну незалежність. Країна змінила свою офіційну назву на Україна, і це допомогло заснувати Співдружність Незалежних Держав СНД, обєднання країн, які раніше були республіками Радянського Союзу. Клімат Україна лежить у помірному кліматичному поясі, на який впливає помірно тепле, вологе повітря з Атлантичного океану. Зими на заході значно мякші, ніж на сході. [...]
3	[...] сельскагаспадарчага рэгіёна займаюць ворныя землі лясы займаюць толькі каля 1/8 тэрыторыі. Далей на поўдзень, каля Чорнага, Азоўскага мораў і Крымскіх гор, лесастэп злучаецца са стэпавай зонай, плошча якой складае каля 89 000 квадратных міль 231 000 квадратных кіламетраў. Многія з плоскіх бязлесных раўнін у гэтым рэгіёне апрацоўваюцца, хаця малая гадавая колькасць ападкаў і гарачае лета робяць неабходным дадатковае абрашэнне. [...]	[...] сільськогосподарського регіону займають орні землі Ліси займають лише близько восьмої частини площі. Далі на південь, біля Чорного моря, Азовського моря і Кримських гір, лісостеп приєднується до степової зони, площа якої становить близько 89 000 квадратних миль 231 000 квадратних км. Багато плоских, безлісих рівнин в цьому регіоні обробляються, хоча низька річна кількість опадів і спекотне літо роблять необхідним додаткове зрошення. [...]
4	[...] Запарожжа. Слаба індустрыялізаваныя гарады на захадзе, такія як Ужгарад і Хмяльніцкі, сутыкаюцца з забруджваннем паветра, выкліканым перавагай неэфектыўных аўтамабіляў. Асноўныя рэкі, у тым ліку Днепр, Днестр, Інгул і Данец, сурёзна забруджаныя хімічнымі ўгнаеннямі і пестыцыдамі [...]	[...] Запоріжжя. Слабо індустріалізовані міста на заході, такі як Ужгород та Хмельницький, стикаються із забрудненням повітря, спричиненим переважанням неефективних автомобілів. Великі річки, включаючи Дніпро, Дністер, Інгул і Донець, серйозно забруднені хімічними добривами і пестицидами [...]

Tabela 4.8. Fragmenty tekstu uznanego za podobne dla analizy ID 3 z tabeli 4.7

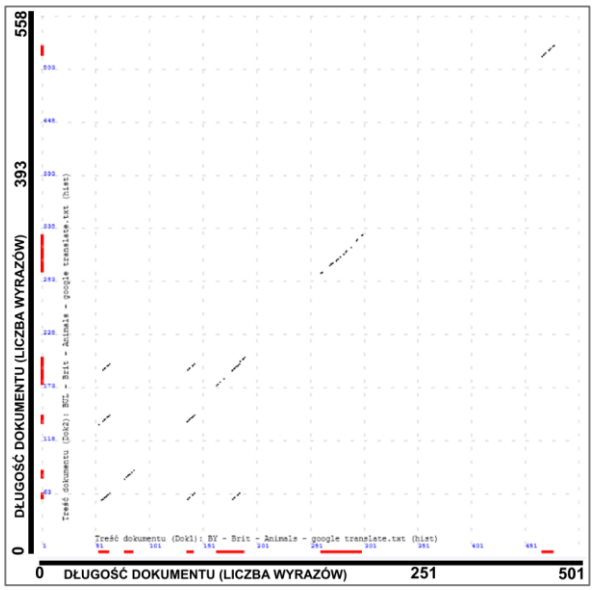
Powyżej zostały przedstawione (tab. 4.8) wybrane fragmenty tekstów uznane za podobne. Dane są rezultatem analizy na podstawie parametrów ID 3 z tabeli 4.7. Widać wyraźne podobieństwo pomiędzy tekstami, w tym nawet pomiędzy bardzo długimi wyrazami. Teksty składają się zarówno z wyrazów podobnych (np. tabela 4.8. ID 4: індустрыялізаваныя vs. індустріалізовані), jak również całkowicie różnych (np. tabela 4.8. ID 4: гарады vs. міста) - charakterystycznych dla danego języka. Odpowiednie ustawienie opisanych parametrów analizy sprawia, że wyrazy całkowicie różne są ignorowane w celu osiągnięcia ciągłości wektora podobnych wyrazów.

TEST 3. Artykuł encyklopedyczny o Ukrainie w innych językach pisanych cyrylicą

W celu uzupełnienia informacji, poniżej przedstawione zostały dodatkowo analizy porównawcze fragmentu[74] wyciętego z powyższych tekstów napisanych w językach białoruskim i ukraińskim[75] z innymi językami posługującymi się cyrylicą[76]. Poniższe języki pochodzą zarówno z państw europejskich, jak również azjatyckich. Parametry analizy są podobne i wynoszą odpowiednio – bp: 50%, wv: 5, gw: 4-8.



Rysunek 4.19. Języki: białoruski - ukraiński

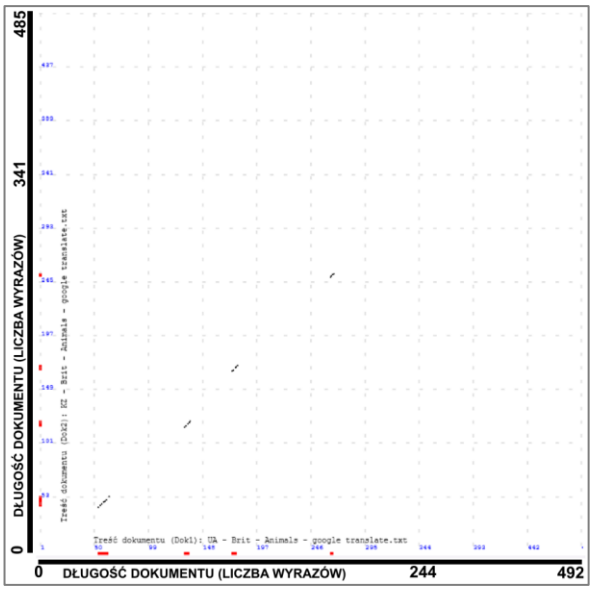


Rysunek 4.20. Języki: białoruski - bułgarski

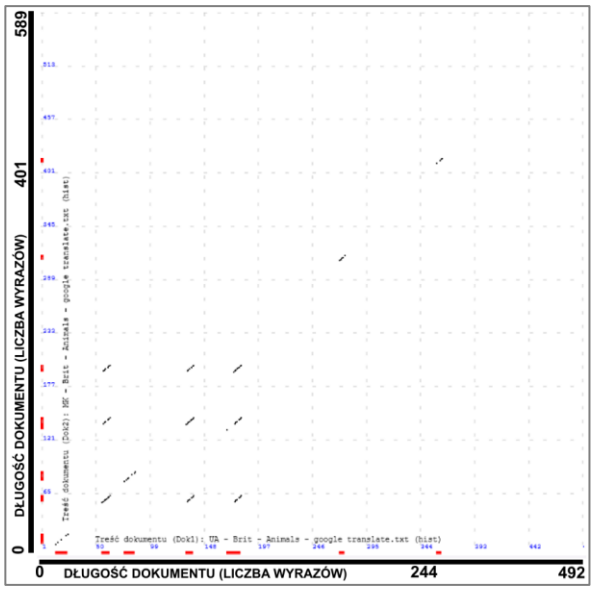
[74] Rozdział „Plant and animal life” w języku angielskim z encyklopedii https://antyplagius.n-dms.com/tests/Belarusian-Ukrainian/CYR/ENG%20-%20Brit%20-%20Animals%20-%20google%20translate.txt

[75] Przetłumaczone za pomocą narzędzia https://translate.google.com

[76] Cyrylica to alfabet używany przez wiele języków, głównie na terenie Europy Wschodniej i Azji Środkowej. Oto niektóre z języków, które używają cyrylicy: Rosyjski, Ukraiński, Białoruski, Bułgarski, Serbski, Macedoński, Czarnogórski, Kazachski, Kirgiski, Tadżycki, Uzbecki (czasami zamiennie z łacińskim), Osetyjski, Abchaski, Baszkirski, Czuwaski, Komi, Mordwiński, Tatarski, Tuwiński, Jakucki.



Rysunek 4.21. Języki: ukraiński - kazachski



Rysunek 4.22. Języki: ukraiński - macedoński

ID	Język dokumentu (1)	Język dokumentu (2)	*bp*	*wv*	*gw*	*Liczba wyrazów (liczba znaków) (1) x (2)*	WYNIK (1)	WYNIK (2)
1	Białoruski[77]	Ukraiński[78]	50%	5	8	501 X 492 (3687 X 3611)	57,49%	57,93%
2	Białoruski	Bułgarski[79]	50%	5	8	501 X 558 (3687 X 3715)	13,37%	12,01%
3	Białoruski	Serbski[80]	50%	5	8	501 X 512 (3687 X 3444)	10,78%	10,55%
4	Białoruski	Macedoński[81]	50%	5	4	501 X 569 (3687 X 3815)	6,39%	5,62%
5	Białoruski	Kazachski[82]	50%	5	4	501 X 485 (3687 X 3729)	4,19%	4,33%
6	Ukraiński	Bułgarski	50%	5	8	492 X 558 (3611 X 3715)	17,48%	15,77%
7	Ukraiński	Serbski	50%	5	8	492 X 512 (3611 X 3444)	14,43%	14,06%
8	Ukraiński	Macedoński	50%	5	4	492 X 569 (3611 X 3815)	9,35%	8,08%
9	Ukraiński	Kazachski	50%	5	4	492 X 485 (3611 X 3729)	4,33%	4,27%

Tabela 4.9. Tabela przedstawiająca wyniki kilku analiz porównania tekstów, z tymi samymi parametrami analizy, pomiędzy wybranymi językami posługującymi się cyrylicą

[77] Tekst dostępny pod adresem: http://antyplagius.n-dms.com/tests/Belarusian-Ukrainian/CYR/BY%20-%20Brit%20-%20Animals%20-%20google%20translate.txt

[78] Tekst dostępny pod adresem: http://antyplagius.n-dms.com/tests/Belarusian-Ukrainian/CYR/UA%20-%20Brit%20-%20Animals%20-%20google%20translate.txt

[79] Tekst dostępny pod adresem: http://antyplagius.n-dms.com/tests/Belarusian-Ukrainian/CYR/BUL%20-%20Brit%20-%20Animals%20-%20google%20translate.txt

[80] Tekst dostępny pod adresem: http://antyplagius.n-dms.com/tests/Belarusian-Ukrainian/CYR/SR%20-%20Brit%20-%20Animals%20-%20google%20translate.txt

[81] Tekst dostępny pod adresem: http://antyplagius.n-dms.com/tests/Belarusian-Ukrainian/CYR/MK%20-%20Brit%20-%20Animals%20-%20google%20translate.txt

[82] Tekst dostępny pod adresem: http://antyplagius.n-dms.com/tests/Belarusian-Ukrainian/CYR/KZ%20-%20Brit%20-%20Animals%20-%20google%20translate.txt

Poniżej znajduje się tabela zawierająca wynik pełnej analizy podobieństwa ID 5 z tabeli 4.9.

ID	Język białoruski (1)	Język kazachski (2)	Fragment tekstu (1)	Fragment tekstu (2)
1	[...] — каля 44 000 квадратных міль 114 000 квадратных кіламетраў — [...]	[...] — шамамен 44 000 шаршы миль 114 000 шаршы км — [...]	55-65	45-55
2	[...] 78 000 квадратных міль 202 000 [...]	[...] 78 000 шаршы мильді 202 000 [...]	136-141	117-122
3	[...] 89 000 квадратных міль 231 000 [...]	[...] 89 000 шаршы мильді 231 000 [...]	178-183	167-172
4	[...] 6 міль 10 км [...]	[...] 6 миль 10 км [...]	269-272	251-254

Tabela 4.10. Wynik analizy porównawczej tekstu napisanego w językach białoruskim i kazachskim

Jak widać z powyższych analiz, te same teksty napisane w językach używających cyrylicę, nie gwarantują automatycznie pomiędzy sobą wysokiego stopnia podobieństwa. Dużą rolę odgrywa tutaj przynależność do danej grupy językowej i wspólna historia ewolucji języka. W tabeli 4.10. widać jak małe podobieństwo wykazują niektóre teksty, a głównym elementem podwyższającym wartości podobieństwa są liczby.

4.4. Analiza dokumentów tekstowych napisanych w językach: duńskim i norweskim[83]

Język duński i norweski to dwa języki germańskie używane w Skandynawii. Duński jest oficjalnym językiem w Danii, natomiast norweski jest oficjalnym językiem w Norwegii. Obie te języki mają wspólne korzenie w językach staronordyjskich używanych przez Wikingów we wczesnym średniowieczu[84].

Wspólne cechy języków duńskiego i norweskiego obejmują:

- Podobne systemy gramatyczne: obie mają zbliżone struktury gramatyczne, takie jak deklinacje rzeczowników, odmiany czasowników, przymiotniki, itp.
- Słownictwo: choć występują różnice w pisowni i wymowie, wiele słów jest wspólnych lub bardzo podobnych w obu językach, np. "god morgen" (duński) / "god morgen" (norweski) - dzień dobry, "tak" (duński) / "takk" (norweski) - dziękuję, "hus" (duński) / "hus" (norweski) - dom.

[83] Więcej przykładów analizy tych języków znajduje się na stronie oficjalnego kanału programu: https://www.youtube.com/watch?v=OhQOvuBAy98&list=PLPFeTDhxdQPawnjGhPytgFJeJb-YOXmHC&index=6

[84] Więcej informacji na stronie encyklopedii: https://pl.wikipedia.org/wiki/J%C4%99zyk_staronordyjski

- Skandynawski system liczbowy: duński i norweski mają podobny system liczbowy, który jest oparty na liczeniu "na dziesiątki" (np. siedemdziesiąt dwie - "tooghalvfjerds" w języku duńskim i "syttito" w języku norweskim).
- Bliska wzajemna zrozumiałość: mówiący językiem duńskim i norweskim są często w stanie wzajemnie się zrozumieć, chociaż mogą występować pewne różnice w wymowie, akcencie i niektórych słowach. Dlatego też osoby, które znają jeden z tych języków, mogą łatwiej nauczyć się drugiego.
- Alfabet: obie używają alfabetu łacińskiego z dodatkiem specjalnych liter, takich jak "æ", "ø" i "å".

Mimo że język duński i norweski mają wiele wspólnych cech, istnieją też różnice, szczególnie w dialektach i wymowie. Niemniej jednak ich bliskie pokrewieństwo historyczne i kulturowe sprawia, że są uważane za języki blisko spokrewnione.

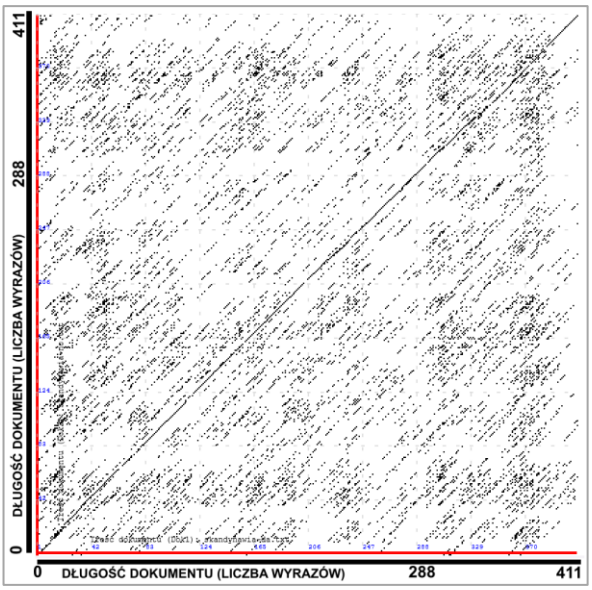
TEST. Artykuł o Skandynawii

Artykuł przedstawiający Skandynawię, jako modelowy region w dziedzinie zrównoważonego rozwoju, innowacyjności i wysokiej jakości życia. Tekst w języku polskim (rozdz. 7.7.1)[85] przetłumaczony został na dwa języki: duński (rozdz. 7.7.2)[86] oraz norweski (rozdz. 7.7.3)[87] za pomocą zaawansowanego modelu językowego opracowanego przez firmę OpenAI – ChatGPT w wersji 4. Teksty zostały między sobą zestawione pod kątem podobieństwa, a graficzna interpretacja porównanych tekstów znajduje się poniżej na rys. 4.23-4.26. Stała *gw* będzie taka sama dla wszystkich testów w podrozdziale, ponieważ specyfika problemu nie wymusza jej ciągłego dostosowywania do tekstu – teksty pochodzą z tej samej grupy językowej, a języki są wysoce do siebie podobne.

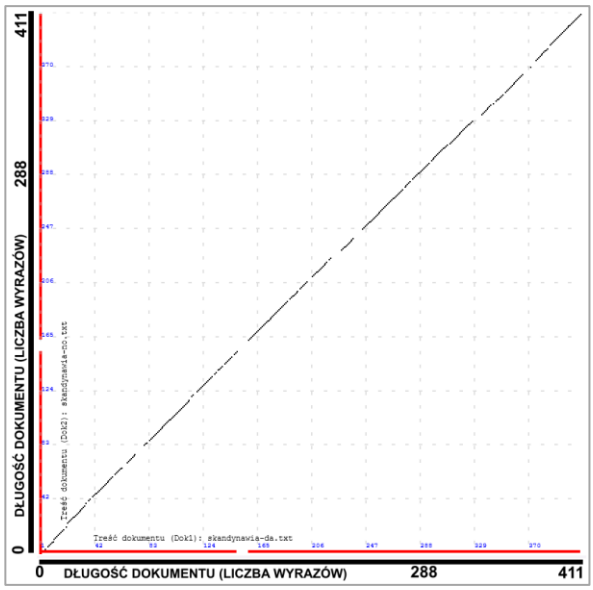
[85] Tekst dostępny pod adresem: https://antyplagius.n-dms.com/tests/Danish-Norwegian/skandynawia-pl.txt

[86] Tekst dostępny pod adresem: https://antyplagius.n-dms.com/tests/Danish-Norwegian/skandynawia-da.txt

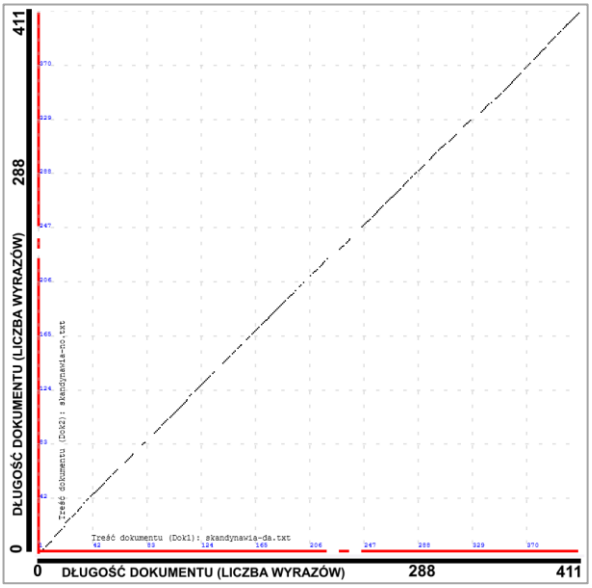
[87] Tekst dostępny pod adresem: https://antyplagius.n-dms.com/tests/Danish-Norwegian/skandynawia-no.txt



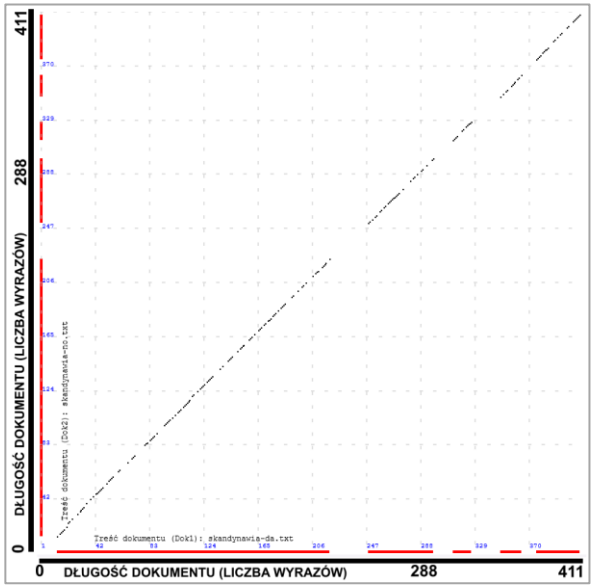
Rysunek 4.23. Parametry analizy – *bp*: 20%, *wv*: 7, *gw*: 8



Rysunek 4.24. Parametry analizy – *bp*: 50%, *wv*: 7, *gw*: 8



Rysunek 4.25. Parametry analizy – *bp*: 70%, *wv*:7, *gw*: 8



Rysunek 4.26. Parametry analizy – bp: 100%, *wv*: 7,*gw*: 8

ID	Język dokumentu (1)	Język dokumentu (2)	*Bp*	*wv*	*gw*	*Liczba wyrazów (liczba znaków) (1) x (2)*	WYNIK (1)	WYNIK (2)
1	Duński	Norweski	20%	7	8	411 X 411 (2611 X 2614)	100%	99,76%
2	Duński	Norweski	50%	7	8	411 X 411 (2611 X 2614)	82,68%	82,48%
3	Duński	Norweski	70%	7	8	411 X 411 (2611 X 2614)	72,68%	72,51%
4	Duński	Norweski	100%	7	8	411 X 411 (2611 X 2614)	45,85%	45,74%

Tabela 4.11. Wynik analizy porównawczej tekstu napisanego w językach duńskim i norweskim

ID	Język duński	Język norweski
1	[...] Skandinavien Innovation og bæredygtig udvikling Introduktion Skandinavien, en region i Nordeuropa bestående af tre lande Danmark, Norge og Sverige, ses ofte som et forbillede for bæredygtig udvikling, innovation og høj livskvalitet. Finland og Island, selvom de nogle gange regnes for skandinaviske lande, tilhører geografisk og kulturelt den nordiske region. [...]	[...] Skandinavia Innovasjon og bærekraftig utvikling Innledning Skandinavia, en region i NordEuropa som består av tre land Danmark, Norge og Sverige, er ofte sett på som et forbilde for bærekraftig utvikling, innovasjon og høy livskvalitet. Finland og Island, selv om de noen ganger regnes som skandinaviske land, tilhører geografisk og kulturelt den nordiske regionen. [...]
2	[...] Uddannelsessystemerne i regionen lægger stor vægt på kreativitet, selvstændig tænkning og læring gennem handling. Universiteter såsom Københavns Universitet, Universitetet i Oslo og Stockholms Universitet er højt værdsatte for deres forskning og teknologiske udvikling. [...]	[...] Utdanningssystemene i regionen legger stor vekt på kreativitet, selvstendig tenkning og læring gjennom handling. Universiteter som Københavns Universitet, Universitetet i Oslo og Stockholms Universitet er høyt verdsatt for sin forskning og teknologiske utvikling. [...]

Tabela 4.12. Wybrane fragmenty tekstów uznane za podobne w ramach analizy ID 3 z tabeli 4.11

Wyniki zawarte w tabelach 4.11-4.12 oraz w postaci rysunków 4.24-4.26 udowadniają wysokie podobieństwo pomiędzy językami pomimo różnych wartości dla parametry *bp* (zwłaszcza dla wartości 100%, czyli identyczności wyrazów). Ustawienie wartości podobieństwa równej 20% sprawia, że graficzny rezultat porównania jest mniej czytelny, pojawia się szum i poprzez to linia ukośna, która odpowiada za wizualne potwierdzenie podobieństwa tekstów, staje się niewidoczna (rys. 4.23). Podobnie jak w przykładach w poprzednich testach, również w tym przypadku podwyższenie wyniku podobieństwa pomiędzy dokumentami spowodowane zmniejszeniem wartości parametru *bp*, nie jest efektem oczekiwanym, ale mylącym. Analogicznie do poprzednich badanych przypadków, przerwy na rysunku należy rozumieć, jako umieszczone pomiędzy terminami inne wyrazy lub zmianę szyku wyrazów.

4.5. Analiza dokumentów tekstowych napisanych w językach: niderlandzkim i niemieckim[88]

Język niderlandzki i niemiecki to dwa zachodniogermańskie języki używane w Europie. Niderlandzki jest oficjalnym językiem w Holandii, Belgii (Region Flamandzki) oraz Surinamie, natomiast niemiecki jest oficjalnym językiem w Niemczech, Austrii oraz jednym z języków urzędowych w Szwajcarii, Belgii (Region Wspólnoty Niemieckiej) i Liechtensteinie.

[88] Więcej przykładów analizy tych języków znajduje się na stronie oficjalnego kanału programu: https://www.youtube.com/watch?v=IfeQu2mIjIs&list=PLPFeTDhxdQPawnjGhPytgFJeJb-YOXmHC&index=10

Wspólne cechy języków niderlandzkiego i niemieckiego obejmują:

- Gramatyka: obie mają zbliżone systemy gramatyczne, takie jak deklinacje rzeczowników, odmiany czasowników, przymiotniki, itp. W obu językach występuje system przypadków, choć w języku niderlandzkim system ten jest mniej skomplikowany niż w niemieckim.
- Słownictwo: wiele słów jest wspólnych lub bardzo podobnych w obu językach, choć występują różnice w pisowni i wymowie. Przykłady: "goedemorgen" (niderlandzki) / "guten Morgen" (niemiecki) - dzień dobry, "dank je" (niderlandzki) / "danke" (niemiecki) - dziękuję, "huis" (niderlandzki) / "Haus" (niemiecki) - dom.
- Alfabet: obie używają alfabetu łacińskiego z kilkoma dodatkowymi literami, takimi jak "ä", "ö" i "ü" w języku niemieckim oraz "ij" w języku niderlandzkim.

Mimo że język niderlandzki i niemiecki mają wiele wspólnych cech, istnieją też różnice, szczególnie w dialektach, wymowie oraz pewnych aspektach gramatycznych. Niemniej jednak, osoby, które znają jeden z tych języków, często mają łatwiejsze zadanie w nauce drugiego ze względu na liczne podobieństwa.

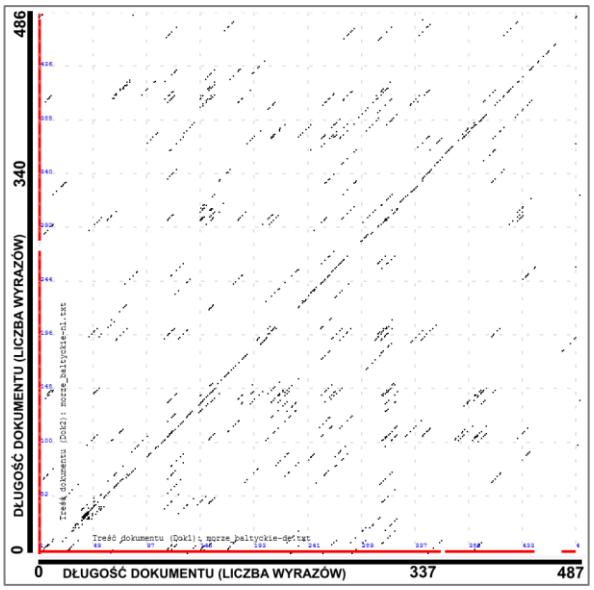
TEST. Artykuł o Morzu Bałtyckim

Artykuł opisuje Morze Bałtyckie, jedne z najmłodszych i najbardziej dynamicznie zmieniających się mórz, charakteryzujące się unikalnymi właściwościami zarówno pod względem geograficznym, jak i ekologicznym. Tekst w języku polskim (rozdz. 7.8.1)[89] przetłumaczony został na dwa języki: niemiecki (rozdz. 7.8.2)[90] oraz niderlandzki (rozdz. 7.8.3)[91] za pomocą zaawansowanego modelu językowego opracowanego przez firmę OpenAI – ChatGPT w wersji 4. Teksty zostały między sobą porównane, graficzna interpretacja porównanych tekstów znajduje się poniżej, a tabele zawierają dodatkowe informacje o podobieństwie.

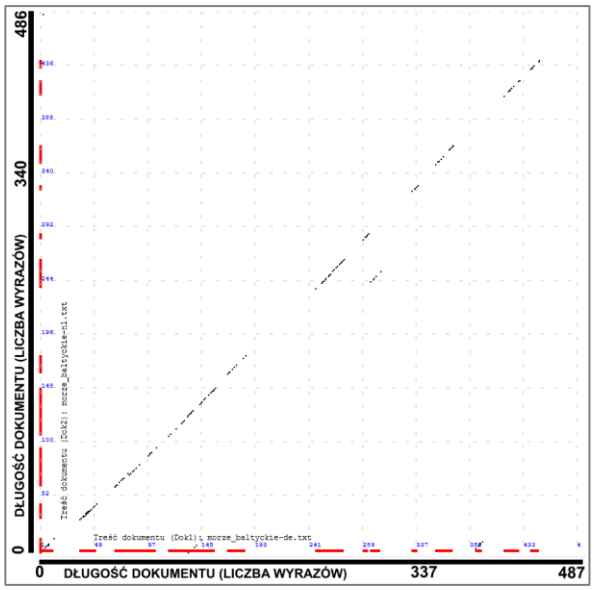
[89] Tekst dostępny pod adresem: https://antyplagius.n-dms.com/tests/German-Dutch/morze_baltyckie-pl.txt

[90] Tekst dostępny pod adresem: https://antyplagius.n-dms.com/tests/German-Dutch/morze_baltyckie-de.txt

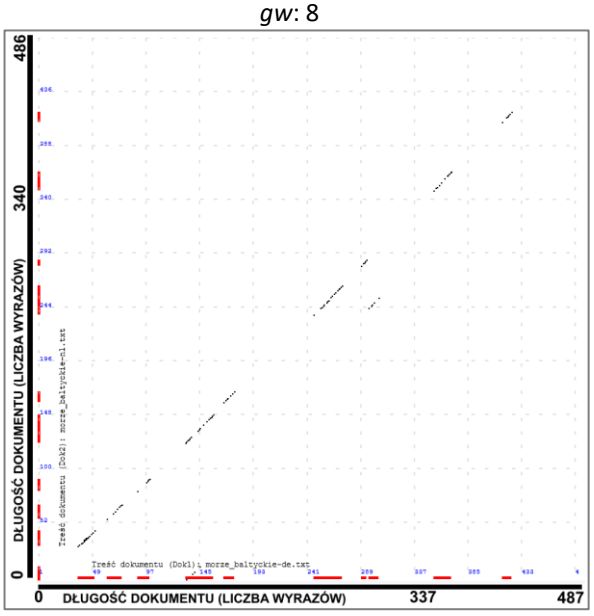
[91] Tekst dostępny pod adresem: https://antyplagius.n-dms.com/tests/German-Dutch/morze_baltyckie-nl.txt



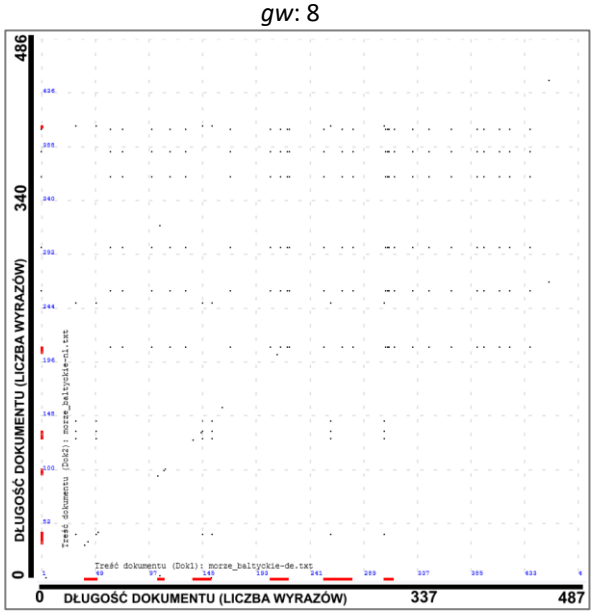
Rysunek 4.27. Parametry analizy – *bp*: 30%, *wv*: 6, *gw*: 8



Rysunek 4.28. Parametry analizy – *bp*: 40%, *wv*: 6, *gw*: 8



Rysunek 4.29. Parametry analizy – *bp*: 50%, *wv*: 6, *gw*: 8



Rysunek 4.30. Parametry analizy – bp: 100%, *wv*: 2,*gw*: 8

ID	Język dokumentu (1)	Język dokumentu (2)	*Bp*	*wv*	*gw*	*Liczba wyrazów (liczba znaków) (1) x (2)*	WYNIK (1)	WYNIK (2)
1	Niemiecki	Niderlandzki	30%	6	8	487 X 486 (3329 X 3009)	81,65%	81,82%
2	Niemiecki	Niderlandzki	40%	6	8	487 X 486 (3329 X 3009)	28,04%	26,45%
3	Niemiecki	Niderlandzki	50%	6	8	487 X 486 (3329 X 3009)	19,38%	19,21%
4	Niemiecki	Niderlandzki	100%	2	8	487 X 486 (3329 X 3009)	9,48%	5,58%

Tabela 4.13. Wyniki badania tekstów napisanych w językach niemieckim i niderlandzkim

Najbardziej nadające się do uznania za prawidłowe wyniki, to te zawarte w ramach rysunków 4.28 i 4.29. Im wyższy stopień podobieństwa *bp* wyrazów, tym linia ukośna staje się mnie widoczna i podobieństwo zanika. Dla testu ID 4, gdzie podobieństwo wyrazów zostało ustawione na 100%, nie pomogło obniżenie wymaganej liczby terminów *wv* celu zbudowania wektora ciągłości wyrazów. Oznacza to, że języki pod względem pisowni są od siebie znacząco różne. Jeżeli podobieństwo jest mniejsze od 50%, wtedy pojawia się szum i wykres staje się mniej czytelny, a wynik zakłamany (pomimo wyższego stopnia podobieństwa jak w tabeli 4.14 test ID 1).

ID	Język niemiecki	Język niderlandzki
1	[...] Die Ostsee, auch bekannt als Baltisches Meer, ist eines der jüngsten und dynamischsten Meere [...]	[...] De Oostzee, ook bekend als de Baltische Zee, is een van de jongste en meest [...]
2	[...] neun Ländern umgeben Polen, Litauen, Lettland, Estland, Russland, Finnland, Schweden, Dänemark und Deutschland. In diesem Essay [...]	[...] negen landen Polen, Litouwen, Letland, Estland, Rusland, Finland, Zweden, Denemarken en Duitsland. In dit essay [...]
3	[...] steht, erörtern. Geographie und natürliche Umwelt Die Ostsee ist ein relativ flaches Meer mit einer durchschnittlichen Tiefe von nur Metern, was sie besonders anfällig für Verschmutzung und Umweltveränderungen macht. Ihr tiefster Punkt, die LandsortTief, erreicht gerade einmal Meter. [...]	[...] staat, bespreken. Geografie en natuurlijke omgeving De Oostzee is relatief ondiep, met een gemiddelde diepte van slechts meter, waardoor het bijzonder kwetsbaar is voor vervuiling en milieuveranderingen. Het diepste punt, de Landsort Diepte, bereikt slechts meter. [...]

Tabela 4.14. Wybrane fragmenty tekstów uznane za podobne w ramach analizy ID 2 z tabeli 4.13

Powyższa tabela (4.14) zawiera wybrane fragmenty uznane za podobne będące wynikiem analizy ID 2 (tab. 4.13) porównania omawianych tekstów. Każdy z powyższych wyrazów uznany za podobny do swojego odpowiednika w drugim tekście ma swoją interpretację graficzną w postaci punktu na macierzy (rys. 4.28). Jak widać na powyższych przykładach, języki są na tyle odmienne w aspekcie pisowni, że obniżenie progu podobieństwa do ok. 40% pomiędzy terminami daje wyraźny obraz podobieństwa bez pojawienia się szumu na macierzy (czyli zbędnych terminów).

4.6. Analiza dokumentów tekstowych napisanych w językach: włoskim i francuskim[92]

Język włoski i francuski to dwa języki romańskie używane w Europie. Włoski jest oficjalnym językiem we Włoszech, San Marino, Watykanie oraz jednym z języków urzędowych w

[92] Więcej przykładów analizy tych języków znajduje się na stronie oficjalnego kanału programu: https://www.youtube.com/watch?v=t-q_QSvSh74&list=PLPFeTDhxdQPawnjGhPytgFJeJb-YOXmHC&index=11

Szwajcarii, natomiast francuski jest oficjalnym językiem we Francji, Belgii, Szwajcarii, Luksemburgu, Monako oraz wielu krajach afrykańskich.

Wspólne cechy języków włoskiego i francuskiego obejmują:

- Gramatyka: obie mają zbliżone systemy gramatyczne, takie jak deklinacje rzeczowników, odmiany czasowników, przymiotniki, itp. Obie używają systemu rodzajów gramatycznych (męski i żeński) oraz liczby pojedynczej i mnogiej.
- Słownictwo: wiele słów jest wspólnych lub bardzo podobnych w obu językach, choć występują różnice w pisowni i wymowie. Przykłady: "buongiorno" (włoski) / "bonjour" (francuski) - dzień dobry, "grazie" (włoski) / "merci" (francuski) - dziękuję, "casa" (włoski) / "maison" (francuski) - dom.
- Alfabet: obie używają alfabetu łacińskiego.
- Wspólne korzenie języków romańskich: Zarówno włoski, jak i francuski wywodzą się z łaciny, języka starożytnego Rzymu, co sprawia, że mają wiele wspólnych cech i podobieństw.
- Akcent: w obu językach akcent jest zazwyczaj stały i pada na tę samą sylabę w przypadku słów o podobnym brzmieniu.

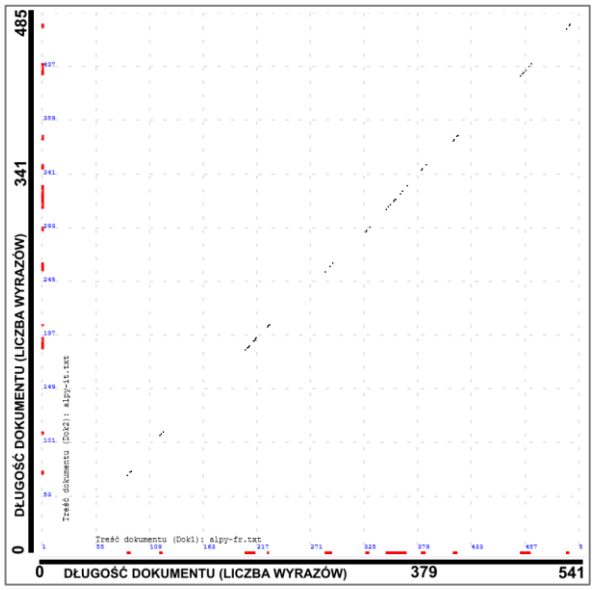
Mimo że język włoski i francuski mają wiele wspólnych cech, istnieją też różnice, szczególnie w dialektach, wymowie oraz niektórych aspektach gramatycznych i słownictwa.

TEST. Artykuł o Alpach

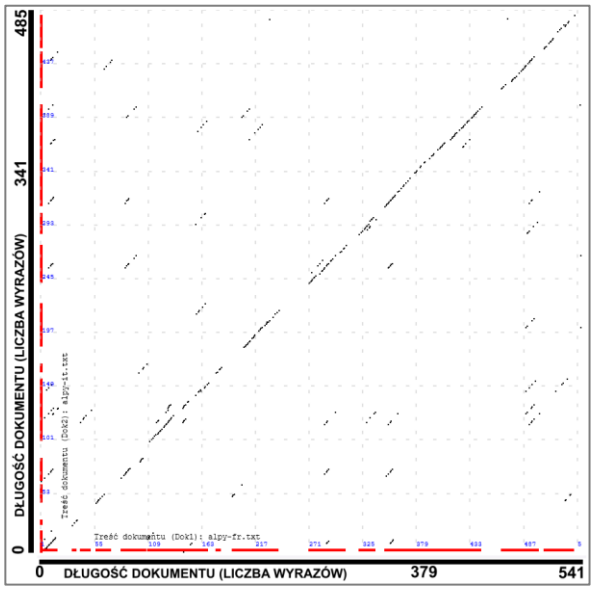
Artykuł opisuje Alpy - europejski łańcuch górski, jeden z najbardziej odwiedzanych przez turystów na świecie. Opis Alp wybrany został dlatego, że przebiega przez m.in. Francję i Włochy, których języki będą w tym rozdziale analizowane. Tekst w języku polskim (rozdz. 7.9.1)[93] przetłumaczony został na dwa języki: francuski (7.9.2)[94] oraz włoski (7.9.3)[95] za pomocą zaawansowanego modelu językowego opracowanego przez firmę OpenAI – ChatGPT w wersji 4. Teksty zostały między sobą porównane, a graficzna interpretacja porównanych tekstów znajduje się poniżej (rys. 4.31-4.34). Wyniki analizy porównawczej dadzą odpowiedź

[93] Tekst dostępny pod adresem: https://antyplagius.n-dms.com/tests/French-Italian/alpy-pl.txt
[94] Tekst dostępny pod adresem: https://antyplagius.n-dms.com/tests/French-Italian/alpy-fr.txt
[95] Tekst dostępny pod adresem: https://antyplagius.n-dms.com/tests/French-Italian/alpy-it.txt

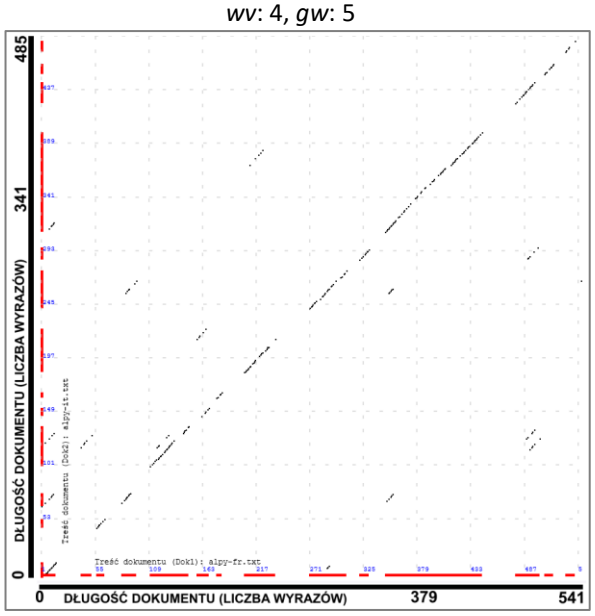
na pytanie, czy również w przypadku tych języków zadziała prawidłowo metoda macierzowej analizy danych tekstowych bazująca na odległości edycyjnej – metoda niezależna od języka.



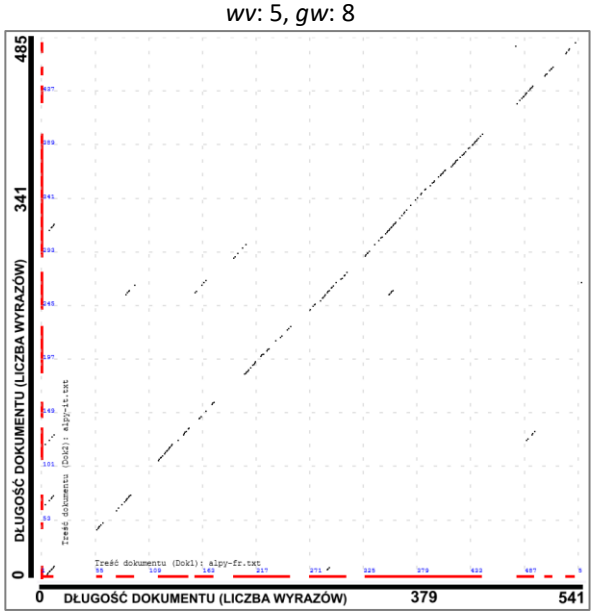
Rysunek 4.31. Parametry analizy – *bp*: 70%, *wv*: 4, *gw*: 5



Rysunek 4.32. Parametry analizy – *bp*: 40%, *wv*: 5, *gw*: 8



Rysunek 4.33. Parametry analizy – *bp*: 40%, *wv*: 6, *gw*: 8



Rysunek 4.34. Parametry analizy – bp: 50%, *wv*: 6, *gw*: 8

Interesującym wynikiem analizy porównawczej tekstów w postaci graficznej jest rysunek 4.31. Jest to wykres o najmniejszym szumie, czyli terminach, które mogą zostać uznane za zbędne w procesie porównania dokumentów. Wynik numeryczny podobieństwa to zaledwie ok. 10% (tabela 4.15, test ID 1). Za pomocą graficznej interpretacji można jednak wywnioskować podobieństwo pomiędzy dokumentami dzięki kształtującej się linii, pomimo słabego wyniku w postaci wspomnianej wartości liczbowej. Zmniejszenie stopnia podobieństwa pomiędzy

terminami (*bp*), przy jednoczesnym zwiększeniu dopuszczalnej przerwy pomiędzy wyrazami w wektorze ciągłości do *gw=8* sprawia, że wynik się poprawia, ale pojawia się nieznaczny szum. Najbardziej właściwym zestawem parametrów dla porównania tych języków wydaje się test ID 4 ujęty w tabeli poniżej (4.15) oraz na rysunku 4.34.

ID	**Język dokumentu (1)**	**Język dokumentu (2)**	***bp***	***wv***	***gw***	***Liczba wyrazów (liczba znaków) (1) x (2)***	**WYNIK (1)**	**WYNIK (2)**
1	Francuski	Włoski	70%	4	5	552 X 487 (3019 X 2738)	8,79%	9,90%
2	Francuski	Włoski	40%	5	8	552 X 487 (3019 X 2738)	53,48%	56,08%
3	Francuski	Włoski	40%	6	8	552 X 487 (3019 X 2738)	41,58%	45,36%
4	Francuski	Włoski	50%	6	8	552 X 487 (3019 X 2738)	37,73%	40,82%

Tabela 4.15. Wynik analizy porównawczej tekstu napisanego w językach niemieckim i niderlandzkim

ID	Język francuski	Język włoski
1	[...] Alpes occidentales, centrales et orientales, qui diffèrent géologiquement et topographiquement. Les Alpes occidentales sont les plus massives avec les sommets les plus élevés, tandis que les Alpes orientales sont plus basses et [...]	[...] Alpi Occidentali, Centrali e Orientali, che differiscono geologicamente e topograficamente. Le Alpi Occidentali sono le più massicce con le vette più alte, mentre le Alpi Orientali sono più basse e [...]
2	[...] Flore et faune La biodiversité des Alpes est impressionnante – la région abrite plus de espèces de plantes et de nombreuses espèces animales. La flore des Alpes varie avec laltitude, des forêts denses de feuillus et de conifères dans les basses altitudes aux prairies alpines fleuries et [...]	[...] Flora e fauna La biodiversità delle Alpi è impressionante – la regione ospita oltre . specie di piante e molte specie animali. La flora alpina cambia con laltitudine, da fitte foreste di latifoglie e conifere nelle altitudini più basse a praterie alpine fiorite e [...]
3	[...] le marmot, laigle royal et lours brun, bien que ce dernier soit maintenant rarement rencontré. Culture et histoire Les Alpes sont également une région riche en histoire et en culture. Depuis des temps immémoriaux, ces montagnes ont été un [...]	[...] la marmotta, laquila reale e lorso bruno, sebbene questultimo sia ora raramente incontrato. Cultura e storia Le Alpi sono anche una regione ricca di storia e cultura. Da tempi immemorabili, queste montagne sono state un [...]

Tabela 4.16. Wybrane wyniki analizy tekstów uznanych za podobne dla analizy ID 4 z tabeli 4.15

W tabeli 4.16 ujęte zostały fragmenty z analizy ID 4. Widać pomiędzy tekstami zachodzące podobieństwo, pomimo, że prawie każdy wyraz tworzących jedno zdanie jest inny w swojej budowie względem odpowiednika w drugim zdaniu. Stąd konieczność wprowadzenia takiego zestawu parametrów, co jest wynikiem różnic w słownictwie pomiędzy francuskim a włoskim. Konsekwencją ustawienia parametru podobieństwa wyrazów *bp=100%* byłby wynik świadczący o różności dokumentów.

4.7. Analiza dokumentów tekstowych napisanych w językach: hiszpańskim i rumuńskim[96]

Język rumuński to język romański używany głównie w Rumunii i Mołdawii. Jest oficjalnym językiem tych dwóch krajów oraz jednym z 24 oficjalnych języków Unii Europejskiej. Język hiszpański, również należący do rodziny języków romańskich, jest oficjalnym językiem w Hiszpanii, krajach Ameryki Łacińskiej oraz Gwinei Równikowej.

Wspólne cechy języków rumuńskiego i hiszpańskiego obejmują:

- Gramatyka: obie mają zbliżone systemy gramatyczne, takie jak deklinacje rzeczowników, odmiany czasowników, przymiotniki, itp. Obie używają systemu rodzajów gramatycznych (męski i żeński) oraz liczby pojedynczej i mnogiej.
- Słownictwo: wiele słów jest wspólnych lub bardzo podobnych w obu językach, choć występują różnice w pisowni i wymowie. Przykłady: "bună dimineața" (rumuński) / "buenos días" (hiszpański) - dzień dobry, "mulțumesc" (rumuński) / "gracias" (hiszpański) - dziękuję, "casă" (rumuński) / "casa" (hiszpański) - dom.
- Alfabet: obie używają alfabetu łacińskiego. W przypadku rumuńskiego występują dodatkowe litery, takie jak "ă", "â", "î", "ș" i "ț".

Mimo że zarówno rumuński, jak i hiszpański wywodzą się z łaciny, języka starożytnego Rzymu, co sprawia, że mają wiele wspólnych cech i podobieństw, istnieją też różnice, szczególnie w dialektach, wymowie oraz niektórych aspektach gramatycznych i słownictwa. Warto też zauważyć, że język rumuński zachował również pewne wpływy słowiańskie, które odróżniają go od innych języków romańskich, takich jak hiszpański. Niemniej jednak, osoby, które znają jeden z tych języków, często mają łatwiejsze zadanie w nauce drugiego ze względu na liczne podobieństwa.

Badania zawarte w tym podrozdziale sprawdzą, czy da się obliczyć podobieństwo metodą macierzowej analizy danych tekstowych, która bazuje na odległości edycyjnej, pomiędzy tekstami napisanymi w językach hiszpańskim i rumuńskim, pomimo, że są najsłabiej do siebie podobne we wspólnej grupie językowej. Dodatkowo język rumuński jest jedynym językiem z

[96] Więcej przykładów analizy tych języków znajduje się na stronie oficjalnego kanału programu Antyoplagius: https://www.youtube.com/watch?v=JhfdwbyIsFc&list=PLPFeTDhxdQPawnjGhPytgFJeJb-YOXmHC&index=12

powyższych, w którym rodzajniki dołączane są na końcu rzeczownika, a nie na początku, co znacząco zmienia strukturę wyrazów i może utrudnić analizę podobieństwa.

Ciągi tekstowe poddane analizie pochodzą ze źródeł internetowych w postaci artykułów encyklopedycznych z dwóch różnych znanych encyklopedii. Poddane zostały tłumaczeniu przez translatory bazujące na rozwiązaniach z dziedziny sztucznej inteligencji[97], które obecnie uznawane są za wyjątkowo skuteczne. Oba testy reprezentuje odrębne podejścia związane z tłumaczeniem. Pierwsze podejście to dostosowanie jednego z języków do drugiego poprzez przetłumaczenie tego pierwszego (TEST 1). Drugi test rozpoczyna się przetłumaczeniem języka angielskiego na dwa badane języki (TEST 2). Przedstawione podejścia są dowiązaniem do badań jakie mają miejsce w kwestiach związanych z plagiatami typu *cross-language*, popełnianymi coraz częściej na świecie w szkołach oraz na uczelniach[37,38].

TEST 1. Artykuł encyklopedyczny o Hiszpanii

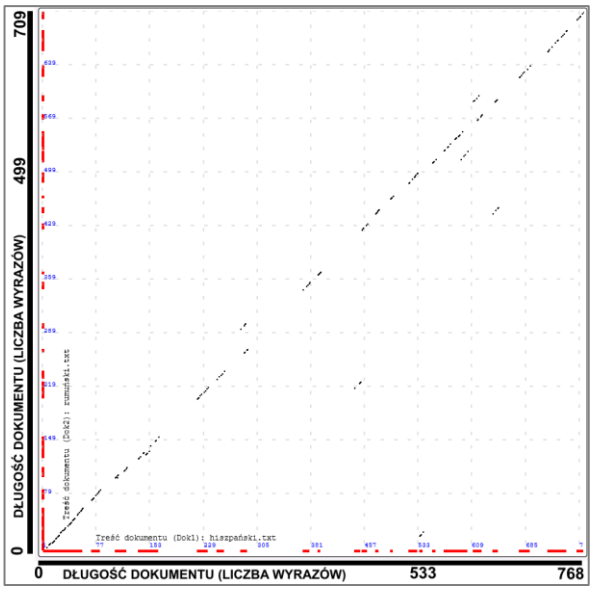
W tej analizie użyty został artykuł encyklopedyczny o Hiszpanii (rozdz. 7.10.1)[98] w języku hiszpańskim przetłumaczony przez translator na język rumuński (rozdz. 7.10.2)[99]. Teksty zostały między sobą porównane, dodatkowo jedna z analiz została zamieszczona w postaci filmu na kanale YouTube[100]. Graficzna interpretacja porównanych tekstów znajduje się poniżej. Stała *gw* będzie taka sama dla wszystkich testów w rozdziale, ponieważ specyfika problemu nie wymusza jej ciągłego dostosowywania do tekstu.

[97] translatory używane w tłumaczeniach: https://translate.google.com oraz https://www.bing.com/translator

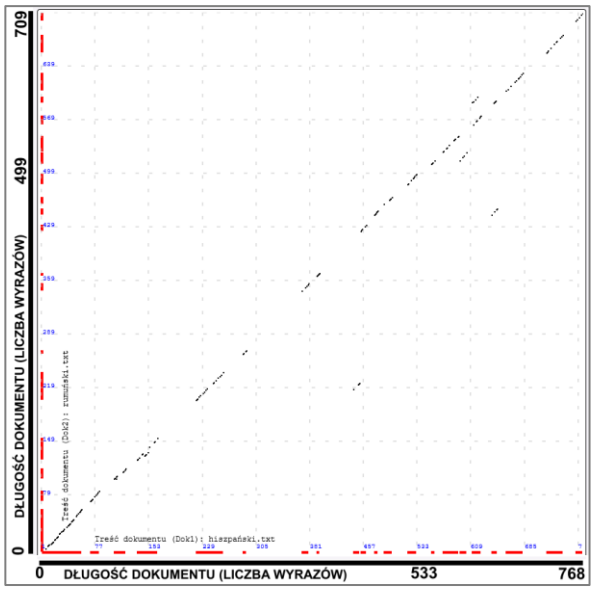
[98] https://es.wikipedia.org/wiki/Espa%C3%B1a

[99] https://antyplagius.n-dms.com/tests/Spanish-Romanian/Espania-Romanian-wikipedia-google-translate.txt, https://antyplagius.n-dms.com/tests/Spanish-Romanian/Espania-Spanish-wikipedia-google-translate.txt

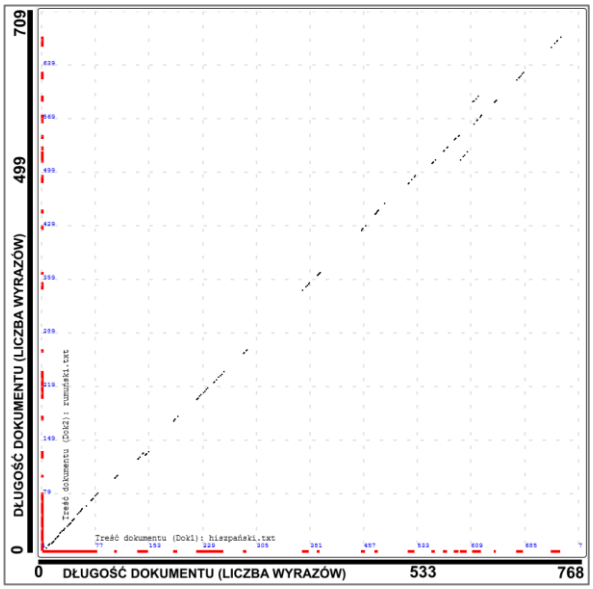
[100] https://www.youtube.com/watch?v=JhfdwbyIsFc



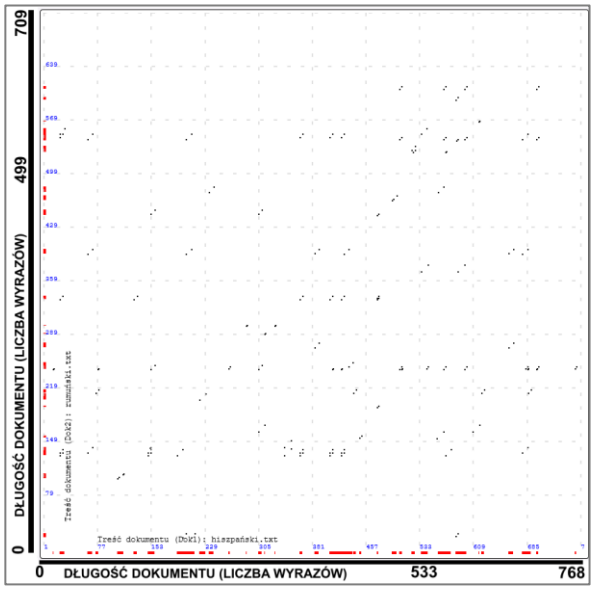
Rysunek. 4.35. Parametry analizy – *bp*: 42%, *wv*: 5, *gw*: 8



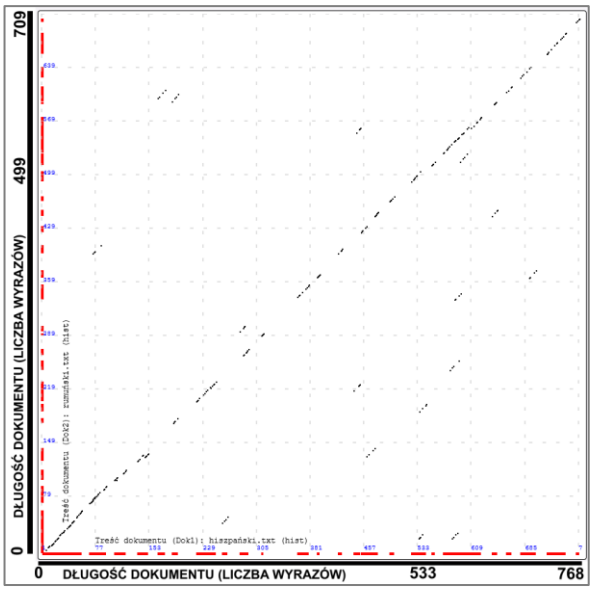
Rysunek. 4.36. Parametry analizy – *bp*: 45%, *wv*: 5, *gw*: 8



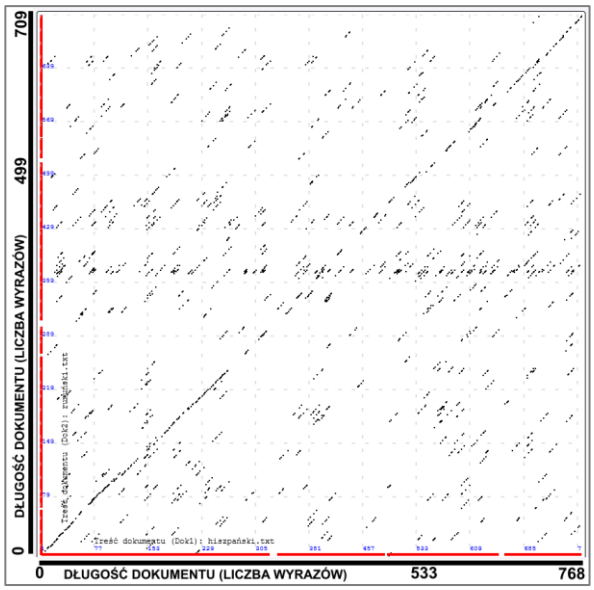
Rysunek 4.37. Parametry analizy – *bp*: 50%, *wv*: 5, *gw*: 8



Rysunek 4.38. Parametry analizy – bp: 100%, *wv*: 3,*gw*: 8



Rysunek 4.39. Parametry analizy – bp: 40%, *wv*: 5,*gw*: 8



Rysunek 4.40. Parametry analizy – bp: 30%, *wv*: 5,*gw*: 8

ID	Język dokumentu (1)	Język dokumentu (2)	*bp*	*wv*	*Liczba wyrazów (liczba znaków) (1) x (2)*	WYNIK (1)	WYNIK (2)
1	Hiszpański	Rumuński	42%	5	768 X 709 (4807 X 4651)	24,61%	25,81%
2	Hiszpański	Rumuński	45%	5	768 X 709 (4807 X 4651)	23,96%	25,25%
3	Hiszpański	Rumuński	50%	5	768 X 709 (4807 X 4651)	19,27%	20,45%
4	Hiszpański	Rumuński	90%	5	768 X 709 (4807 X 4651)	0,0%	0,0%
5	Hiszpański	Rumuński	90%	3	768 X 709 (4807 X 4651)	13,54%	10,16%
6	Hiszpański	Rumuński	100%	5	768 X 709 (4807 X 4651)	0,0%	0,0%
7	Hiszpański	Rumuński	100%	3	768 X 709 (4807 X 4651)	12,89%	9,45%
8	Hiszpański	Rumuński	40%	5	768 X 709 (4807 X 4651)	30,73	33,57
9	Hiszpański	Rumuński	30	5	768 X 709 (4807 X 4651)	79,17	79,83

Tabela 4.17. Wyniki badania podobieństwa tekstów, z różnymi parametrami analizy

Na rysunkach 4.35-4.40 i na podstawie tabeli 4.17 widać, że najlepszy rezultat ukazujący znaczne podobieństwo pomiędzy tekstami uzyskuje się poprzez ustawienie podobieństwa wyrazów na poziomie między 42% a 50%. Ustawienie wartości podobieństwa wyrazów (bw) poniżej 42% dla powyższego przykładu sprawia, że graficzny rezultat porównania jest mniej czytelny, pojawia się szum, a linia ukośna (ewentualnie mniejsze linie ukośne), która można powiedzieć odpowiada za wizualne potwierdzenie podobieństwa tekstów, jest mniej widoczna i rozmywa się w szumie. Zmniejszenie stałej *bw* do 0% będzie generowało wynik 100% podobieństwa pomiędzy dokumentami – co będzie oczywistym błędem. Zwiększenie stopnia podobieństwa wyrazów bliżej 100% będzie skutkowało zanikiem punktów na macierzy i brakiem widocznego podobieństwa pomiędzy ciągami tekstowymi.

ID	Język hiszpański	Język rumuński
1	[…] En Europa, ocupa la mayor parte de la península ibérica, conocida como España […]	[…] În Europa, ocupă cea mai mare parte a Peninsulei Iberice, cunoscută drept Spania [...]
2	[...] de facto del G. La primera presencia constatada de homínidos del género Homo se remonta a , millones de años antes del presente, como atestigua el descubrimiento [...]	[...] de facto membră a G. Prima prezență confirmată a hominidelor din genul Homo datează cu , milioane de ani înainte de prezent, fapt dovedit de descoperirea [...]
3	[...] monarcas españoles dominaron el primer imperio de ultramar global, que abarcaba territorios en los cinco continentes,nota dejando un vasto acervo cultural y lingüístico por el globo. [...]	[...] monarhii spanioli dominau primul imperiu global de peste mări, care cuprindea teritorii de pe cinci continente, nota lăsând o vastă moștenire culturală și lingvistică pe tot globul. [...]

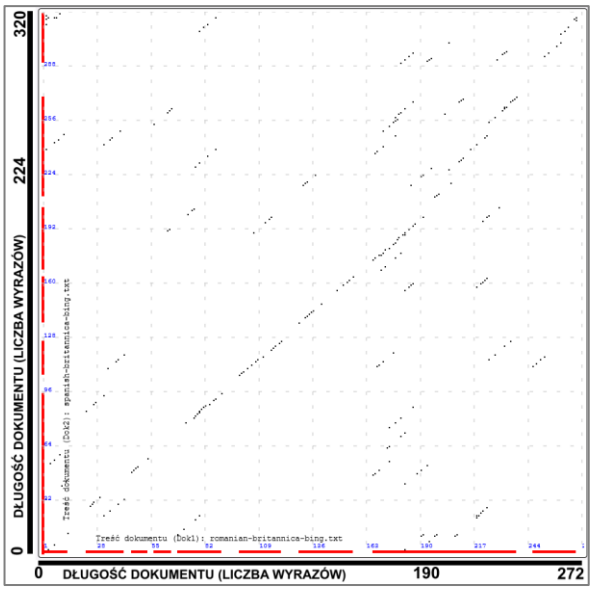
Tabela 4.18. Wybrane fragmenty tekstów uznane za podobne w ramach analizy ID 1 z tabeli 4.17

Powyższa tabela (4.18) zawiera wybrane fragmenty uznane za podobne będące wynikiem analizy porównania omawianych tekstów. Każdy z powyższych wyrazów uznany za podobny do swojego odpowiednika w drugim tekście ma swoją interpretację graficzną w postaci punktu na macierzy.

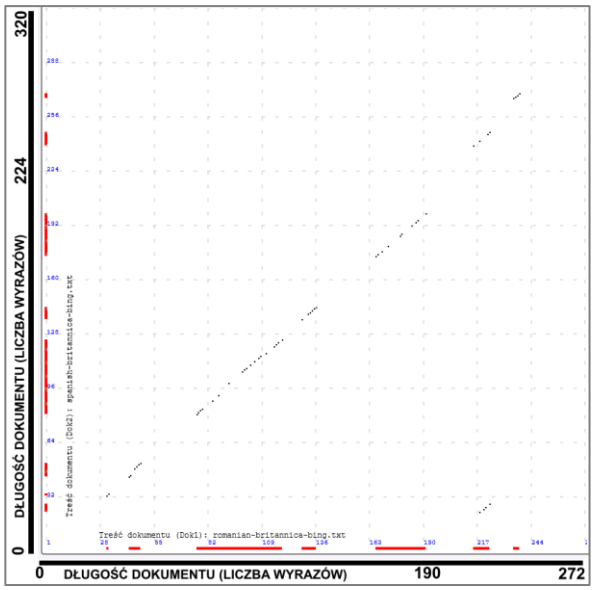
TEST 2. Artykuł encyklopedyczny o Rumunii

W powyższej analizie zestawione zostały dwa teksty o Rumunii pochodzące z encyklopedii internetowej[101]. Wcześniej tekst w języku angielskim został przetłumaczony przez translator innej firmy[102] na języki rumuński (rozdz. 7.10.4)[103] i hiszpański (rozdz. 7.10.3)[104]. Poniżej znajduje się interpretacja graficzna porównania.

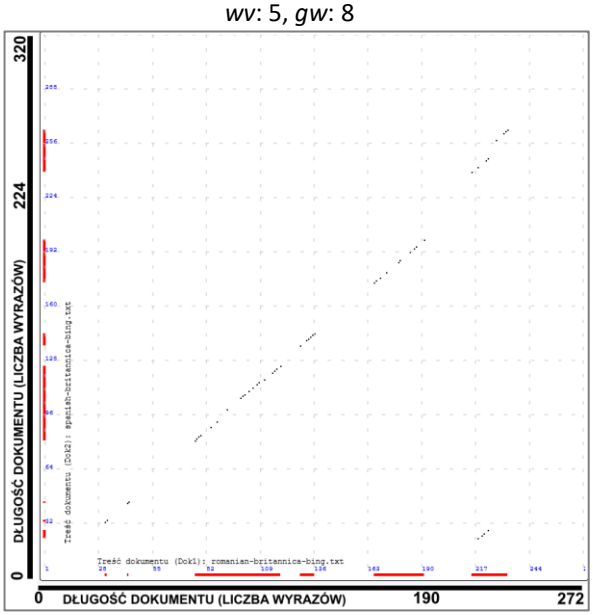
[101] https://www.britannica.com/place/Romania
[102] https://www.bing.com/translator
[103] Treść dostępna pod adresem: https://antyplagius.n-dms.com/tests/Spanish-Romanian/romanian-britannica-bing.txt
[104] Treść dostępna pod adresem: https://antyplagius.n-dms.com/tests/Spanish-Romanian/spanish-britannica-bing.txt



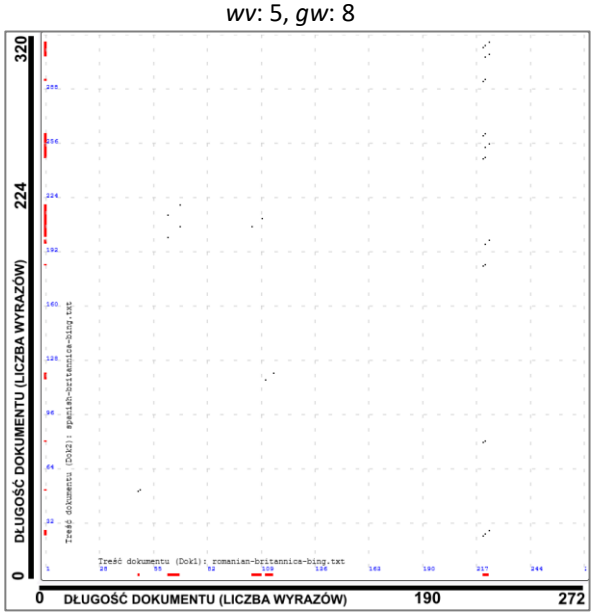
Rysunek 4.41. Parametry analizy – *bp*: 30%, *wv*: 5, *gw*: 8



Rysunek 4.42. Parametry analizy – *bp*: 42%, *wv*: 5, *gw*: 8



Rysunek 4.43. Parametry analizy – *bp*: 50%, *wv*: 5, *gw*: 8



Rysunek 4.44. Parametry analizy – bp: 100%, *wv*:3,*gw*: 8

Podobnie jak w poprzednim teście, najlepiej dopasowane parametry dla analizy tych dwóch języków, to podobieństwo z przedziału 42% do 50%.

ID	Język dokumentu (1)	Język dokumentu (2)	*bp*	*wv*	*Liczba wyrazów (liczba znaków) (1) x (2)*	WYNIK (1)	WYNIK (2)
1	Hiszpański	Rumuński	35%	5	320 X 272 (1961 X 1798)	18,44%	22,43%
2	Hiszpański	Rumuński	45%	5	320 X 272 (1961 X 1798)	18,44%	20,96%
3	Hiszpański	Rumuński	40%	5	320 X 272 (1961 X 1798)	18,12%	20,59%
4	Hiszpański	Rumuński	42%	5	320 X 272 (1961 X 1798)	17,19%	19,49%
5	Hiszpański	Rumuński	50%	5	320 X 272 (1961 X 1798)	15,94%	18,01%
6	Hiszpański	Rumuński	100%	5	320 X 272 (1961 X 1798)	0,0%	0,0%
7	Hiszpański	Rumuński	100%	3	320 X 272 (1961 X 1798)	9,69%	4,04%

Tabela 4.19. Wyniki badania podobieństwa tekstów, z różnymi parametrami analizy

ID	Język hiszpański	Język rumuński
1	[...] , când regimul liderului român Nicolae Ceaușescu [...]	[...] , cuando el régimen del líder rumano Nicolae Ceaușescu [...]
2	[...] a Uniunii Europene UE. Peisajul românesc este de aproximativ o treime muntos și o treime împădurit, restul fiind alcătuit din dealuri și câmpii. Clima este temperată și marcată de patru anotimpuri distincte. România se bucură de o bogăție considerabilă de resurse naturale terenuri fertile [...]	[...] la Unión Europea UE. El paisaje rumano es aproximadamente un tercio montañoso y un tercio boscoso, con el resto formado por colinas y llanuras. El clima es templado y marcado por cuatro estaciones distintas. Rumania goza de una considerable riqueza de recursos naturales tierras fértiles [...]
3	[...] român derivă o mare parte din caracterul său etnic și cultural din influența romană, dar această identitate străveche a fost remodelată continuu de poziția României pe [...]	[...] rumano deriva gran parte de su carácter étnico y cultural de la influencia romana, pero esta antigua identidad ha sido remodelada continuamente por la posición de [...]
4	[...] dacilor care locuiau în munții de la nord de Câmpia Dunăreană și în Bazinul Transilvaniei. Până la retragerea romană sub împăratul Aurelian în , [...]	[...] dacios que vivían en las montañas al norte de la llanura del Danubio y en la cuenca de Transilvania. En el momento de la retirada romana bajo el emperador Aureliano en , [...]

Tabela 4.20. Wybrane fragmenty tekstów uznane za podobne dla analizy ID 2 z tabeli 4.19

Powyższa tabela (4.20) zawiera wybrane fragmenty tekstów uznane za podobne przez algorytm. Dane są rezultatem analizy na podstawie parametrów ID 2 z tabeli 4.19.
Opierając się o powyższe wyniki da się zauważyć, że algorytm, w którym nie są zaimplementowane reguły gramatyczne dotyczące danego języka w ramach tej samej grupy językowej jest w stanie prawidłowo oszacować istniejące podobieństwo pomiędzy tekstami pomimo dodatkowych różnic wynikających z różnych języków.

Powyższe wyniki zostało przedstawione w publikacji *Matrix similarity analysis of texts written in Romanian and Spanish*[39].

4.8. Analiza podobieństwa wypracowań generowanych przez sztuczną inteligencję[105]

W ostatnich latach obserwujemy dynamiczny rozwój technologii sztucznej inteligencji (AI), które coraz śmielej wkraczają w przestrzeń akademicką i edukacyjną. AI, jako narzędzie wspomagające proces pisania, otwiera przed studentami i uczniami nowe perspektywy w zakresie efektywności i kreatywności tworzenia tekstów. Narzędzia te, takie jak zaawansowane algorytmy generujące treści [43,48], mogą służyć jako pomoc w rozwijaniu umiejętności pisarskich, oferując sugestie stylistyczne, gramatyczne, a także wspierając rozwój

[105] Więcej przykładów dostępnych jest na oficjalnej stronie kanału projektu: https://www.youtube.com/watch?v=_ejk1xTPDDQ oraz https://www.youtube.com/watch?v=PxrVB9AwcR0

struktury i argumentacji wypracowań. Jednakże, ten postęp nie jest wolny od wyzwań. Z jednej strony, sztuczna inteligencja może być wartościowym asystentem, ułatwiającym pracę nad tekstem, z drugiej jednak, rodzi obawy związane z uczciwością akademicką. W kontekście edukacyjnym, niezwykle ważne jest, aby potrafić rozróżnić wytwór ludzki od komputerowego. Problem ten nabiera szczególnego znaczenia w świetle łatwości, z jaką AI może generować kompleksowe teksty, potencjalnie wyręczając użytkowników w ich własnych wysiłkach intelektualnych.

W niniejszym podrozdziale podjęto próbę analizy tekstów wygenerowanych przez sztuczną inteligencję (chatGPT OpenAI różnych wersji GPT – 3.5,4,4o) w ramach tematyki z pochodzącej z odmiennych dziedzin. Odpowiedzi zostały wielokrotnie wygenerowane, aby zbadać możliwość wykrycia potencjalnych podobieństw pomiędzy wypracowaniami, mogących sugerować nieuczciwość autora. Testy mają za zadanie sprawdzić, czy za pomocą opisanego w pracy algorytmu, można sprawdzić podobieństwo pomiędzy tekstami napisanymi przez sztuczną inteligencję i ewentualnie poprzez to wywnioskować źródło pochodzenia danego wypracowania. Teksty pierwszego testu dotyczącego wypracowania na temat języka programowania JavaScript zostały zamieszczone w rozdziale 7.4 (Załączniki). Pozostałe teksty kolejnych analiz ze względu na swoją objętość umieszczone zostały w zasobach zewnętrznych, do których prowadzą linki w przypisach dolnych.

TEST 1. Wypracowanie na temat języka programowania JavaScript

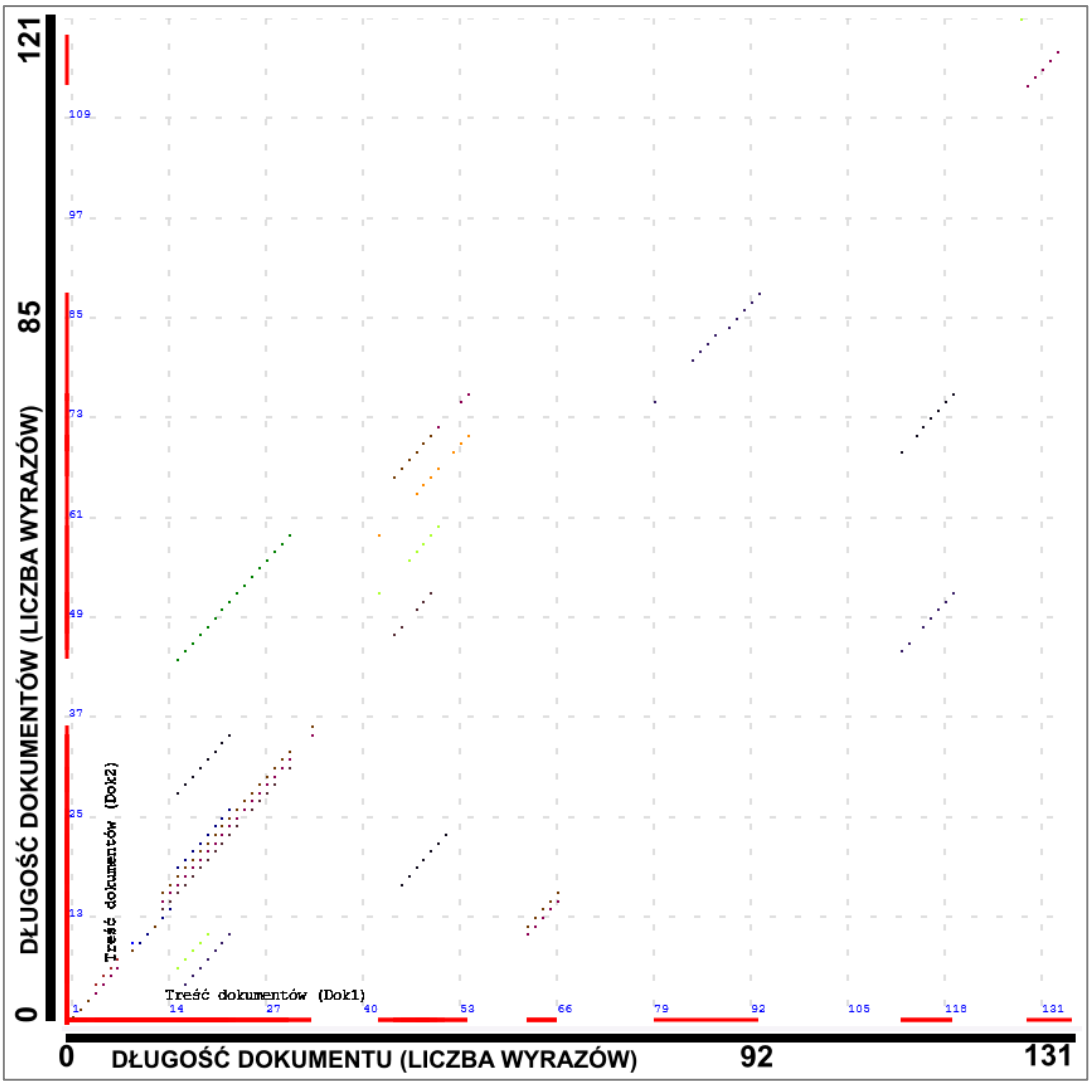
Test dotyczy analizy wypracowania na temat popularnego języka programowania związanego z budowaniem stron internetowych o nazwie JavaScript[106]. Teksty zostały wygenerowane przez model GPT 3.5 oraz 4.0 w liczbie odpowiedzi: 25 (zawartość w rozdziale 7.4)[107]. Treść prośby do modelu GPT: „Czym jest język JavaScript?”. Przykład analizy w postaci filmu zamieszczony został na kanale YouTube[108].

[106] https://encyklopedia.pwn.pl/haslo/JavaScript;3917305.html
[107] Treść tekstów: https://antyplagius.n-dms.com/tests/chatGPT-JavaScript/
[108] Fragment analizy w postaci filmu zamieszczony pod adresem: https://youtu.be/PxrVB9AwcR0?si=WTntBMZN-k7E2Yxx

ID analizy	PLIK (1)	PLIK (2)	WYNIK (1) [%]	WYNIK (2) [%]
44	13.txt	15.txt	26,47	19,43
28	12.txt	13.txt	25,61	31,62
29	13.txt	16.txt	21,32	15,51
5	13.txt	17.txt	17,65	18,6
19	13.txt	3.txt	16,18	14,86
37	1.txt	13.txt	15,15	14,71
67	13.txt	18.txt	13,24	9,09
50	13.txt	14.txt	11,76	10
36	11.txt	13.txt	11,11	16,18
12	10.txt	13.txt	8,23	9,56
63	13.txt	19.txt	7,35	6,54

Tabela 4.21. Wyniki analizy porównania wypracowań napisanych przez AI dla podobieńst powyżej 0%

Rezultaty przeprowadzonych badań, zestawione w tabeli 4.21 zostały zinterpretowane na wykresie 4.45 przedstawiajacym podobieństwo pomiędzy plikami. Wszystkich analiz było 300, z czego 11 wykazało minimalne podobieństwo na podstawie wprowadzonych parametrów (tab. 4.21). Każda z analiz zwróciła inne wartości podobieństwa. Niektóre teksty były na tyle różne, że wynik podobieństwa wyniósł 0%, a niektóre teksty na tyle podobne, że ich podobeństwo to ponad 25%. Na wykresie (4.45) uwzględnione zostały wszystkie analizy, których podobieństwo wyniosło ponad 0% – na osi poziomej umieszczony został porównywany dokument, a na pionowej zbiór dokumentów z uwzględnieniem odpowiednio adekwatnej pozycji wyrazów w tekstach. Kolory punktów na wykresie poniżej pokrywają się z kolorem nazw plików, w których jest zawartość badanego tekstu.



Rysunek 4.45. Interpretacja graficzna porównania wypracowania zawartego w pliku [13.txt] z pozostałymi tekstami. Parametry analizy – *bp*: 60%, *wv*: 6, *gw*: 8

Wynik analizy w postaci wykresu 4.45 pokazuje podobieństwo pomiędzy przykładowym plikiem [13.txt] (oś pozioma) a pozostałymi plikami tekstowymi. Oś pionowa zawiera zbiór plików wykazujących przynajmniej mininalne podobieństwo z plikiem porównywanym i są to pliki: [19.txt], [10.txt], [11.txt], [14.txt], [18.txt], [1.txt], [17.txt], [16.txt], [12.txt], [15.txt], [3.txt]. Na wykresie i na podstawie tabeli widać, że pliki są do siebie podobne niewielkimi fragmentami, ale zebrane w całość tworzą wyraźny obraz podobieństwa pomiędzy tekstem zawartym w pliku [13.txt] – co widać na osiach poprzez zaznaczenie kolorem czerwonym.

Ostateczne podsumowanie wyniku podobieństwa tekstu zawartego w pliku [13.txt] względem wszystkich zbadanych plików wygenerowanych przez AI to: 60% (82 wyrazy podobne / 136 wszystkich wyrazów w dokumencie porównywanym [13.txt]).

Oznacza to, że metoda macierzowej analizy danych tekstowych bazująca na odległości edycyjnej w powyższej konfiguracji (jeden plik sprawdzany względem wielu plików analizowanych) bardzo dobrze sprawdza się w analizie podobieństwa tekstów i dodatkowo w tym przypadku pokazuje jedno źródło pochodzenia wypracowań, czyli algorytm generatywnej sztucznej inteligencji[109][43,48] – które jak widać na wykresie, nie jest mechanizmem doskonałym w aspekcie tworzenia unikalnych wypracowań. Zapewne, gdyby wygenerowanych odpowiedzi było więcej, to podobieństwo byłoby jeszcze wyższe.

Poniżej znajduje się fragment tekstu z pliku [13.txt] uznanego za podobny względem zbioru pozostałych dokumentów. Wszystkie terminy uwzględnione w tekście to terminy uznane za podobne występujące w różnych badanych dokumentach. Indeks przy terminie wskazuje na numer kolejności terminu w pliku [13.txt]. Im większa różnica ma miejsce pomiędzy wartościami indeksów, tym mniej wykrytych w danym fragmencie podobieństw.

JavaScript[1] to[2] wysokopoziomowy,[3] dynamiczny[4] język[5] programowania,[6] który[7] jest[8] głównie[9] używany[10] w[11] kontekście[12] tworzenia[13] stron[14] internetowych.[15] Został[16] stworzony[17] przez[18] Brendana[19] Eicha[20] w[21] 1995[22] roku[23] i[24] pierwotnie[25] nazywał[26] się[27] Mocha,[28] a[29] następnie[30] LiveScript.[31] Nazwa[32] "JavaScript"[33] została[34] ... ECMAScript.[43]JavaScript[44] jest[45] językiem[46] skryptowym,[47] co[48] oznacza,[49] że[50] jest[51] interpretowany[52] przez[53] przeglądarkę[54] internetową,[55] ... tworzenie[63]interaktywnych[64] i[65] dynamicznych[66] stron[67] ... początkowo[80]JavaScript[81] był[82] używany[83] głównie[84] w[85] przeglądarkach[86] internetowych,[87] z[88] biegiem[89] czasu[90] jego[91] zastosowanie[92] rozszerzyło[93] się[94] ... obsługuje[113]różne[114] paradygmaty[115] programowania,[116] takie[117] jak[118] programowanie[119] obiektowe,[120] ... może[130]być[131] używany[132] do[133] tworzenia[134] różnorodnych[135] aplikacji.[136]

[109] Generatywna sztuczna inteligencja jest formą sztucznej inteligencji, która może tworzyć tekst, obrazy i zróżnicowane treści w oparciu o dane, na których jest szkolona.

ID analizy	Plik (1)	Plik (2)	Fragment tekstu (1)	Fragment tekstu (2)
63	13.txt	19.txt	[...] język programowania, który jest głównie używany [...]	[...] język programowania, który jest często używany [...]
12	13.txt	10.txt	[...] Został stworzony przez Brendana Eicha w [...]	[...] został opracowany przez Brendana Eicha w [...]
36	13.txt	11.txt	[...] JavaScript to wysokopoziomowy, dynamiczny język programowania, który jest głównie używany kontekście tworzenia stron internetowych. Został stworzony przez Brendana Eicha w 1995 roku [...]	[...] JavaScript to wysokopoziomowy, dynamiczny język programowania, który jest często używany w kontekście tworzenia stron internetowych i aplikacji webowych. Początkowo został stworzony przez Brendana Eicha w 1995 roku [...]
19	13.txt	3.txt	[...] obsługuje różne paradygmaty programowania, takie jak programowanie obiektowe, [...]	[...] Obsługuje wiele paradygmatów programowania, takich jak programowanie obiektowe, [...]
28	13.txt	12.txt	[...] JavaScript to wysokopoziomowy, dynamiczny język programowania, który jest głównie używany w kontekście tworzenia stron internetowych. Został stworzony przez Brendana Eicha w 1995 roku i pierwotnie nazywał się Mocha, a następnie LiveScript. Nazwa "JavaScript" została [...]	[...] JavaScript to wysokopoziomowy, interpretowany język programowania, który jest używany głównie do tworzenia interaktywnych i dynamicznych stron internetowych. Został stworzony przez Brendana Eicha w 1995 roku i pierwotnie nazywał się Mocha, a następnie LiveScript, zanim ostatecznie został [...]

Tabela 4.22. Wyniki analizy tesktów uznanych za podobne w ramach porównania pliku [13.txt] względem pozostałych[110]

Tabela powyżej (4.22) zawiera wybrane fragmenty tekstów z poszczególnych analiz porównania pliku [13.txt] z pozostałymi. Zaprezentowane wyniki pokazują, że teksty nie są identyczne, ale pewne elementy są podobne. Pomimo tego, że zaproponowana w pracy metoda analizy macierzowej tekstów bazująca na odległości edycyjnej nie opiera się na słowniku wyrazów bliskoznacznych i poszczególne terminy nie są sprowadzane do form podstawowych – to widać, że jest wystarczająco skuteczna w wykrywaniu podobieństwa.

TEST 2. Recenzja filmu Troja (2004)

Kolejny test polega na analizie tekstu, który nie jest tekstem technicznym. Wypracowania zostały wygenerowane przez modele GPT 3.5 oraz 4.0 w liczbie 31[111]. Treść prośby do modelu GPT: „Napisz swoją krótką opinię o filmie Troja z 2004 roku Wolfganga

[110] Pełny raport porównania tekstów wyeksportowany z programu znajduje się pod adresem: https://antyplagius.n-dms.com/tests/chatGPT-JavaScript/chatGPT-13.txt.pdf

[111] Treść tekstów: https://antyplagius.n-dms.com/tests/chatGPT-Troja/

Petersena". Przykład porównania w postaci filmu zamieszczony został na kanale YouTube[112]. Analiza podobnie jak w poprzednim przykładzie polegała na porównaniu każdego wypracowania z każdym. Wykres 4.46 i tabela 4.23 poniżej przedstawiają podobieństwo pomiędzy wybranym do analizy plikiem [1-Troja-chatGPT-4.txt] (oś pozioma) z pozostałymi plikami, których podobieństwo jest większe od 0%. Oś pionowa zawiera zbiór plików, które wykazują podobieństwo względem pliku porównywanego na osi poziomej. Wykres i tabela podobnie jak w teście poprzedzającym, pokazują, że poszczególne fragmenty plików są tylko częściowo podobne, jednak w zestawieniu dają wyraźny obraz podobieństwa do wypracowania zawartego w pliku [1-Troja-chatGPT-4.txt].

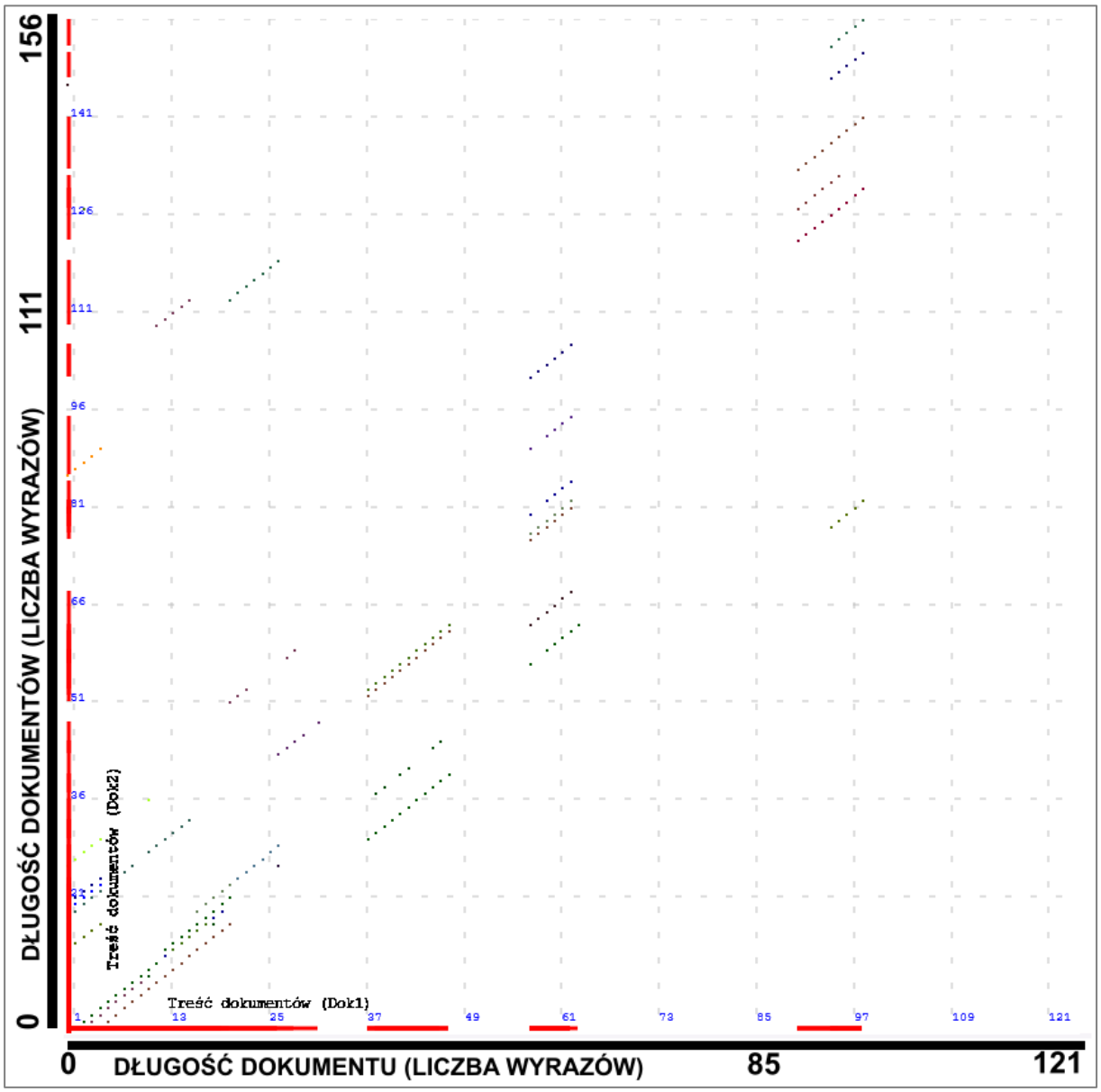
ID analizy	PLIK (1)	PLIK (2)	WYNIK (1) [%]	WYNIK (2) [%]
28	1-Troja-chatGPT-4.txt	14-Troja-chatGPT-4.txt	33,33	25,77
2	1-Troja-chatGPT-4.txt	2-Troja-chatGPT-4.txt	30,16	28,57
11	1-Troja-chatGPT-4.txt	5-Troja-chatGPT-4.txt	20,63	20,45
3	1-Troja-chatGPT-4.txt	17-Troja-chatGPT-4.txt	19,84	15,72
21	1-Troja-chatGPT-4.txt	18-Troja-chatGPT-4.txt	19,05	13,64
30	1-Troja-chatGPT-4.txt	21-Troja-chatGPT-4.txt	19,05	15,38
18	1-Troja-chatGPT-4.txt	19-Troja-chatGPT-4.txt	18,25	15,13
12	1-Troja-chatGPT-4.txt	16-Troja-chatGPT-4.txt	15,87	12,5
24	1-Troja-chatGPT-4.txt	4-Troja-chatGPT-4.txt	15,08	15,32
20	1-Troja-chatGPT-4.txt	15-Troja-chatGPT-4.txt	15,08	10,98
15	1-Troja-chatGPT-4.txt	9-Troja-chatGPT-4.txt	15,08	13,67
10	1-Troja-chatGPT-4.txt	13-Troja-chatGPT-4.txt	14,29	10,78
4	1-Troja-chatGPT-4.txt	20-Troja-chatGPT-4.txt	14,29	10,71
9	1-Troja-chatGPT-4.txt	12-Troja-chatGPT-4.txt	14,29	12,18
22	1-Troja-chatGPT-4.txt	7-Troja-chatGPT-4.txt	13,49	11,04
13	1-Troja-chatGPT-4.txt	10-Troja-chatGPT-4.txt	12,7	9,76
31	1-Troja-chatGPT-4.txt	3-Troja-chatGPT-4.txt	12,7	8,74
16	1-Troja-chatGPT-4.txt	25-Troja-chatGPT-4.txt	11,9	12,3
6	1-Troja-chatGPT-4.txt	30-Troja-chatGPT-3.5.txt	10,32	15,85
17	1-Troja-chatGPT-4.txt	24-Troja-chatGPT-4.txt	9,52	7,55
8	1-Troja-chatGPT-4.txt	22-Troja-chatGPT-4.txt	9,52	7,89
0	1-Troja-chatGPT-4.txt	6-Troja-chatGPT-4.txt	8,73	8,21
1	1-Troja-chatGPT-4.txt	26-Troja-chatGPT-3.5.txt	7,94	11,24
23	1-Troja-chatGPT-4.txt	8-Troja-chatGPT-4.txt	7,94	5,41
19	1-Troja-chatGPT-4.txt	11-Troja-chatGPT-4.txt	7,94	5,85
26	1-Troja-chatGPT-4.txt	28-Troja-chatGPT-3.5.txt	4,76	7,5
5	1-Troja-chatGPT-4.txt	23-Troja-chatGPT-4.txt	4,76	4,65
25	1-Troja-chatGPT-4.txt	31-Troja-chatGPT-3.5.txt	3,97	5,81

[112] Fragment analizy w postaci filmu zamieszczony pod adresem: https://youtu.be/_ejk1xTPDDQ?si=G1aA-IEnhPncAsXX

27	1-Troja-chatGPT-4.txt	29-Troja-chatGPT-3.5.txt	3,97	8,77
7	1-Troja-chatGPT-4.txt	27-Troja-chatGPT-3.5.txt	3,97	4,67

Tabela 4.23. Wyniki analizy porównania wypracowań napisanych przez AI dla podobieństw powyżej 0%

Każda z analiz zawarta w tabeli 4.23 została nałożona na poniższy wykres (4.46) obrazujący podobieństwo pomiędzy plikami. Wszystkich analiz było 465, z czego 30 wykazało minimalne podobieństwo na podstawie wprowadzonych parametrów (tab. 4.23).



Rysunek 4.46. Interpretacja graficzna porównania wypracowania zawartego w pliku [1-Troja-chatGPT-4.txt] z pozostałymi tekstami. Parametry analizy – *bp*: 60%, *wv*: 6, *gw*: 8

Wynik analizy ujęty w postaci wykresu 4.46 pokazuje podobieństwo pomiędzy plikiem [1-Troja-chatGPT-4.txt] (oś pozioma) z pozostałymi plikami (oś pionowa): [27-Troja-chatGPT-3.5.txt], [29-Troja-chatGPT-3.5.txt], [31-Troja-chatGPT-3.5.txt], [23-Troja-chatGPT-4.txt], [28-Troja-chatGPT-3.5.txt], [11-Troja-chatGPT-4.txt], [8-Troja-chatGPT-4.txt], [26-Troja-chatGPT-3.5.txt], [6-Troja-chatGPT-4.txt], [22-Troja-chatGPT-4.txt], [24-Troja-chatGPT-4.txt], [30-Troja-chatGPT-3.5.txt], [25-Troja-chatGPT-4.txt], [3-Troja-chatGPT-4.txt], [10-Troja-chatGPT-4.txt], [7-Troja-chatGPT-4.txt], [12-Troja-chatGPT-4.txt], [20-Troja-chatGPT-4.txt], [13-Troja-chatGPT-4.txt], [9-Troja-chatGPT-4.txt], [15-Troja-chatGPT-4.txt], [4-Troja-chatGPT-4.txt], [16-Troja-chatGPT-4.txt], [19-Troja-chatGPT-4.txt], [21-Troja-chatGPT-4.txt], [18-Troja-chatGPT-4.txt], [17-Troja-chatGPT-4.txt], [5-Troja-chatGPT-4.txt], [14-Troja-chatGPT-4.txt], [2-Troja-chatGPT-4.txt]. Na wykresie i na podstawie tabeli widać, że pliki są do siebie podobne niewielkimi fragmentami, ale zebrane w całość tworzą wyraźny obraz podobieństwa pomiędzy tekstem zawartym w pliku [1-Troja-chatGPT-4.txt] – analogicznie do przykładu wcześniejszego. Oznacza to, że przedstawiona w pracy metoda analizy macierzowej tekstów bazująca na odelgłości edycyjnej w powyższej konfiguracji analizy (tzn. jeden plik porównywany względem zbioru wielu plików) również w tym przypadku bardzo dobrze sprawdza się w analizie podobieństwa tekstów i dodatkowo pokazuje jedno źródło pochodzenia wypracowań, czyli algorytm generatywnej sztucznej inteligencji[113][43,48]. Wraz z większą liczbą wygenerowanych przez AI odpowiedzi podobieństwo byłoby jeszcze wyższe. Ogólny wynik podobieństwa tekstu zawartego w pliku [1-Troja-chatGPT-4.txt] względem wszystkich zbadanych plików wygenerowanych przez AI to: 46% (59 wyrazów podobnych / 126 wszystkich wyrazów w pliku).

Poniżej znajduje się fragment tekstu z pliku [1-Troja-chatGPT-4.txt] uznany za podobny. Wszystkie terminy uwzględnione w tekście to terminy podobne, występujące w różnych badanych dokumentach.

Film[1] "Troja"[2] z[3] 2004[4] roku[5] w[6] reżyserii[7] Wolfganga[8] Petersena[9] to[10] epicka[11] adaptacja[12] mitu[13] o[14] wojnie[15] trojańskiej,[16] oparta[17] na[18] "Iliadzie"[19] Homera.[20] Film[21] ma[22] swoje[23] mocne[24] strony,[25] takie[26] jak[27] imponujące[28] sceny[29] batalistyczne,[30] bogate[31] kostiumy[32] ... jak[38]Brad[39] Pitt[40] Achilles,[41] Eric[42] Bana[43] Hektor[44] czy[45] Orlando[46] Bloom[47] Parys[48] ...

[113] Generatywna sztuczna inteligencja jest formą sztucznej inteligencji, która może tworzyć tekst, obrazy i zróżnicowane treści w oparciu o dane, na których jest szkolona.

Jednakże[58]film[59] nie[60] jest[61] pozbawiony[62] wad.[63] Niektórzy[64] ... Podsumowując,[91]"Troja"[92] to[93] widowiskowy[94] film,[95] który[96] warto[97] obejrzeć[98] dla[99] ...[126]

ID analizy	Plik (1)	Plik (2)	Fragment tekstu (1)	Fragment tekstu (2)
26	1-Troja-chatGPT-4.txt	28-Troja-chatGPT-3.5.txt	[...] Film "Troja" z 2004 roku w reżyserii Wolfganga Petersena to epicka [...]	[...] film "Troja" z 2004 roku reżyserii Wolfganga Petersena, oparty na epickiej [...]
6	1-Troja-chatGPT-4.txt	30-Troja-chatGPT-3.5.txt	[...] Film "Troja" z 2004 roku w reżyserii Wolfganga Petersena to epicka adaptacja mitu o wojnie trojańskiej, [...]	[...] film "Troja" z 2004 roku reżyserowany przez Wolfganga Petersena jest epicką adaptacją mitu o wojnie trojańskiej. [...]
16	1-Troja-chatGPT-4.txt	25-Troja-chatGPT-4.txt	[...] jak imponujące sceny batalistyczne, bogate kostiumy [...]	[...] jak imponujące sceny batalistyczne, piękne kostiumy [...]
22	1-Troja-chatGPT-4.txt	7-Troja-chatGPT-4.txt	[...] Brad Pitt Achilles, Eric Bana Hektor czy Orlando Bloom [...]	[...] Brad Pitt, Hektor Eric Bana i Parys Orlando Bloom. [...]
15	1-Troja-chatGPT-4.txt	9-Troja-chatGPT-4.txt	[...] Film ma swoje mocne strony, takie jak imponujące sceny [...]	[...] Film ma swoje zalety, jak przepiękne kostiumy, imponujące sceny [...]

Tabela 4.24. Wynik analizy porównania tekstów uznanych za podobne w ramach porównania pliku [1-Troja-chatGPT-4.txt] względem pozostałych plików[114]

Tabela 4.24 zawiera wybrane z wyników fragmenty tekstów uznane za podobne. Fragmenty pochodzą z poszczególnych analiz porównania pliku [1-Troja-chatGPT-4.txt] z pozostałymi. Teksty nie są identyczne, natomiast są podobne, np. wyrazy są odmienione, usunięte lub podmienione przez ich odpowiedniki. Jak widać, również na tym przykładzie zaproponowana w pracy metoda jest wystarczająco skuteczna w wykrywaniu podobieństwa dla tego typu zestawu danych.

TEST 3. Wypracowanie na temat „Internetu Rzeczy”

Test dotyczy analizy wypracowania na temat „Internetu Rzeczy” (ang. Internet of Things, IoT), czyli koncepcji połączenia ze sobą urządzeń codziennego użytku poprzez sieć[115]. Treść prośby do modelu GPT: „Napisz wypracowanie na jedną stronę kartki A4 na temat: "Internet Rzeczy: Przyszłość Technologii i Komunikacji"”. Teksty zostały wygenerowane przez

[114] Pełny raport porównania tekstów wyeksportowany z programu znajduje się pod adresem: https://antyplagius.n-dms.com/tests/chatGPT-Troja/chatGPT-Troja-1.txt.pdf

[115] https://pl.wikipedia.org/wiki/Internet_rzeczy

najnowszy (stan na maj 2024) model GPT 4o („omni")[116] w liczbie 40[117]. Analiza będzie polegała na porównaniu wybranego tekstu z pozostałymi wypracowaniami i zestawieniu ich ze sobą – analogicznie do wcześniejszych testów starszych wersji modelu.

ID analizy	Plik (1)	Plik (2)	WYNIK (1) [%]	WYNIK (2) [%]
53	10-IoT-chatGPT-4o.txt	24-IoT-chatGPT-4o.txt	29,97	28,17
59	10-IoT-chatGPT-4o.txt	29-IoT-chatGPT-4o.txt	20,46	16,55
49	10-IoT-chatGPT-4o.txt	22-IoT-chatGPT-4o.txt	17,58	15,93
46	10-IoT-chatGPT-4o.txt	18-IoT-chatGPT-4o.txt	17,29	17,57
60	10-IoT-chatGPT-4o.txt	30-IoT-chatGPT-4o.txt	16,71	15,65
75	10-IoT-chatGPT-4o.txt	9-IoT-chatGPT-4o.txt	16,71	14,93
57	10-IoT-chatGPT-4o.txt	28-IoT-chatGPT-4o.txt	16,43	13,57
50	10-IoT-chatGPT-4o.txt	21-IoT-chatGPT-4o.txt	16,43	16,29
48	10-IoT-chatGPT-4o.txt	2-IoT-chatGPT-4o.txt	15,27	13,78
38	10-IoT-chatGPT-4o.txt	12-IoT-chatGPT-4o.txt	15,27	12,27
-22	10-IoT-chatGPT-4o.txt	1-IoT-chatGPT-4o.txt	15,56	15,25
55	10-IoT-chatGPT-4o.txt	26-IoT-chatGPT-4o.txt	14,99	15,25
58	10-IoT-chatGPT-4o.txt	3-IoT-chatGPT-4o.txt	14,7	12,56
62	10-IoT-chatGPT-4o.txt	32-IoT-chatGPT-4o.txt	14,7	12,56
51	10-IoT-chatGPT-4o.txt	20-IoT-chatGPT-4o.txt	13,83	13,28
44	10-IoT-chatGPT-4o.txt	14-IoT-chatGPT-4o.txt	13,83	12,01
74	10-IoT-chatGPT-4o.txt	7-IoT-chatGPT-4o.txt	13,83	14,13
47	10-IoT-chatGPT-4o.txt	19-IoT-chatGPT-4o.txt	12,97	11,55
63	10-IoT-chatGPT-4o.txt	33-IoT-chatGPT-4o.txt	12,97	12,96
66	10-IoT-chatGPT-4o.txt	36-IoT-chatGPT-4o.txt	12,68	13,02
69	10-IoT-chatGPT-4o.txt	40-IoT-chatGPT-4o.txt	12,68	10,28
52	10-IoT-chatGPT-4o.txt	23-IoT-chatGPT-4o.txt	12,39	11,2
73	10-IoT-chatGPT-4o.txt	6-IoT-chatGPT-4o.txt	12,39	11,53
39	10-IoT-chatGPT-4o.txt	11-IoT-chatGPT-4o.txt	12,1	11,6
68	10-IoT-chatGPT-4o.txt	4-IoT-chatGPT-4o.txt	12,1	11,76
67	10-IoT-chatGPT-4o.txt	37-IoT-chatGPT-4o.txt	11,82	10,07
72	10-IoT-chatGPT-4o.txt	5-IoT-chatGPT-4o.txt	11,82	10,22
70	10-IoT-chatGPT-4o.txt	38-IoT-chatGPT-4o.txt	11,53	12,86
54	10-IoT-chatGPT-4o.txt	25-IoT-chatGPT-4o.txt	11,24	10,77
40	10-IoT-chatGPT-4o.txt	16-IoT-chatGPT-4o.txt	11,24	9,73
64	10-IoT-chatGPT-4o.txt	34-IoT-chatGPT-4o.txt	10,95	9,45
76	10-IoT-chatGPT-4o.txt	8-IoT-chatGPT-4o.txt	10,95	10
45	10-IoT-chatGPT-4o.txt	17-IoT-chatGPT-4o.txt	10,66	10,25
41	10-IoT-chatGPT-4o.txt	13-IoT-chatGPT-4o.txt	10,37	9,61
32	10-IoT-chatGPT-4o.txt	15-IoT-chatGPT-4o.txt	8,65	6,86
61	10-IoT-chatGPT-4o.txt	31-IoT-chatGPT-4o.txt	8,36	7,11

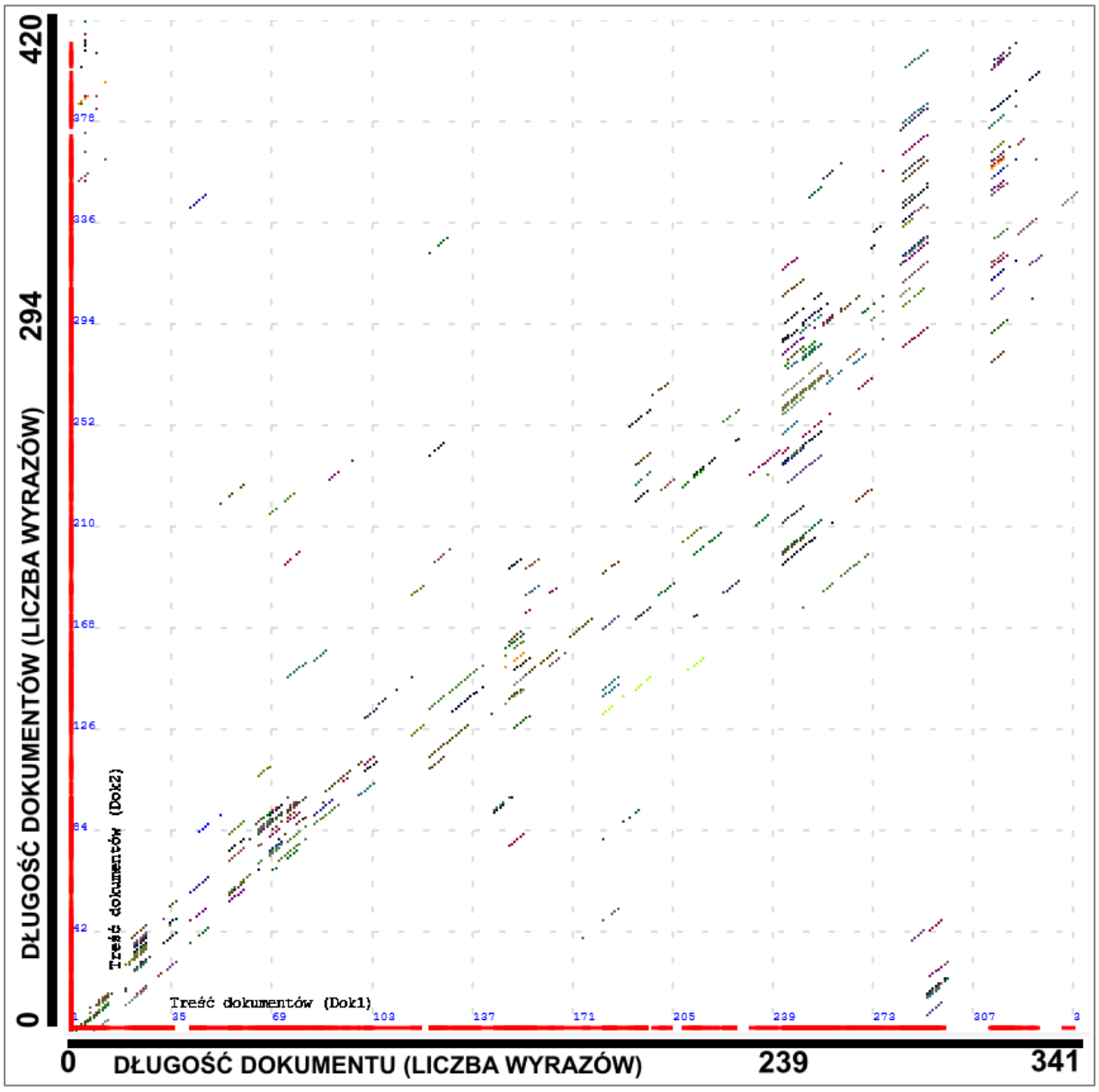
[116] https://openai.com/index/hello-gpt-4o/

[117] Treść tekstów: https://antyplagius.n-dms.com/tests/chatGPT-IoT/

65	10-IoT-chatGPT-4o.txt	35-IoT-chatGPT-4o.txt	6,63	6,13
71	10-IoT-chatGPT-4o.txt	39-IoT-chatGPT-4o.txt	5,76	5,99
56	10-IoT-chatGPT-4o.txt	27-IoT-chatGPT-4o.txt	4,9	4,26

Tabela 4.25. Wyniki analizy porównania wypracowań napisanych przez AI dla podobieństw powyżej 0%

Tabela 4.25 zawiera wyniki analizy podobieństwa pomiędzy plikiem o nazwie [10-IoT-chatGPT-4o.txt] a pozostałymi plikami, których podobieństwa pomiędzy plikiem porównywanym wynoszą ponad 0%. Wszystkich analiz było 780, z czego 39 wykazało minimalne podobieństwo (tab. 4.25).



Rysunek 4.47. Interpretacja graficzna porównania wypracowania zawartego w pliku [10-IoT-chatGPT-4o.txt] z pozostałymi tekstami. Parametry analizy – *bp*: 60%, *wv*: 6, *gw*: 8

Zbiór dokumentów wykazujących podobieństwo do dokumentu [10-IoT-chatGPT-4o.txt], to pliki: [27-IoT-chatGPT-4o.txt], [39-IoT-chatGPT-4o.txt], [35-IoT-chatGPT-4o.txt], [31-IoT-chatGPT-4o.txt], [15-IoT-chatGPT-4o.txt], [13-IoTchatGPT-4o.txt], [17-IoT-chatGPT-4o.txt], [8-IoT-chatGPT-4o.txt], [34-IoT-chatGPT-4o.txt], [16-IoT-chatGPT-4o.txt], [25-IoT-chatGPT-4o.txt], [38-IoT-chatGPT-4o.txt], [5-IoT-chatGPT-4o.txt], [37-IoT-chatGPT-4o.txt], [4-IoT-chatGPT-4o.txt], [11-IoT-chatGPT-4o.txt], [6-IoT-chatGPT-4o.txt], [23-IoT-chatGPT-4o.txt], [40-IoT-chatGPT-4o.txt], [36-IoT-chatGPT-4o.txt], [33-IoT-chatGPT-4o.txt], [19-IoT-chatGPT-4o.txt], [7-IoT-chatGPT-4o.txt], [14-IoT-chatGPT-4o.txt], [20-IoT-chatGPT-4o.txt], [32-IoT-chatGPT-4o.txt], [3-IoT-chatGPT-4o.txt], [26-IoT-chatGPT-4o.txt], [1-IoT-chatGPT-4o.txt], [12-IoT-chatGPT-4o.txt], [2-IoT-chatGPT-4o.txt], [21-IoT-chatGPT-4o.txt], [28-IoT-chatGPT-4o.txt], [9-IoT-chatGPT-4o.txt], [30-IoT-chatGPT-4o.txt], [18-IoT-chatGPT-4o.txt], [22-IoT-chatGPT-4o.txt], [29-IoT-chatGPT-4o.txt], [24-IoT-chatGPT-4o.txt].

Wykres 4.47 wyraźnie pokazuje podobieństwo pomiędzy plikiem [10-IoT-chatGPT-4o.txt] (oś pozioma) z pozostałymi plikami (analogicznie do tabeli 4.25). Również w ramach tego testu widać, że pliki są do siebie podobne niewielkimi fragmentami, ale zebrane w całość tworzą jasny obraz podobieństwa pomiędzy wypracowaniem zawartym w pliku [10-IoT-chatGPT-4o.txt] a pozostałymi. Oznacza to, że przedstawiona metoda podobnie jak w poprzednich analizach sprawdza się w analizie podobieństwa tekstów i również w tym przypadku pokazuje jedno źródło pochodzenia wypracowań, czyli AI w postaci chatGPT. Wraz z większą liczbą wygenerowanych odpowiedzi podobieństwo byłoby jeszcze wyższe.

Ogólny wynik podobieństwa tekstu zawartego w pliku [10-IoT-chatGPT-4o.txt] względem wszystkich zbadanych plików wygenerowanych przez AI wynosi: 90% (314 wyrazów podobnych / 347 wszystkich wyrazów zawartych w pliku). Poniżej znajduje się fragment tekstu z pliku [10-IoT-chatGPT-4o.txt] uznanego za podobny względem zbioru pozostałych dokumentów. Wszystkie terminy uwzględnione w tekście to terminy uznane za podobne występujące w różnych badanych dokumentach. Indeks przy terminie wskazuje na numer kolejności terminu w pliku porównywanym.

###[1] Internet[2] Rzeczy[3] Przyszłość[4] Technologii[5] i[6] Komunikacji[7] Internet[8] Rzeczy[9] IoT[10] to[11] koncept,[12] który[13] rewolucjonizuje[14] współczesny[15] świat,[16] wprowadzając[17] nowy[18] wymiar[19] technologii[20] i[21] komunikacji.[22] IoT[23] odnosi[24] się[25] do[26] sieci[27] urządzeń[28] fizycznych[29] połączonych[30] z[31] Internetem,[32] które[33] zbierają[34] i[35] wymieniają[36] dane.[37] ... od[42]inteligentnych[43] domów[44] po[45] zaawansowane[46] systemy[47] przemysłowe,[48] a[49] jej[50]

potencjał[51] jest[52] niemal[53] nieograniczony.[54] Jednym[55] z[56] kluczowych[57] aspektów[58] IoT[59] jest[60] automatyzacja[61] i[62] zwiększenie[63] efektywności.[64] Inteligentne[65] domy,[66] wyposażone[67] w[68] urządzenia[69] takie[70] jak[71] termostaty,[72] oświetlenie,[73] lodówki[74] czy[75] systemy[76] bezpieczeństwa,[77] mogą[78] być[79] zarządzane[80] zdalnie[81] przez[82] użytkowników[83] za[84] pomocą[85] aplikacji[86] na[87] smartfonach.[88] Dzięki[89] temu[90] możliwe[91] jest[92] oszczędzanie[93] energii,[94] poprawa[95] bezpieczeństwa[96] i[97] podniesienie[98] komfortu[99] życia.[100] Na[101] przykład,[102] inteligentny[103] termostat[104] może[105] dostosować[106] temperaturę[107] w[108] domu[109] na[110] podstawie[111] preferencji[112] mieszkańców[113] i[114] warunków[115] pogodowych,[116] co[117] prowadzi[118] do[119] znacznych[120] oszczędności[121] ... W[123]przemyśle,[124] IoT[125] przyczynia[126] się[127] do[128] tworzenia[129] tzw.[130] Przemysłu[131] 4.[132] 0,[133] gdzie[134] maszyny[135] i[136] systemy[137] produkcyjne[138] są[139] zintegrowane[140] w[141] ramach[142] jednej[143] sieci.[144] Dzięki[145] temu[146] możliwe[147] jest[148] monitorowanie[149] procesów[150] produkcyjnych[151] w[152] czasie[153] rzeczywistym,[154] co[155] zwiększa[156] efektywność,[157] redukuje[158] koszty[159] i[160] minimalizuje[161] ryzyko[162] awarii.[163] Przykładem[164] może[165] być[166] zastosowanie[167] czujników[168] w[169] fabrykach,[170] które[171] monitorują[172] stan[173] maszyn[174] i[175] przewidują[176] potrzebę[177] konserwacji,[178] zanim[179] nastąpi[180] awaria.[181] IoT[182] ma[183] również[184] ogromny[185] wpływ[186] na[187] opiekę[188] zdrowotną.[189] Dzięki[190] urządzeniom[191] noszonym,[192] takim[193] jak[194] smartwatche[195] czy[196] opaski[197] fitness,[198] możliwe[199] jest[200] monitorowanie[201] stanu[202] zdrowia[203] pacjentów[204] na[205] bieżąco.[206] ... mogą[209]śledzić[210] parametry[211] życiowe,[212] takie[213] jak[214] tętno,[215] poziom[216] tlenu[217] we[218] krwi[219] czy[220] aktywność[221] fizyczną,[222] i[223] przesyłać[224] te[225] dane[226] do[227] lekarzy.[228] ... możliwe[232]jest[233] wczesne[234] wykrywanie[235] problemów[236] zdrowotnych[237] i[238] szybka[239] reakcja[240] na[241] nie.[242] Jednakże,[243] rozwój[244] IoT[245] niesie[246] ze[247] sobą[248] również[249] wyzwania,[250] zwłaszcza[251] w[252] zakresie[253] bezpieczeństwa[254] i[255] prywatności.[256] Zwiększona[257] liczba[258] urządzeń[259] połączonych[260] z[261] Internetem[262] oznacza[263] więcej[264] potencjalnych[265] punktów[266] ataku[267] dla[268] cyberprzestępców.[269] Konieczne[270] jest[271] więc[272] opracowanie[273] zaawansowanych[274] mechanizmów[275] zabezpieczających[276] oraz[277] polityk[278] prywatności,[279] aby[280] chronić[281] dane[282] użytkowników.[283] Podsumowując,[284] Internet[285] Rzeczy[286] to[287] przyszłość[288] technologii[289] i[290] komunikacji,[291] która[292] zmienia[293] sposób,[294] w[295] jaki[296] żyjemy[297] i[298] pracujemy.[299] ... korzyści.[313]Jednakże,[314] aby[315] w[316] pełni[317] wykorzystać[318] potencjał[319] IoT,[320] niezbędne[321] jest[322] zwrócenie[323] uwagi[324] na[325] kwestie[326] bezpieczeństwa[327] i[328] prywatności.[329] Przyszłość[330] z[331] ... innowacyjne[338]rozwiązania,[339] które[340] uczynią[341] nasze[342] życie[343] ...[347]

ID analizy	Plik (1)	Plik (2)	Fragment tekstu (1)	Fragment tekstu (2)
65	10-IoT-chatGPT-4o.txt	35-IoT-chatGPT-4o.txt	[...] Jednakże, aby w pełni wykorzystać potencjał IoT, niezbędne jest [...]	[...] Jednak aby w pełni wykorzystać te możliwości, konieczne jest [...]
61	10-IoT-chatGPT-4o.txt	31-IoT-chatGPT-4o.txt	[...] rozwój IoT niesie ze sobą również wyzwania, zwłaszcza w zakresie bezpieczeństwa [...]	[...] rozwój IoT wiąże się także z wyzwaniami, zwłaszcza w zakresie bezpieczeństwa. [...]

32	10-IoT-chatGPT-4o.txt	15-IoT-chatGPT-4o.txt	[...] IoT ma również ogromny wpływ na opiekę zdrowotną. Dzięki urządzeniom noszonym, takim jak smartwatche czy opaski fitness, [...]	[...] IoT ma również ogromny potencjał w sektorze zdrowia. Wearable devices, takie jak smartwatche i opaski fitness, [...]
76	10-IoT-chatGPT-4o.txt	8-IoT-chatGPT-4o.txt	[...] Podsumowując, Internet Rzeczy to przyszłość technologii i komunikacji, [...]	[...] Podsumowując, Internet Rzeczy reprezentuje przyszłość technologii i komunikacji, [...]
54	10-IoT-chatGPT-4o.txt	25-IoT-chatGPT-4o.txt	[...] opracowanie zaawansowanych mechanizmów zabezpieczających oraz polityk prywatności, aby chronić dane użytkowników. Podsumowując, Internet Rzeczy [...]	[...] opracowanie zaawansowanych systemów zabezpieczeń oraz standardów, które zapewnią ochronę prywatności użytkowników. Podsumowując, Internet Rzeczy [...]

Tabela 4.26. Wyniki analizy tekstów uznanych za podobne w ramach porównania pliku [10-IoT-chatGPT-4o.txt] względem pozostałych[118]

Wyniki ujęte w tabeli powyżej (4.26) przedstawiają wybrane fragmenty tekstów z poszczególnych analiz porównania pliku [10-IoT-chatGPT-4o.txt] z pozostałymi plikami. W tabeli widać teksty, które nie są identyczne, ale podobne. Pomimo tego, że metoda nie opiera się na słowniku wyrazów bliskoznacznych i poszczególne wyrazy nie są sprowadzane do ich form podstawowych – to jest wystarczająco skuteczna w wykrywaniu podobieństwa.

4.9. Wnioski

Opracowana przeze mnie metoda macierzowej analizy tekstów bazująca na odległości edycyjnej okazała się skutecznym narzędziem do wykrywania plagiatów oraz nadużyć związanych z generowaniem tekstów przez sztuczną inteligencję. W pracy doktorskiej, zwłaszcza w rozdziale 4 przeprowadziłem liczne testy, które dowiodły, że zastosowanie algorytmu Levenshteina do wypełnienia wartościami logicznymi macierzy porównawczej tekstów i późniejsza jej analiza, umożliwia precyzyjną identyfikację podobieństw między tekstami, zarówno w ramach tego samego języka, jak i w różnych językach należących do tej samej grupy językowej. Jednym z kluczowych osiągnięć mojej metody jest skuteczne wykrywanie plagiatów bez konieczności stosowania algorytmów stemmingu ani lematyzacji. Tradycyjne metody, bazujące na przetwarzaniu morfologicznym tekstu, często napotykają na trudności w analizie języków o złożonej morfologii. Zaproponowana w pracy metoda, bazująca

[118] Pełny raport porównania tekstów wyeksportowany z programu znajduje się pod adresem: https://antyplagius.n-dms.com/tests/chatGPT-IoT/chatGPT-IoT-4o-10.txt.pdf

na prostym porównaniu odległości edycyjnej, unika tych problemów, zachowując wysoką skuteczność nawet w przypadku tekstów napisanych w językach o bogatej fleksji.

Co więcej, wykazałem, że metoda opracowana w ramach mojej pracy jest w stanie skutecznie identyfikować nadużycia polegające na generowaniu tekstów przez systemy sztucznej inteligencji, takie jak ChatGPT (w różnych jego wersjach). Przeprowadzone testy dowiodły, że analiza dużych zbiorów wypracowań umożliwia wykrycie wzorców charakterystycznych dla tekstów generowanych automatycznie, co jest szczególnie istotne w kontekście edukacji i nauki, gdzie oryginalność pracy jest kluczowym kryterium oceny.

Rezultaty mojej pracy wskazują na wysoką skuteczność i uniwersalność opracowanej metody. Może ona znaleźć zastosowanie w różnych dziedzinach, takich jak edukacja, prawo autorskie, a także w szeroko pojętej analizie językowej. Przeprowadzone badania potwierdzają, że moja metoda stanowi wartościowe narzędzie do analizy podobieństwa tekstów, charakteryzujące się prostotą implementacji oraz szerokim zakresem zastosowań.

5. Podsumowanie

Zaproponowana przeze mnie metoda analizy porównawczej tekstów pod kątem podobieństwa, bazująca na algorytmie Levenshteina i wypełnieniu wartościami logicznymi macierzy korelacji dokumentów, okazała się niezwykle skuteczna w różnorodnych zastosowaniach. Testy przeprowadzone w ramach mojej pracy doktorskiej dowiodły, że metoda ta pozwala na precyzyjne wykrywanie plagiatów (nawet bezpośrednio pomiędzy tekstami napisanymi w różnych językach w ramach tej samej grupy językowej) oraz nadużyć związanych z generowaniem tekstów przez systemy sztucznej inteligencji, takie jak ChatGPT.

Szczególnie istotnym osiągnięciem (które nie zostało opisane w niniejszej pracy, gdyż wymaga dopracowania) jest również skuteczność metody w analizie porównawczej tekstów napisanych cyrylicą, które zostały skonwertowane w ramach transkrypcji do alfabetu łacińskiego (przykład analizy w postaci filmu: [119]). Wstępne wyniki badań wskazują, że metoda ta zachowuje wysoką efektywność nawet w przypadku tak złożonych operacji konwersji, co otwiera nowe możliwości jej zastosowania w analizie tekstów w różnych alfabetach. Moje dalsze badania nad udoskonaleniem tej analizy są nadal kontynuowane. Wstępne wyniki

[119] Przykład analizy tekstów napisanych w tramach transkrypcji cyrylicy dostępny jest na stronie projektu: https://www.youtube.com/watch?v=wlEJaIHP8lE

wskazują, że opracowany algorytm sprawdza się również w analizie tekstów napisanych w języku chińskim[120], w jego głównej podstawowej formie. Przyszłe badania będą koncentrowały się na analizie podobieństwa dokumentów napisanych w różnych dialektach języka chińskiego oraz na wykrywaniu podobieństw między nimi.

Metoda sprawdza się również bardzo dobrze w poszukiwaniu plików z zawartością tekstową na dysku na podstawie domniemanego tekstu w nich zawartego. Funkcjonalność ta została zaimplementowana w oprogramowaniu mojego autorstwa - N-DMS Antyplagius, które umożliwia efektywne przeszukiwanie i identyfikowanie tekstów na podstawie ich zawartości, a które opisane zostało w dalszej części rozdziału.

Dalsza praca nad modyfikacją algorytmu będzie obejmowała wykrywanie nadużyć w wypracowaniach napisanych w różnych językach (przeprowadzone analizy miały miejsce w oparciu o język polski) przez różne systemy sztucznej inteligencji, takie jak ChatGPT, GPT-3, BERT[121][46], czy innych znanych produktów bazujących na rozwiązaniach z dziedziny sztucznej inteligencji. Badania będą się skupiały na analizie wzorców charakterystycznych dla tekstów generowanych automatycznie, co pozwoli na jeszcze bardziej precyzyjne identyfikowanie takich nadużyć.

Algorytmy opracowane w ramach mojej pracy doktorskiej zostały poddane szczegółowym testom w rozdziale 4 i wykazały swoją skuteczność zarówno w teoretycznych analizach, jak i w rzeczywistych, praktycznych zastosowaniach. Wyniki te potwierdzają, że metoda ta stanowi wartościowe narzędzie w analizie tekstów, oferując szerokie możliwości w różnych dziedzinach, takich jak edukacja, prawo autorskie oraz analiza językowa.

Jednym z mierzalnych wyników badań opisanych w niniejszej pracy jest aplikacja o nazwie N-DMS Antyplagius - umożliwiająca analizę podobieństwa dokumentów napisanych w różnych alfabetach, w tym łacińskim, cyrylicy i chińskim. Jedną z cech aplikacji jest analiza lokalna dokumentów tekstowych, bez konieczności wysyłania plików na tzw. chmury[122] do bliżej nieokreślonych serwerów w nieokreślonych państwach. W takim przypadku użytkownik programu ma pewność, że dokumenty, które analizuje są bezpieczne, gdyż nie są wysyłane przez sieć. Strona internetowa programu to: https://antyplagius.n-dms.com.

[120] Przykład analizy tekstów napisanych w języku chińskim dostępny jest na stronie projektu: https://www.youtube.com/watch?v=WjPaY6lbwgs

[121] https://arxiv.org/abs/1810.04805

[122] Chmura obliczeniowa (ang. cloud computing) - https://pl.wikipedia.org/wiki/Chmura_obliczeniowa

Rysunek 5.1. Interfejs graficzny aplikacji *Antyplagius* przedstawiający wybrane moduły związane z analizą danych tekstowych

Oprócz porównania dokumentów tekstowych, aplikacja umożliwia również za pomocą metody analizy macierzowej tekstów bazującej na odległości edycyjnej odnajdywanie plików tekstowych na dysku komputera po wpisaniu przez użytkownika fragment tekstu jaki pamięta, a który nie musi być dokładnie taki sam, jak w dokumencie – może być odmieniony przez przypadki, zawierać błędy ortograficzne lub w ogóle nie zawierać części terminów[123], itp. Aplikacja potrafi również wydobywać z tekstu słowa kluczowe bazując na ich podobieństwie ([124],[125]), czy konwertować cyrylicę w ramach transkrypcji[126] i wykonać porównanie (rysunek 5.2, tabela 5.1). Metoda analizuje dokumenty tekstowe bardzo szczegółowo i dokładnie. Dlatego aplikacja Antyplagius została przeze mnie napisana tak, aby wykorzystywać w pełni potencjał obliczeniowy jednostki komputerowej przy jednoczesnym dbaniu o jej stabilność.

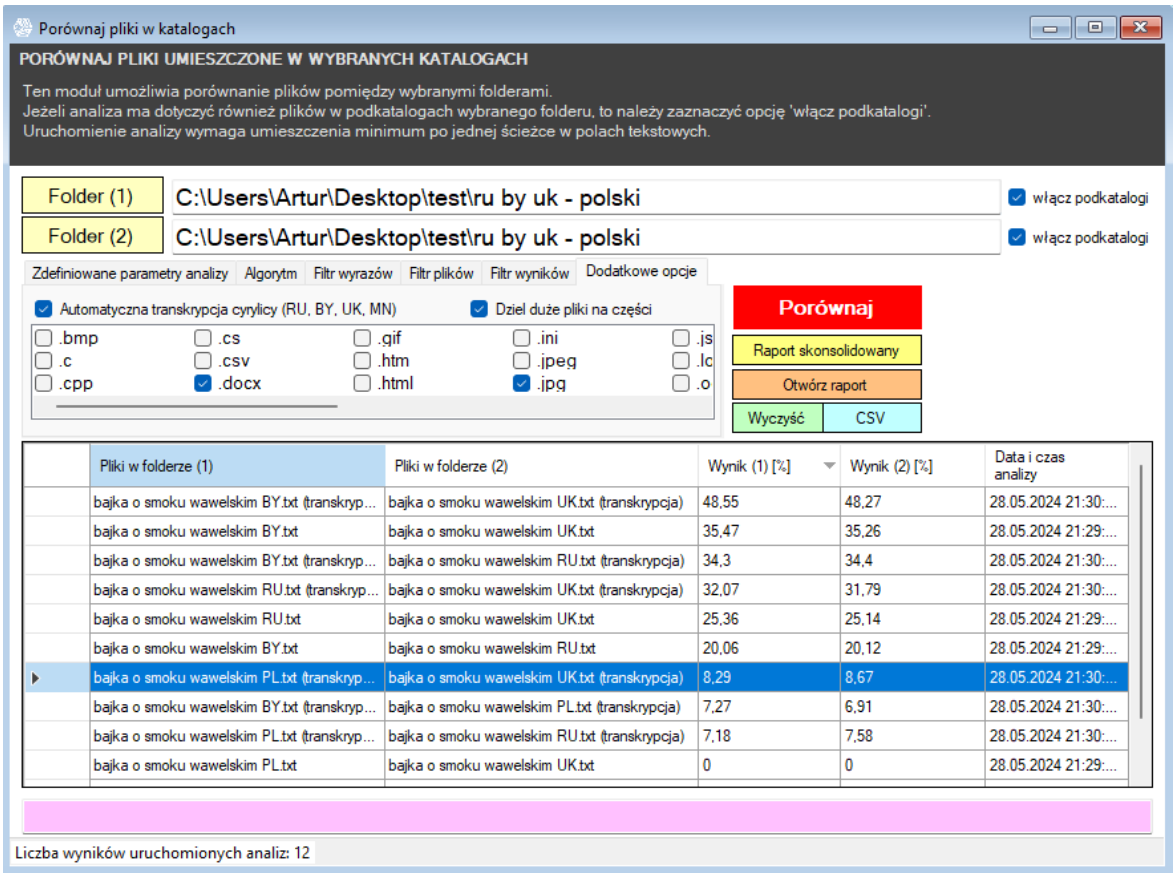
[123] Przykład poszukiwania plików tekstowych, pdf i graficznych po podobnym tekście w nich zawartym na kanale YouTube projektu: https://www.youtube.com/watch?v=QyJ8DUWD8w0

[124] Przykład analizy ekstrakcji słów kluczowych z tekstów napisanych w języku polskim, z wykorzystaniem miary podobieństwa bazującej na odległości edycyjnej dostępny jest na stronie: https://www.youtube.com/watch?v=yW3ZDbK20CQ&list=PLPFeTDhxdQPawnjGhPytgFJeJb-YOXmHC&index=8

[125] Analiza ekstrakcji słów kluczowych z tekstów napisanych w różnych językach (polskim, niemieckim, francuskim i angielskim: https://www.youtube.com/watch?v=Pqt9Sygs32g&list=PLPFeTDhxdQPawnjGhPytgFJeJb-YOXmHC&index=9

[126] Przykład analizy tekstów napisanych w tramach transkrypcji cyrylicy dostępny jest na stronie projektu: https://www.youtube.com/watch?v=wlEJaIHP8lE

Przykładowo program analizuje na bieżąco zajętość pamięci RAM i powstrzymuje uruchamianie nowych wątków dla równoległych analiz wielu plików do momentu jej zwolnienia zgodnie z parametrami ustawionymi przez użytkownika. Zastosowana została tutaj technologia obliczeń równoległych opierająca się o rozwiązania platformy Microsoft .NET. Program dostosowuje również liczbę wątków dzielących macierz korelacji dokumentów w ramach których wykonywane są obliczenia względem liczby procesorów logicznych. Program umożliwia również automatyczne dzielenie bardzo dużych danych tekstowych (nie jest to obligatoryjne), czyli książek na mniejsze fragmenty w celu optymalizacji całości analizy porównawczej, a wynikami są wtedy mniejsze macierze jeżeli podobieństwo istnieje.



Rysunek 5.2. Jeden z modułów interfejsu graficznego aplikacji *Antyplagius* wynik analizy dokumentów tekstowych poddanych transkrypcji. Jeden z podkreślonych wyników pokazuje podobieństwo pomiędzy tekstem polskim a ukraińskim

ID analizy	Plik (1)	Plik (2)	Fragment tekstu (1)	Fragment tekstu (2)
7	bajka o smoku wawelskim PL.txt	bajka o smoku wawelskim UK.txt	[...] polskimi ziemiami rządził król Krak, w Krakowie [...]	[...] polskimi zjemljami panuwaw korol Krak, u Krakowi [...]
7	bajka o smoku wawelskim PL.txt	bajka o smoku wawelskim UK.txt	[...] z nimi równać Jednak Krak był mądrym władcą i [...]	[...] z nimi poriwnjuwati Protje Krak buw mudrim prawitjeljem i [...]
7	bajka o smoku wawelskim PL.txt	bajka o smoku wawelskim UK.txt	[...] jamy. Najciszej jak tylko potrafił zakradł się do samego wejścia, rzucił wypchanego barana i [...]	[...] jami drakona. JAk tilki mig, win pidkrawsja do samogo wchodu, kinuw opudala barana i [...]

Tabela 5.1. Wybrane fragmenty tekstu uznane za podobne w ramach analizy porównawczej przedstawionej na rysunku 5.2

Program Antyplagius umożliwia oprócz analizy tekstu i poszukiwania tekstu w plikach o standardowych rozszerzeniach tekstowych takich jak txt, docx, odt itp. również analizę tekstu zawartego w obrazach (jpg, bmp, png itd.) oraz obrazach w plikach pdf – dzięki zastosowaniu biblioteki OCR[21] (*Tesseract OCR*) bazującej na sieciach neuronowych. Dlatego jak widać, możliwości i zastosowań aplikacji wykorzystujących opisane w pracy rozwiązanie może być więcej. To świadczy o tym, że wraz z połączeniem pozostałych innowacyjnych podejść do analizy danych tekstowych z powyższą techniką, może powstać szereg unikatowych i wartościowych narzędzi, otwierających nowe możliwości dla badań, rozwoju produktów i usprawnienia istniejących procesów w różnych dziedzinach. Mowa tutaj m.in. o:

- systemach antyplagiatowych – głębsze porównanie struktury tekstu i semantyki, przekraczając granice tradycyjnego wyszukiwania zbieżności słów. Może to znacznie poprawić skuteczność wykrywania plagiatu, nawet w przypadku tekstów silnie zmodyfikowanych. Przykładem jest zastosowanie algorytmu do porównywania np. prac naukowych z globalną bazą danych publikacji, wykrywając unikalne frazy oraz strukturalne podobieństwa, które mogą umknąć prostszym systemom opartym na dopasowaniu słów kluczowych,
- oprogramowaniu do stylometrii - przez analizowanie unikalnych wzorców w użyciu słów, składni, a także struktury tekstu, algorytm może pomóc w identyfikacji autorów tekstów, analizie literackiej czy weryfikacji autentyczności dzieł. Może to mieć zastosowanie nie tylko w świecie akademickim, ale również w kryminalistyce,

- algorytmach ksploracji i ekstrakcji informacji - wykorzystując zaawansowane techniki macierzowe, algorytm umożliwia efektywne wydobywanie kluczowych informacji z dużych zbiorów danych, co może być przydatne w automatycznym podsumowywaniu tekstów, analizie sentymentu, czy nawet w budowaniu systemów rekomendacyjnych bazujących na treści. Przykład: wykorzystanie algorytmu do automatycznego podsumowywania artykułów prasowych, wydobywając najważniejsze informacje i kluczowe punkty, ułatwiając szybką analizę dużych zbiorów danych,
- innych koncepcjach i metodach obejmujących zagadnienie text-mining - od identyfikacji trendów w danych po rozpoznawanie wzorców i korelacji w dużych zbiorach tekstowych, algorytm może znaleźć szerokie zastosowanie w przetwarzaniu języka naturalnego, pomagając w rozwiązywaniu złożonych problemów dotyczących interpretacji i analizy danych tekstowych. Przykładem może być implementacja algorytmu w narzędziach do monitorowania mediów społecznościowych, analizujących trendy i nastroje wokół określonych tematów, pomagając firmom w dostosowywaniu strategii marketingowych.

W przedstawionej pracy doktorskiej skupiłem się na problemie analizy danych tekstowych, który stanowi istotne wyzwanie w kontekście dynamicznie rosnącej ilości informacji dostępnej w formie elektronicznej. W szczególności, praca koncentruje się na opracowaniu skutecznych i nieobciążających przesadnie zasobów komputera metod porównawczych tekstów pod kątem ich podobieństwa, co jest kluczowe w wielu dziedzinach. Na początku pracy, przeanalizowałem problematykę analizy danych tekstowych, podkreślając znaczenie tego zagadnienia w dzisiejszym świecie. Następnie, dokonałem przeglądu wybranych metod analizy danych tekstowych stosowanych na przestrzeni lat, począwszy od tradycyjnych technik, aż po najnowsze podejścia wykorzystujące zaawansowane algorytmy komputerowe, w tym te związane z rozwojem AI.

W ramach pracy zaproponowałem koncepcję budowy metod porównawczych bazujących na odległości edycyjnej, która pozwala na mierzenie różnic między tekstami poprzez analizę operacji edycyjnych, takich jak wstawienie, usunięcie czy zamiana znaków. Szczególną uwagę poświęciłem algorytmowi Levenshteina, ponieważ okazał się być skutecznym narzędziem do tego typu analiz.

Głównym punktem pracy było opracowanie nowej metody analizy porównawczej tekstów, bazującej na macierzy i odległości Levenshteina. Metoda ta, poprzez wypełnianie macierzy wartościami logicznymi, a później odpowiednią analizę, umożliwiła precyzyjną identyfikację podobieństw między tekstami bez konieczności stosowania czasochłonnych algorytmów stemmingu i lematyzacji. Zaletą tego podejścia była i jest jego uniwersalność i możliwość zastosowania do różnych języków, w tym nawet do tekstów napisanych cyrylicą, jak również we wspomnianych pod koniec pracy tekstach po transkrypcji cyrylicy do łaciny oraz w języku chińskim (jak również zapewne innych językach, np. greckim, co wymaga dalszych analiz).

W ramach przeprowadzonych badań, metoda ta została przetestowana pod kątem wykrywania plagiatów (w tym tzw. *cross-language*) oraz nadużyć związanych z generowaniem tekstów przez systemy sztucznej inteligencji, takie jak ChatGPT, w tym najnowszy model 4o. Wyniki testów zamieszczonych w postaci danych zawartych w tabelach oraz w postaci graficznej przedstawiającej macierz korelacji dokumentów, podparte dodatkowo nagranymi filmami zamieszczonymi na kanale YouTube, wykazały wysoką skuteczność proponowanej metody zarówno w analizie tekstów napisanych w tych samych językach, jak i w różnych językach należących do tej samej grupy językowej.

W pracy przedstawiłem również dalsze kierunki badań, które obejmują udoskonalanie metody pod kątem analizy tekstów w różnych dialektach języka chińskiego oraz wykrywanie nadużyć w tekstach generowanych przez różne systemy sztucznej inteligencji. W rezultacie, chciałem pokazać, że opracowana metoda stanowi wartościowe narzędzie w analizie tekstów, oferując szerokie możliwości zastosowań w różnych dziedzinach i otwierając nowe perspektywy dla przyszłych badań.

6. Literatura

[1] Richard Friedenthal. *Marcin Luter. Jego życie i czasy.* Czesław Tarnogórski (tłum.). Warszawa: Państwowy Instytut Wydawniczy, 1991, s. 229–231, seria: Biografie Sławnych Ludzi. ISBN 83-06-01897-4.

[2] Susan Hockey. *Guide to Computer Applications in the Humanities.* Gerald Duckworth & Co Ltd, 01.1980. ISBN 978-0715613153 (ISBN-10 0715613154).

[3] Julie Beth Lovins. Development of a Stemming Algorithm. „Mechanical Translation and Computational Linguistics". 11, s. 22–31, 1968.

[4] Porter, Martin F. (1980); *An Algorithm for Suffix Stripping*, Program, 14(3): 130–137

[5] Agnieszka Dziob, Paulina Łazarewicz. *Słowosieć jako narzędzie wspomagające pracę tłumacza*. Rocznik Kognitywistyczny, 2011.

[6] Ewa Rudnicka, Francis Bond, Łukasz Grabowski, Maciej Piasecki, Tadeusz Piotrowski. *Lexical Perspective on Wordnet to Wordnet Mapping*. Proceedings of the 9th Global Wordnet Conference, Singapore, 8-12 January 2018, 2018.

[7] Marek Maziarz, Maciej Piasecki, Ewa Rudnicka. *Słowosieć -- polski wordnet. Proces tworzenia tezaurusa*. Polonica, 2014.

[8] Agnieszka Dziob, Maciej Piasecki. *Implementation of the Verb Model in plWordNet 4.0*. Proceedings of the 9th Global Wordnet Conference, Singapore, 8-12 January 2018, 2018.

[9] В. И. Левенштейн. *Двоичные коды с исправлением выпадений, вставок и замещений символов*. Доклады Академий Наук СССР, 1965. 163.4:845-848.

[10] Niewiarowski A. *Short text similarity algorithm based on the edit distance and thesaurus / Algorytm podobieństwa krótkich fragmentów tekstów oparty na odległości edycyjnej i słowniku wyrazów bliskoznacznych*. Czasopismo Techniczne, Nauki Podstawowe / Technical Transactions / Fundamental Sciences. 1-NP/2016, pp. 159-173. ISSN 2081-2671. Politechnika Krakowska. 2016 r.

[11] Niewiarowski A., Stanuszek M. *Parallelization of the Levenshtein distance algorithm*. Czasopismo Techniczne, Nauki Podstawowe / Technical Transactions / Fundamental Sciences. 3-NP/2014, pp. 109-122. ISSN 2081-2671. Politechnika Krakowska. 2014 r.

[12] Niewiarowski A., Stanuszek M. *Mechanizm identyfikacji i klasyfikacji treści / The mechanism of identification and classification of content*. Studia Informatica. Volume 34, Number 2B (112), s. 205-222, ISSN PL 0208-7286. Silesian University of Technology Press. Gliwice 2013.

[13] Niewiarowski A., Stanuszek M. *Mechanizm analizy krótkich fragmentów tekstów na bazie odległości Levenshteina / Mechanism of analysis of similarity short texts, based on the Levenshtein distance.* Studia Informatica. Volume 34, Number 1 (110), s. 107-114, ISSN PL 0208-7286. Silesian University of Technology Press. Gliwice 2013.

[14] Niewiarowski A. *Optymalizacja schematu ważenia terminów dla modelu wektorowego / Term frequency optimization for the vector space model*. Czasopismo

Techniczne Technical Transactions, s. 155-165, ISSN 0011-4561 ISSN 1897-6328. Politechnika Krakowska. 2013 r.

[15] Paice, C.D. *Another stemmer*, SIGIR Forum, 24(3), 56-61, 1990.

[16] Krovetz, R. *Viewing morphology as an inference process*. InProceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 191-202. ACM, 1993.

[17] Dawson, J. *Suffix removal and word conflation*. ALLC Bulletin, 2(3), 33-46, 1974.

[18] Ismailov A., Abdul Jalil M.M., Abdullah Z., Abd Rahim N.H. *A Comparative Study of Stemming Algorithms for use with the Uzbek Language*. Conference: ICCOINS2016, ISBN 978-1-5090-2549-7, 2016.

[19] Sandeep R. Sirsat, Vinay Chavan, Hemant S. Mahalle. *Strength and Accuracy Analysis of Affix Removal Stemming Algorithms*. International Journal of Computer Science and Information Technologies, Vol. 4 (2) , pp. 265 – 269, 2013.

[20] Wahiba Ben Abdessalem Karaa. *A New Stemmer to Improve Information Retrieval*. International Journal of Network Security & Its Applications (IJNSA), Vol.5, No.4, pp. 143-154, July 2013.

[21] Hockey S*. OCR: The Kurzweil Data Entry Machine*. Literary and Linguistic Computing, 1 (1986), 63-67.

[22] Hockey S*. A Survey of Practical Aspects of Computer-Aided Maintenance and Processing of Natural Language Data* in Computational Linguistics. Ein internationales Handbuch zur computergestutzten Sprachforschung und ihrer Anwendung. Berlin, 1989, p. 752-759.

[23] Piasecki M., Walkowiak T., Eder M. *Open Stylometric System WebSty: Integrated Language Processing, Analysis and Visualisation*. CMST, Vol. 24 (1) 2018, 43-58

[24] Eder, M., Piasecki, M., & Walkowiak, T. *An open stylometric system based on multilevel text analysis*. Cognitive Studies | Études cognitives, 2017(17)

[25] Walkowiak, T. *Language Processing Modelling Notation – Orchestration of NLP Microservices.* Advances in Dependability Engineering of Complex Systems: Proceedings of the Twelfth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, 2017, Springer International Publishing, pp. 464-473

[26] Weiss D.: *A Survey of Freely Available Polish Stemmers and Evaluation of Their Applicability in Information Retrieval*. 2nd Language and Technology Conference, Poznań, Poland, 2005, pp. 216-221.

[27] Weiss D.: *Stempelator: A Hybrid Stemmer for the Polish Language*. Institute of Computing Science, Poznań University of Technology, Poland, Research Report RA-002/05, 2005.

[28] Broda B., Piasecki M. *SuperMatrix: a General Tool for Lexical Semantic Knowledge Acquisition*. In Speech and Language Technology, 239-254. Polish Phonetics Assocation, 2008.

[29] Broda B., Piasecki M. *Parallel, Massive Processing in SuperMatrix -- a General Tool for Distributional Semantic Analysis of Corpora.* International Journal of Data Mining, Modelling and Management, 2011.

[30] Mendenhall, T.C. *The characteristic curves of composition*. Science, 9, 237-249, 1887.

[31] Mendenhall, T.C. *A mechanical solution of a literary problem*. The Popular Science Monthly, 60, 97-105, 1901.

[32] Williams, C.B. *Mendenhall's studies of word-length distribution in the works of Shakespeare and Bacon*. Biometrika, 62, 207-212, 1975.

[33] Lutosławski W. Principes de stylométrie appliqués à la chronologie des œuvres de Platon, Revue des Études Grecques Année, 11-41 pp. 61-81, 1898.

[34] Katzner Kenneth. The Languages of the World, Taylor & Francis Ltd, 2002. ISBN-13: 9780415250047

[35] Wu Y., Schuster M., Chen Z., Le Q., Norouzi M. *Google's Neural Machine Translation System: Bridging the Gapbetween Human and Machine Translation*. arXiv:1609.08144v2 [cs.CL] 8 Oct 2016.

[36] Komorovskaya Viktoria. *The future of the Belarusian language: is it doomed to extinction? Controversies and challenges in the language maintenance and revitalization*. Issue: Acta Philologica. Uniwersytet Warszawski. 2016; 48 : 15-28.

[37] Bakhteev, O. et al. (2022). *Cross-Language Plagiarism Detection: A Case Study of European Languages Academic Works*. In: Bjelobaba, S., Foltýnek, T., Glendinning, I., Krásničan, V., Dlabolová, D.H. (eds) Academic Integrity: Broadening Practices, Technologies, and the Role of Students. Ethics and Integrity in Educational Contexts, vol 4. Springer, Cham.

[38] Agarwal, B. (2019). *Cross-lingual plagiarism detection techniques for English-Hindi language pairs*. Journal of Discrete Mathematical Sciences and Cryptography, 22(4), 679–686.

[39] NIEWIAROWSKI, Artur; PLICHTA, Anna. *Matrix similarity analysis of texts written in Romanian and Spanish*. ECMS 2023 : proceedings of the 37th ECMS International Conference on Modelling and Simulation, June 20th – June 23rd, 2023 Florence, Italy. Vol. 37, Iss. 1, p. 507-512. ISSN: 2522-2422. ISBN: 978-3-937436-80-7.

[40] Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean. *Efficient Estimation of Word Representations in Vector Space*. arXiv:1301.3781v3 [cs.CL] 7 Sep 2013.

[41] Jeffrey Pennington, Richard Socher, and Christopher Manning. *GloVe: Global Vectors for Word Representation*. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1532–1543, Doha, Qatar. Association for Computational Linguistics. 2014.

[42] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.* arXiv:1810.04805v2 [cs.CL] 24 May 2019.

[43] Radford, Alec; Narasimhan, Karthik; Salimans, Tim; Sutskever, Ilya. *"Improving Language Understanding by Generative Pre-Training"*. OpenAI. June 11, 2018.

[44] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah. *Language Models are Few-Shot Learners*. arXiv:2005.14165v4 [cs.CL] 22 Jul 2020.

[45] OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman. GPT-4 Technical Report. arXiv:2303.08774v6 [cs.CL] 4 Mar 2024.

[46] Devlin Jacob, Chang Ming-Wei, Lee Kenton, Toutanova Kristina. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. arXiv:1810.04805v2 [cs.CL] 24 May 2019.

[47] Niewiarowski A. *Similarity detection based on document matrix model and edit distance algorithm*. Computer Assisted Methods in Engineering and Science, Vol. 26, No. 3-4, pp. 163-175. ISSN: 2299-3649. 2019.

[48] Korzynski P., Mazurek G., Altmann A., Ejdys J., Kazlauskaite R., Paliszkiewicz J., Wach K., Ziemba E. *Generative artificial intelligence as a new context for management theories: analysis of ChatGPT*. Central European Management Journal, pp. 3-13, ISSN: 2658-0845, 2023.

[49] Blei, D. M., Ng, A. Y., & Jordan, M. I. *Latent Dirichlet Allocation*. Journal of Machine Learning Research, 3, 993-1022. 2003.

[50] Cortes, C., & Vapnik, V. *Support-vector networks*. Machine Learning, 20(3), 273-297. 1995.

[51] Vikramkumar, Vijaykumar B., & Trilochan. Bayes and Naive Bayes Classifier. 2014. arXiv: https://ar5iv.org/abs/1404.0933

[52] Boyer, R. S., & Moore, J. S. *A fast string searching algorithm.* Communications of the ACM, 20(10), 762-772. 1997.

[53] Knuth, D. E., Morris, J. H., & Pratt, V. R. *Fast pattern matching in strings*. SIAM Journal on Computing, 6(2), 323-350. 1977.

[54] Karp, R. M., & Rabin, M. O. *Efficient randomized pattern-matching algorithms*. IBM Journal of Research and Development, 31(2), 249-260. 1987.

[55] Vieira, R., Cabral, L., Lima, L., & Santoro, F. *Mathematical properties of soft cardinality: Enhancing Jaccard, Dice, and Cosine similarity coefficients*. Information Sciences, 418-419, 1-20. 2017.

[56] McCallum, Andrew; Nigam, Kamal. *A comparison of event models for Naive Bayes text classification*. AAAI-98 workshop on learning for text categorization. Vol. 752. 1998.

[57] Sohangir, S., & Wang, D. *Improved sqrt-cosine similarity measurement*. Journal of Big Data, 4(1), 25. 2017.

7. Załączniki

7.1. Fragmenty tekstu napisanego w językach: hiszpańskim (a) i portugalskim (b)

a) Pero, a pesar de esta variedad de posibilidades que la voz posee, sería muy pobre instrumento de comunicación si no contara más que con ella. La capacidad de expresión del hombre no dispondría de más medios que la de los animales. La voz, sola, es para el hombre apenas una materia informe, que para convertirse en un instrumento perfecto de comunicación debe ser sometida a un cierto tratamiento. Esa manipulación que recibe la voz son las 'articulaciones'.

b) Porém, apesar de esta variedade de possibilidades que a voz possui, seria muito pobre instrumento de comunicação se não contasse com mais além dela. A capacidade de expressão do homem não disporia de mais meios que a dos animais. A voz, sozinha, é para o homem apenas uma matéria informe, que para converter-se num instrumento perfeito de comunicação deve ser submetida a um certo tratamento. Essa manipulação que recebe a voz são as 'articulações'.

7.2. Fragment tekstu: Opowieść Wigilijna w języku angielskim

a) Marley was dead: to begin with. There is no doubt whatever about that. The register of his burial was signed by the clergyman, the clerk, the undertaker, and the chief mourner. Scrooge signed it: and Scrooge's name was good upon 'Change, for anything he chose to put his hand to. Old Marley was as dead as a door-nail.

b) Mind! I don't mean to say that I know, of my own knowledge, what there is particularly dead about a door-nail. I might have been inclined, myself, to regard a coffin-nail as the deadest piece of ironmongery in the trade. But the wisdom of our ancestors is in the simile; and my unhallowed hands shall not disturb it, or the Country's done for. You will therefore permit me to repeat, emphatically, that Marley was as dead as a door-nail.

c) Scrooge knew he was dead? Of course he did. How could it be otherwise? Scrooge and he were partners for I don't know how many years. Scrooge was his sole executor, his sole administrator, his sole assign, his sole residuary legatee, his sole friend and sole mourner. And even Scrooge was not so dreadfully cut up by the sad event, but that he was an excellent man of business on the very day of the funeral, and solemnised it with an undoubted bargain.

d) The mention of Marley's funeral brings me back to the point I started from. There is no doubt that Marley was dead. This must be distinctly understood, or nothing wonderful can come of the story I am going to relate. If we were not perfectly convinced that Hamlet's Father died before the play began, there would be nothing more remarkable in his taking a stroll at night, in an easterly wind, upon his own ramparts, than there would be in any other middle-aged gentleman rashly turning out

after dark in a breezy spot -- say Saint Paul's Churchyard for instance -- literally to astonish his son's weak mind.

7.3. Fragment tekstu książki pt. „W pustyni i w puszczy” Henryka Sienkiewicza (a) oraz jego modyfikacja (b)

a) Ale wakacje jego zaczynały się dopiero za kilka dni, ponieważ zaś rewizja robót przy kanałach była sprawą pilną i starsi panowie nie mogli tracić czasu, ułożyli się przeto, że wyjadą niezwłocznie, a dzieci wraz z panią Olivier w tydzień później. I Nel, i Staś mieli ochotę jechać zaraz, ale Staś nie śmiał o to prosić.

b) ~~**[Ale]**~~ Jego wakacje **zaczynaly** się dopiero za kilka dni~~**[,]**~~ **poniewaz** zaś rewizja **robut** przy **kanałah jest** sprawą pilną i ~~**[starsi]**~~ panowie nie mogli **stracić** czasu **ułorzyli** się przeto, że wyjadą **nie zwłocznie** ~~**[, a]**~~ dzieci wraz z ~~**[panią]**~~ Olivier w **tydzien potem,** i **nel** i **staś** mieli **ochote jechac** zaraz~~**[,]**~~ ale *nie śmiał Staś* o to **poprosić**.

7.4. Zbiór wypracowań na temat języka programowania JavaScript (chatGPT)[127]

Plik	Zawartość tekstowa
[1.txt]	JavaScript to wysokopoziomowy, dynamiczny, obiektowy język programowania, który jest powszechnie stosowany do tworzenia interaktywnych stron internetowych. Początkowo został stworzony przez Brendana Eicha w 1995 roku i został zaimplementowany jako część przeglądarki internetowej Netscape Navigator. Dzisiaj jest jednym z podstawowych elementów tworzenia stron WWW, obok HTML (języka do definiowania struktury strony) i CSS (języka do stylizacji strony). JavaScript jest językiem o szerokim zastosowaniu, który pozwala na dodawanie interakcji, animacji, obsługi zdarzeń, walidacji danych, manipulację elementami na stronie i wiele innych funkcji. Jest obsługiwany przez większość współczesnych przeglądarek internetowych i jest nieodzownym elementem nowoczesnego rozwoju stron internetowych. W ostatnich latach, dzięki rozwojowi środowisk takich jak Node.js, JavaScript zyskał również popularność jako język stosowany po stronie serwera. To umożliwia tworzenie pełnoprawnych aplikacji internetowych opartych wyłącznie na JavaScript, zarówno po stronie klienta, jak i serwera.
[2.txt]	JavaScript to wysokopoziomowy, obiektowy język programowania, którego głównym celem jest umożliwienie twórcy stron internetowych manipulowania treścią i zachowaniem strony w przeglądarce internetowej. JavaScript został stworzony w 1995 roku przez Brendana Eicha, pracującego wówczas w firmie Netscape Communications. Początkowo nosił nazwę "Mocha", a następnie "LiveScript", zanim ostatecznie zmieniono jego nazwę na "JavaScript". JavaScript jest jednym z trzech głównych filarów technologii internetowych, obok HTML (Hypertext Markup Language) i CSS

[127] Zawartość plików dostępna jest pod adresem: https://antyplagius.n-dms.com/tests/chatGPT-JavaScript

	(Cascading Style Sheets). HTML jest odpowiedzialny za strukturę i treść strony, CSS definiuje jej styl i wygląd, a JavaScript pozwala na interaktywność i dynamiczne zachowanie strony. Współcześnie JavaScript jest wykorzystywany nie tylko w przeglądarkach internetowych, ale także poza nimi, np. w technologii Node.js do tworzenia serwerów i aplikacji internetowych, w narzędziach takich jak Electron do tworzenia aplikacji desktopowych, czy w środowiskach takich jak React Native do tworzenia aplikacji mobilnych. JavaScript jest językiem programowania opartym na prototypach, co oznacza, że dziedziczenie odbywa się poprzez tzw. łańcuchy prototypów. Wspiera również różne paradygmaty programowania, takie jak programowanie imperatywne, obiektowe (w oparciu o prototypy) oraz funkcyjne. W 2015 roku został wprowadzony standard ECMAScript 6 (znany również jako ES2015), który przyniósł wiele ulepszeń i nowych funkcji, takich jak klasy, moduły czy strzałkowe funkcje lambda. Od tego czasu standard ECMAScript jest rozwijany i uaktualniany corocznie.
[3.txt]	JavaScript to popularny język programowania wysokiego poziomu, którego głównym zastosowaniem jest rozwój interaktywnych stron internetowych. Jest to język skryptowy, co oznacza, że jest interpretowany, a nie kompilowany. Początkowo został stworzony przez Brendana Eicha w 1995 roku w firmie Netscape Communications, jako narzędzie do dodawania dynamizmu i interakcji do stron internetowych. JavaScript jest językiem o słabym typowaniu, co oznacza, że nie musisz deklarować typów danych, które będą używane w zmiennych. Obsługuje wiele paradygmatów programowania, takich jak programowanie obiektowe, imperatywne oraz funkcyjne. Współcześnie JavaScript jest podstawowym językiem używanym w przeglądarkach internetowych, ale jego zastosowanie się rozszerzyło również poza przeglądarkę, np. dzięki środowisku Node.js, które pozwala na uruchamianie JavaScript na serwerach. Jest też używany w różnych technologiach, takich jak Electron do tworzenia aplikacji desktopowych, czy React Native do tworzenia aplikacji mobilnych. Wraz z HTML i CSS, JavaScript jest jednym z trzech podstawowych języków wykorzystywanych do tworzenia i projektowania stron internetowych.
[4.txt]	JavaScript to wysokopoziomowy, interpretowany język programowania, który został stworzony przez Brendana Eicha w 1995 roku, a następnie został rozwinięty przez firmę Netscape i standardy ECMA. JavaScript jest głównie używany do tworzenia i manipulowania zawartością na stronach internetowych, dzięki czemu stają się one interaktywne i dynamiczne. Jest jednym z trzech podstawowych języków używanych w tworzeniu stron internetowych, obok HTML (HyperText Markup Language) i CSS (Cascading Style Sheets). JavaScript to język skryptowy, co oznacza, że jest interpretowany w czasie rzeczywistym przez przeglądarkę internetową, a nie kompilowany do kodu maszynowego przed uruchomieniem. Pozwala to na szybkie tworzenie i testowanie kodu. Jest językiem wieloparadygmatowym, co oznacza, że może

	być stosowany w różnych stylach programowania, takich jak imperatywnym, obiektowym i funkcyjnym. W ostatnich latach popularność JavaScriptu wzrosła również poza przeglądarki, dzięki platformie Node.js, która umożliwia tworzenie serwerów i narzędzi opartych na tym języku. Dzięki temu JavaScript jest obecnie używany zarówno na froncie (strona klienta), jak i na backendzie (strona serwera) w architekturze aplikacji webowych.
[5.txt]	JavaScript to wysokopoziomowy, dynamiczny język programowania, który jest powszechnie stosowany w tworzeniu stron internetowych i aplikacji webowych. Został stworzony w 1995 roku przez Brendana Eicha, który pracował wówczas dla firmy Netscape. JavaScript jest szczególnie znany z tego, że pozwala na tworzenie interaktywnych elementów na stronach internetowych, takich jak animacje, formularze, przyciski czy dynamiczne ładowanie treści. JavaScript jest językiem programowania obsługiwanym przez większość przeglądarek internetowych, dzięki czemu można go używać do manipulowania elementami strony, reagowania na zdarzenia (takie jak kliknięcia czy ruchy myszą) oraz komunikowania się z serwerem za pomocą technologii takich jak AJAX (Asynchronous JavaScript and XML). W ostatnich latach JavaScript znacznie ewoluował, a także zyskał na popularności w rozwoju oprogramowania poza przeglądarkami, zwłaszcza w aplikacjach serwerowych za pomocą platformy Node.js. JavaScript jest także często wykorzystywany w innych środowiskach, takich jak aplikacje mobilne, czy narzędzia do tworzenia gier.
[6.txt]	JavaScript to wysokopoziomowy, dynamiczny, interpretowany język programowania, który został stworzony w latach 90. XX wieku przez Brendana Eicha, pracującego wówczas w firmie Netscape. Język ten został pierwotnie opracowany jako język skryptowy do obsługi stron internetowych, ale z czasem zyskał większą funkcjonalność i popularność, co pozwoliło na jego wykorzystanie również poza przeglądarkami internetowymi. JavaScript jest obecnie powszechnie używany do tworzenia aplikacji internetowych, zarówno na stronie klienta (front-end), jak i serwera (back-end), za pomocą środowisk takich jak Node.js. Język ten pozwala na tworzenie dynamicznych i interaktywnych stron internetowych, obsługujących zdarzenia, animacje czy komunikację z serwerem. JavaScript należy do rodziny języków programowania opartych na prototypach i stosuje paradygmaty takie jak imperatywny, obiektowy oraz funkcyjny. Wspiera zarówno programowanie asynchroniczne, jak i synchroniczne. W przeciwieństwie do języków kompilowanych, takich jak C++ czy Java, JavaScript jest interpretowany, co oznacza, że jego kod źródłowy jest wykonywany bezpośrednio przez przeglądarkę lub inne środowisko uruchomieniowe. Język JavaScript nie należy mylić z Java, gdyż są to dwa odrębne języki programowania o różnych cechach i zastosowaniach.
[7.txt]	JavaScript to wysokopoziomowy, interpretowany język programowania, który został opracowany przez Brendana Eicha w 1995 roku, pierwotnie pod nazwą Mocha, a następnie

	LiveScript. JavaScript stał się jednym z kluczowych języków używanych w internecie, służąc głównie do tworzenia dynamicznych i interaktywnych stron internetowych. Język JavaScript jest obiektowy, oparty na prototypach, co oznacza, że obiekty mogą dziedziczyć funkcje i właściwości od innych obiektów. JavaScript wspiera różne paradygmaty programowania, takie jak imperatywny, obiektowy i funkcyjny. Mimo że nazwa JavaScript może sugerować związek z językiem Java, są to dwa odrębne języki programowania. Jedynym wspólnym elementem jest składnia, która w przypadku JavaScript została zainspirowana językiem Java, ale też wpływami języków takich jak C, C++ i Python. JavaScript jest powszechnie używany w przeglądarkach internetowych, gdzie jest jednym z trzech głównych technologii, obok HTML (Hypertext Markup Language) i CSS (Cascading Style Sheets), które stanowią podstawę tworzenia stron internetowych. Wraz z pojawieniem się technologii Node.js, JavaScript zaczął być również stosowany po stronie serwera, co rozszerzyło jego zastosowanie na cały stos technologiczny - zarówno frontend, jak i backend.
[8.txt]	avaScript to wysokopoziomowy, interpretowany język programowania, który jest głównie używany do tworzenia interaktywnych stron internetowych. Został stworzony w 1995 roku przez Brendana Eicha, pracującego wówczas dla firmy Netscape Communications. JavaScript jest obecnie jednym z najpopularniejszych języków programowania na świecie. JavaScript jest językiem programowania, który działa po stronie klienta, co oznacza, że jest uruchamiany w przeglądarce internetowej użytkownika. Dzięki temu możliwe jest tworzenie animacji, obsługa zdarzeń, manipulacja elementami strony (np. zmiana tekstu czy ukrywanie elementów) oraz komunikacja z serwerem bez konieczności przeładowania strony. W ostatnich latach, rola JavaScript w świecie programowania znacznie się rozszerzyła. Obecnie jest wykorzystywany również po stronie serwera, dzięki platformie Node.js, co pozwala na tworzenie kompletnych aplikacji webowych korzystając tylko z jednego języka programowania. JavaScript jest językiem dynamicznym, co oznacza, że jego składnia jest elastyczna, co sprawia, że łatwo można go nauczyć i zastosować w praktyce. Wspiera różne paradygmaty programowania, takie jak programowanie obiektowe, imperatywne i funkcyjne.
[9.txt]	JavaScript to wysokopoziomowy, interpretowany język programowania, który został stworzony przez Brendana Eicha i wprowadzony przez Netscape Communications w 1995 roku. Język ten stał się podstawowym składnikiem technologii webowych, obok HTML i CSS, i jest powszechnie stosowany w przeglądarkach internetowych na całym świecie. JavaScript pozwala na tworzenie interaktywnych stron internetowych, obsługę zdarzeń, animacje oraz manipulację elementami strony bez konieczności przeładowania strony. JavaScript jest językiem o dynamicznym typowaniu, co oznacza, że programiści nie muszą deklarować typów zmiennych. Język

	ten jest również wieloparadygmatowy, co oznacza, że może być używany w różnych stylach programowania, takich jak programowanie obiektowe, funkcyjne czy imperatywne. W ostatnich latach JavaScript znacznie ewoluował, a jego zastosowanie rozszerzyło się poza przeglądarki. Dzięki platformie Node.js można używać JavaScript także po stronie serwera, co pozwala na tworzenie szybkich i skalowalnych aplikacji internetowych. Ponadto JavaScript jest stosowany w różnych innych środowiskach, takich jak aplikacje mobilne (React Native), desktopowe (Electron) czy nawet gry komputerowe (Three.js, Phaser). Rozwój języka JavaScript jest nadzorowany przez konsorcjum ECMA International, które regularnie publikuje nowe wersje standardu ECMAScript - oficjalnej specyfikacji języka JavaScript.
[10.txt]	JavaScript to popularny, obiektowy język programowania, używany głównie do tworzenia interaktywnych stron internetowych i aplikacji webowych. Pierwotnie został opracowany przez Brendana Eicha w firmie Netscape w 1995 roku. JavaScript umożliwia tworzenie dynamicznych treści, takich jak animacje, przetwarzanie zdarzeń (np. kliknięcia czy ruchy myszką), walidację formularzy oraz obsługę asynchronicznych żądań, dzięki czemu strony internetowe stają się bardziej responsywne i użyteczne. JavaScript działa po stronie klienta, co oznacza, że jest wykonywany przez przeglądarkę internetową na komputerze użytkownika, nie wymagając serwera do przetwarzania kodu. Dzięki temu można zmniejszyć obciążenie serwera i opóźnienia w komunikacji. JavaScript jest jednym z trzech podstawowych technologii używanych do tworzenia stron internetowych, obok HTML (Hypertext Markup Language) do definiowania struktury strony oraz CSS (Cascading Style Sheets) do opisu jej wyglądu. W ciągu ostatnich lat JavaScript ewoluował, a jego zastosowania rozszerzyły się również poza przeglądarki. Dzięki środowiskom takim jak Node.js, JavaScript może być używany również po stronie serwera, umożliwiając tworzenie skalowalnych aplikacji internetowych i obsługę baz danych.
[11.txt]	JavaScript to wysokopoziomowy, dynamiczny język programowania, który jest często używany w kontekście tworzenia stron internetowych i aplikacji webowych. Początkowo został stworzony przez Brendana Eicha w 1995 roku dla firmy Netscape i pierwotnie nazywał się LiveScript. Później nazwa została zmieniona na JavaScript. JavaScript jest szczególnie znany jako język skryptowy dla stron internetowych, co oznacza, że jest używany do dodawania interaktywnych funkcji do stron, takich jak animacje, przyciski, formularze czy dynamiczne ładowanie treści. Działa on w przeglądarkach internetowych, gdzie jest interpretowany i wykonany przez silnik JavaScript wbudowany w przeglądarkę. Chociaż JavaScript początkowo był używany głównie na stronach internetowych, ewoluował w ciągu lat i teraz jest używany również w innych kontekstach. Na przykład Node.js to środowisko, które pozwala na używanie JavaScript do tworzenia serwerów i aplikacji po stronie serwera. JavaScript może być

	również używany do tworzenia aplikacji na urządzenia mobilne, a także w innych technologiach, takich jak Internet Rzeczy (IoT). Warto również wspomnieć, że JavaScript jest językiem o luźnym typowaniu, co oznacza, że nie jest wymagane deklarowanie typów zmiennych przed ich użyciem. Jest także obiektowy, co oznacza, że struktury danych i funkcje są reprezentowane jako obiekty. Ponadto, JavaScript obsługuje zarówno programowanie imperatywne, jak i funkcyjne, co czyni go elastycznym językiem.
[12.txt]	JavaScript to wysokopoziomowy, interpretowany język programowania, który jest używany głównie do tworzenia interaktywnych i dynamicznych stron internetowych. Został stworzony przez Brendana Eicha w 1995 roku i pierwotnie nazywał się Mocha, a następnie LiveScript, zanim ostatecznie został nazwany JavaScript. JavaScript jest językiem obiektowym, który pozwala na tworzenie złożonych aplikacji webowych. Jest powszechnie stosowany w połączeniu z HTML i CSS, aby zapewnić interaktywność i bogate doświadczenia użytkownika. Działa na stronie klienta, co oznacza, że jest uruchamiany w przeglądarce internetowej użytkownika, co pozwala na szybkie i płynne działanie aplikacji bez konieczności przekazywania danych na serwer. W ostatnich latach JavaScript zyskał również popularność jako język programowania po stronie serwera, dzięki technologii Node.js. Dzięki Node.js, JavaScript może być używany do tworzenia kompletnych aplikacji webowych, zarówno na froncie, jak i backendzie. Współcześnie, JavaScript stał się jednym z najbardziej popularnych i szeroko stosowanych języków programowania na świecie. Wprowadzono również wiele bibliotek i frameworków, takich jak React, Angular i Vue.js, które ułatwiają i przyspieszają rozwój aplikacji opartych na JavaScript.
[13.txt]	JavaScript to wysokopoziomowy, dynamiczny język programowania, który jest głównie używany w kontekście tworzenia stron internetowych. Został stworzony przez Brendana Eicha w 1995 roku i pierwotnie nazywał się Mocha, a następnie LiveScript. Nazwa "JavaScript" została przyjęta później, gdy język stał się częścią standardu ECMAScript. JavaScript jest językiem skryptowym, co oznacza, że jest interpretowany przez przeglądarkę internetową, a nie kompilowany przed uruchomieniem. Umożliwia to tworzenie interaktywnych i dynamicznych stron WWW, które reagują na działania użytkownika, takie jak kliknięcia czy przesunięcia. Chociaż początkowo JavaScript był używany głównie w przeglądarkach internetowych, z biegiem czasu jego zastosowanie rozszerzyło się na inne obszary, takie jak serwery (za pomocą Node.js), aplikacje na smartfony, czy Internet Rzeczy (IoT). JavaScript obsługuje różne paradygmaty programowania, takie jak programowanie obiektowe, proceduralne czy funkcyjne, dzięki czemu jest bardzo elastyczny i może być używany do tworzenia różnorodnych aplikacji.

[14.txt]	JavaScript to wysokopoziomowy, dynamiczny, interpretowany język programowania, który został zaprojektowany głównie do użycia w kontekście przeglądarek internetowych. Jest to język obiektowy z pierwszoklasowymi funkcjami, co oznacza, że funkcje w JavaScript można przekazywać jako argumenty, przypisywać do zmiennych oraz tworzyć je w sposób dynamiczny. JavaScript został stworzony przez Brendana Eicha w 1995 roku i pierwotnie nazywał się Mocha, a następnie LiveScript, zanim ostatecznie przyjął nazwę JavaScript. Pomimo nazwy, JavaScript nie jest bezpośrednio powiązany z językiem Java, choć oba języki mają pewne podobieństwa składniowe. JavaScript jest powszechnie stosowany w programowaniu front-endu, pozwalając na tworzenie interaktywnych stron internetowych, obsługę zdarzeń, manipulację elementami DOM (Document Object Model) oraz obsługę asynchronicznych operacji, takich jak żądania AJAX. W ostatnich latach, dzięki platformie Node.js, JavaScript zyskał także na popularności jako język do tworzenia aplikacji back-endowych, co pozwala na tworzenie pełnoprawnych aplikacji webowych wykorzystujących jedynie JavaScript. Warto dodać, że JavaScript jest jednym z trzech podstawowych języków używanych do tworzenia stron internetowych, obok HTML (struktura) i CSS (stylowanie).
[15.txt]	JavaScript to wysokopoziomowy, dynamiczny, interpretowany język programowania, który jest używany głównie do tworzenia interaktywnych i dynamicznych stron internetowych. Został stworzony przez Brendana Eicha w 1995 roku i pierwotnie nazywał się Mocha, a następnie LiveScript. Ostatecznie nazwa została zmieniona na JavaScript, co miało na celu pokazanie, że język jest luźno powiązany z Javą, popularnym wówczas językiem programowania. JavaScript jest obecnie jednym z najpopularniejszych języków programowania na świecie. Jest językiem skryptowym, co oznacza, że jego kod jest wykonywany w przeglądarce internetowej klienta, a nie na serwerze. JavaScript jest szeroko wykorzystywany do dodawania interaktywności do stron internetowych, takiej jak animacje, przewijanie obrazów, walidacja formularzy czy tworzenie gier. JavaScript jest językiem o szerokim zastosowaniu, dzięki czemu wraz z rozwojem technologii webowych, zaczęto go również stosować poza przeglądarką. Node.js, na przykład, to platforma oparta na silniku JavaScript V8, która pozwala na używanie JavaScript jako języka programowania po stronie serwera. Współcześnie JavaScript odgrywa kluczową rolę w ekosystemie internetowym i jest jednym z trzech głównych języków wykorzystywanych w tworzeniu stron internetowych, obok HTML (Hypertext Markup Language) i CSS (Cascading Style Sheets).
[16.txt]	JavaScript to wysokopoziomowy, interpretowany język programowania, który jest używany przede wszystkim do tworzenia interaktywnych stron internetowych. Został stworzony przez Brendana Eicha w 1995 roku i początkowo nazywał się Mocha, a później LiveScript, zanim ostatecznie przyjął nazwę JavaScript.

	JavaScript jest językiem obiektowym z możliwością stosowania paradygmatu programowania funkcyjnego. Jest językiem dynamicznym, co oznacza, że typy danych są ustalane w czasie wykonywania kodu, a nie w czasie kompilacji. Najważniejsze cechy języka JavaScript: Przeznaczenie głównie do tworzenia interaktywnych stron internetowych, ale też stosowany w serwerach (np. Node.js) czy aplikacjach mobilnych (React Native, Ionic). Możliwość integracji z technologiami takimi jak HTML i CSS, co umożliwia tworzenie dynamicznych i atrakcyjnych stron internetowych. Dynamiczne typowanie, co sprawia, że kod jest bardziej elastyczny, ale może prowadzić do błędów wynikających z nieoczekiwanych konwersji typów. Wbudowane obiekty i funkcje, które pozwalają na łatwe manipulowanie strukturą strony oraz komunikację z serwerem. Szerokie wsparcie ze strony przeglądarek internetowych, co sprawia, że kod JavaScript może być uruchamiany na różnych platformach i urządzeniach. JavaScript stał się jednym z najpopularniejszych języków programowania na świecie, a jego rosnące zastosowanie w różnych dziedzinach sprawia, że jest niezbędnym narzędziem w arsenale każdego programisty.
[17.txt]	JavaScript to popularny język programowania, stworzony przez Brendana Eicha w 1995 roku. Początkowo został opracowany dla przeglądarek internetowych jako język skryptowy, pozwalający na interakcję z użytkownikiem, kontrolowanie przeglądarki, komunikację z serwerem oraz modyfikację struktury i wyglądu strony internetowej. JavaScript to język wieloparadygmatowy, co oznacza, że obsługuje różne style programowania, takie jak programowanie obiektowe, funkcyjne i imperatywne. Jest to język o dynamicznym typowaniu, co oznacza, że zmienne mogą zmieniać swoje typy podczas działania programu. Chociaż JavaScript początkowo został zaprojektowany do użycia w przeglądarkach internetowych, z czasem jego zastosowanie rozszerzyło się. Dzięki platformie Node.js, JavaScript może być używany także po stronie serwera, co pozwala na tworzenie kompletnych aplikacji internetowych przy użyciu tylko jednego języka programowania. Inne zastosowania JavaScript to: aplikacje na smartfony, gry, narzędzia programistyczne czy systemy IoT (Internet Rzeczy).
[18.txt]	JavaScript to wysokopoziomowy, interpretowany język programowania, który został stworzony przez Brendana Eicha podczas pracy dla firmy Netscape Communications Corporation w 1995 roku. Początkowo znany jako LiveScript, później zmienił nazwę na JavaScript, aby nawiązać do popularności języka Java w tamtych czasach. Jednak warto zaznaczyć, że JavaScript i Java to dwa różne języki programowania. JavaScript jest językiem programowania skryptowego, co oznacza, że jest on wykonywany przez przeglądarkę internetową po stronie klienta. Został zaprojektowany z myślą o tworzeniu dynamicznych i interaktywnych stron internetowych, umożliwiając wykonywanie zadań takich jak obsługa zdarzeń, animacje, manipulacje elementami strony czy walidacja danych.

	JavaScript stał się nieodłącznym elementem większości nowoczesnych stron internetowych. JavaScript jest językiem o luźnym typowaniu i opiera się na prototypach, co oznacza, że obiekty mogą dziedziczyć właściwości i metody od innych obiektów bez konieczności korzystania z klas. W ostatnich latach JavaScript ewoluował i zyskał wsparcie dla wielu innych zastosowań, takich jak aplikacje serwerowe, dzięki środowisku Node.js, oraz rozwój technologii takich jak React, Angular czy Vue.js, które ułatwiają tworzenie zaawansowanych aplikacji internetowych. JavaScript jest obecnie jednym z najbardziej popularnych języków programowania na świecie i jest używany przez programistów do tworzenia różnorodnych aplikacji – od prostych stron internetowych po zaawansowane aplikacje mobilne i serwerowe.
[19.txt]	JavaScript to wysokopoziomowy, obiektowy, interpretowany język programowania, który jest często używany do tworzenia interaktywnych i dynamicznych stron internetowych. Choć pierwotnie został zaprojektowany jako język skryptowy dla przeglądarek internetowych, z czasem zyskał na popularności i zastosowaniach poza przeglądarką. JavaScript jest częścią standardu ECMAScript, co oznacza, że jego składnia i funkcje są zdefiniowane przez oficjalny standard. Dzięki temu jest on zgodny z różnymi środowiskami wykonawczymi, takimi jak przeglądarki czy serwery (np. Node.js). Język JavaScript pozwala na manipulowanie elementami strony internetowej, takimi jak tekst, obrazy czy formularze, reagowanie na zdarzenia, takie jak kliknięcia myszką, przewijanie strony czy wypełnianie formularza, a także komunikację z serwerem w celu pobierania danych lub zaktualizowania treści na stronie bez potrzeby przeładowania. Współcześnie JavaScript stał się jednym z kluczowych języków stosowanych w rozwoju aplikacji internetowych i jest częścią popularnego stosu technologicznego MEAN/MERN (MongoDB, Express.js, Angular/React, Node.js), który umożliwia tworzenie pełnoprawnych aplikacji webowych zarówno po stronie klienta, jak i serwera.
[20.txt]	JavaScript to wysokopoziomowy, obiektowy, dynamicznie typowany język programowania, często używany do tworzenia interaktywnych i responsywnych stron internetowych. Początkowo został stworzony przez Brendana Eicha w 1995 roku i jest rozwijany przez Mozilla Foundation. JavaScript jest jednym z trzech głównych technologii używanych do tworzenia stron internetowych, obok HTML (Hypertext Markup Language) i CSS (Cascading Style Sheets). JavaScript pozwala na manipulację elementami strony, obsługę zdarzeń (takich jak kliknięcia myszą czy naciśnięcia klawiszy), komunikację z serwerem za pomocą technologii AJAX (Asynchronous JavaScript and XML), a także tworzenie animacji i gier. Jest to język interpretowany, co oznacza, że kod jest czytany i wykonywany przez przeglądarkę internetową na urządzeniu użytkownika.

	W ostatnich latach JavaScript zyskał na popularności także poza przeglądarką dzięki środowisku wykonawczemu Node.js, które pozwala na uruchamianie kodu JavaScript po stronie serwera. Dzięki temu język ten stał się popularnym narzędziem do tworzenia aplikacji internetowych, zarówno po stronie klienta, jak i serwera.
[21.txt]	JavaScript to wysokopoziomowy, interpretowany język programowania, który został wprowadzony w 1995 roku przez Brendana Eicha. JavaScript jest szeroko stosowany w technologiach internetowych, zwłaszcza w kontekście tworzenia stron i aplikacji internetowych. Język ten pozwala na dodawanie dynamicznych funkcji oraz interaktywności do stron internetowych, co sprawia, że są one bardziej użyteczne i atrakcyjne dla użytkowników. JavaScript pierwotnie był używany głównie po stronie klienta, czyli w przeglądarkach internetowych. Z czasem jednak jego zastosowanie rozszerzyło się także na inne obszary, takie jak serwery (Node.js) czy aplikacje na różne platformy (React Native). JavaScript jest oparty na standardzie ECMAScript, który jest specyfikacją języka i definiuje jego składnię oraz funkcje. Język ten obsługuje różne paradygmaty programowania, takie jak programowanie obiektowe, imperatywne i funkcyjne. Warto również wspomnieć, że JavaScript jest językiem luźno typowanym (ang. loosely typed), co oznacza, że nie ma konieczności deklarowania typów zmiennych. Ułatwia to szybsze prototypowanie i elastyczność podczas programowania, ale może również prowadzić do błędów związanych z nieoczekiwanymi konwersjami typów.
[22.txt]	JavaScript to wysokopoziomowy, dynamicznie typowany język programowania, który jest szeroko stosowany na całym świecie, zwłaszcza w kontekście tworzenia i obsługi stron internetowych. Choć początkowo JavaScript był używany głównie do tworzenia interaktywnych efektów na stronach internetowych, obecnie jego zastosowanie się rozszerzyło, obejmując zarówno rozwój stron internetowych, jak i aplikacji mobilnych, serwerowych, a także różnych platform, takich jak Node.js. JavaScript jest językiem interpretowanym, co oznacza, że kod nie jest kompilowany przed wykonaniem, a zamiast tego jest interpretowany przez przeglądarkę lub środowisko uruchomieniowe w czasie rzeczywistym. Język ten obsługuje różne paradygmaty programowania, takie jak programowanie imperatywne, obiektowe i funkcyjne, co sprawia, że jest bardzo elastyczny i przystępny dla szerokiego grona programistów. Podstawową zaletą JavaScript jest możliwość tworzenia dynamicznych i interaktywnych stron internetowych, które reagują na działania użytkownika, takie jak kliknięcia czy wpisywanie tekstu. Dzięki temu językowi można również manipulować elementami struktury i stylami stron, co pozwala na tworzenie bogatych, angażujących interfejsów użytkownika. JavaScript jest kluczowym elementem tzw. technologii front-end, wraz z HTML (HyperText Markup Language) i CSS (Cascading Style Sheets), które razem pozwalają na tworzenie nowoczesnych i efektywnych stron internetowych.

[23.txt]	JavaScript to wysokopoziomowy, dynamiczny, interpretowany język programowania, który pierwotnie został stworzony w 1995 roku przez Brendana Eicha, pracującego wówczas w firmie Netscape Communications. JavaScript jest najbardziej znany jako język skryptowy używany w przeglądarkach internetowych do tworzenia interaktywnych stron internetowych. Język ten pozwala na manipulację zawartością strony, obsługę zdarzeń (takich jak kliknięcia czy ruchy myszką), komunikację z serwerem w tle (np. za pomocą AJAX) oraz tworzenie animacji i innych efektów wizualnych. JavaScript jest językiem obiektowym, obsługującym zarówno programowanie proceduralne, jak i funkcyjne. Z biegiem czasu JavaScript ewoluował i zyskał na popularności nie tylko jako język przeglądarkowy, ale również jako technologia stosowana po stronie serwera (np. za pomocą platformy Node.js) oraz w tworzeniu aplikacji mobilnych i desktopowych. Warto również wspomnieć o bibliotekach i frameworkach JavaScript, takich jak jQuery, React, Angular czy Vue, które ułatwiają tworzenie zaawansowanych i interaktywnych aplikacji internetowych.
[24.txt]	JavaScript to wysokopoziomowy, interpretowany język programowania, który został stworzony przez Brendana Eicha w 1995 roku. Jest to jeden z trzech głównych technologii stosowanych w projektowaniu stron internetowych, obok HTML (HyperText Markup Language) i CSS (Cascading Style Sheets). JavaScript pierwotnie został zaprojektowany do ułatwienia tworzenia interaktywnych stron internetowych, poprzez dodawanie efektów, animacji, walidacji formularzy czy dynamicznego ładowania treści. Z biegiem czasu, jego zastosowanie znacząco się rozszerzyło i obecnie jest używany również poza przeglądarkami internetowymi, np. na serwerach (Node.js), urządzeniach IoT czy aplikacjach mobilnych. JavaScript jest językiem o dynamicznym typowaniu, co oznacza, że zmienne nie mają określonego typu aż do momentu przypisania wartości. Język ten wspiera wiele paradygmatów programowania, takich jak obiektowe, funkcyjne czy imperatywne. Warto również wspomnieć, że mimo podobieństwa nazwy, JavaScript nie jest powiązany z językiem Java - oba języki mają zupełnie inną strukturę, składnię oraz zastosowanie.
[25.txt]	JavaScript to wysokopoziomowy, dynamiczny, interpretowany język programowania, który jest powszechnie używany na stronach internetowych i w aplikacjach webowych. Został stworzony przez Brendana Eicha w 1995 roku i jest rozwijany przez organizację Mozilla Foundation oraz ECMA International. JavaScript jest głównym językiem programowania używanym do tworzenia interaktywnych efektów i funkcjonalności na stronach internetowych, takich jak animacje, walidacja formularzy, obsługa zdarzeń czy manipulacja elementami strony. Działa on na stronie klienta (w przeglądarce użytkownika) i jest zgodny z większością współczesnych przeglądarek internetowych.

	JavaScript jest również używany poza przeglądarką, na przykład na serwerach, za pomocą środowiska uruchomieniowego Node.js, które pozwala na tworzenie aplikacji serwerowych w języku JavaScript. Współcześnie JavaScript jest stosowany także w innych dziedzinach, takich jak tworzenie aplikacji mobilnych, desktopowych, a nawet gier.

7.5. Teksty napisane w językach hiszpańskim i portugalskim

7.5.1. Artykuł o Wyspach Kanaryjskich – język polski

Wyspy Kanaryjskie, zwane również „Wyspami Wiecznej Wiosny" ze względu na ich łagodny klimat przez cały rok, to archipelag położony na Oceanie Atlantyckim, stanowiący jedno z autonomicznych wspólnot Hiszpanii. Ich unikalna lokalizacja, nieopodal północno-zachodnich wybrzeży Afryki, czyni je popularnym miejscem wypoczynku oraz ważnym punktem na mapie badań biologicznych i geologicznych.

Geografia i klimat
Archipelag składa się z siedmiu głównych wysp: Tenerife, Fuerteventura, Gran Canaria, Lanzarote, La Palma, La Gomera oraz El Hierro. Każda z nich oferuje unikatowy krajobraz – od piaszczystych plaż po majestatyczne wulkany i gęste lasy laurowe. Wyspy charakteryzują się subtropikalnym klimatem, który jest łagodzony przez chłodne Prądy Kanaryjskie. To sprawia, że temperatury rzadko spadają poniżej 18°C w zimie czy przekraczają 25°C w lecie, co przyciąga turystów przez cały rok.

Historia
Wyspy Kanaryjskie mają bogatą historię. Pierwsi znani mieszkańcy, Guanczowie, byli rdzennymi Amazygami, którzy przybyli na wyspy około 1000 roku p.n.e. W XV wieku Wyspy Kanaryjskie stały się celem ekspedycji Europejczyków, przede wszystkim Hiszpanów, którzy ostatecznie anektowali je do swojego królestwa. Ten okres kolonizacji był również początkiem wpływu europejskiego na kulturę, język i gospodarkę archipelagu.

Kultura
Współczesna kultura Wysp Kanaryjskich jest mieszanką wpływów hiszpańskich i rdzennych tradycji Guanczów. Lokalna muzyka, taniec (jak np. tajna rytmiczna muzyka i taniec), oraz festiwale odzwierciedlają to dziedzictwo. Język hiszpański jest językiem urzędowym, ale można również usłyszeć lokalny dialekt, który zachował wiele rdzennych słów.

Gospodarka
Gospodarka Wysp Kanaryjskich jest silnie uzależniona od turystyki, która stanowi jej główne źródło dochodu. Rocznie archipelag odwiedza ponad 12 milionów turystów, przyciąganych przez piękne plaże, malownicze krajobrazy i bogatą ofertę kulturalną. Poza turystyką, wyspy mają rozwinięty sektor rolnictwa, w tym produkcję bananów, pomidorów i innych owoców subtropikalnych, które są eksportowane głównie do krajów Unii Europejskiej.

Przyroda i ochrona środowiska
Wyspy Kanaryjskie są domem dla wielu endemicznych gatunków roślin i zwierząt, które przystosowały się do unikalnych warunków środowiskowych.

Parki Narodowe, takie jak Teide na Tenerife czy Garajonay na La Gomerze, są chronione w ramach działań na rzecz zachowania tych unikalnych ekosystemów. Ochrona środowiska jest również ważnym aspektem polityki lokalnych władz, zwłaszcza w kontekście rosnącej presji wynikającej z intensywnego rozwoju turystycznego.

Podsumowanie
Wyspy Kanaryjskie, z ich różnorodnymi krajobrazami, bogatą historią i kulturą oraz stałym klimatem, stanowią fascynujący cel podróży. Dla wielu są synonimem ucieczki od codzienności, oferując zarówno przygodę, jak i relaks. Ich znaczenie jako centrum badań naukowych i ochrony przyrody podkreśla ich wartość nie tylko dla turystów, ale i dla naukowców oraz ekologów.

7.5.2. Artykuł o Wyspach Kanaryjskich – język portugalski

As Ilhas Canárias, também conhecidas como as "Ilhas da Eterna Primavera" devido ao seu clima ameno durante todo o ano, são um arquipélago localizado no Oceano Atlântico, formando uma das comunidades autônomas da Espanha. Sua localização única, próxima às costas noroeste da África, torna-as um destino turístico popular e um ponto importante no mapa de pesquisas biológicas e geológicas.

Geografia e clima
O arquipélago é composto por sete ilhas principais: Tenerife, Fuerteventura, Gran Canaria, Lanzarote, La Palma, La Gomera e El Hierro. Cada uma oferece uma paisagem única, desde praias arenosas até vulcões majestosos e densas florestas de loureiros. As Ilhas Canárias têm um clima subtropical, moderado pelas frescas Correntes Canárias. Isso mantém as temperaturas raramente abaixo de 18°C no inverno ou acima de 25°C no verão, atraindo turistas durante todo o ano.

História
As Ilhas Canárias têm uma rica história. Os primeiros habitantes conhecidos, os guanches, eram nativos amazighs que chegaram às ilhas por volta do ano 1000 a.C. No século XV, as Ilhas Canárias tornaram-se alvo das expedições europeias, principalmente dos espanhóis, que eventualmente as anexaram ao seu reino. Esse período de colonização também marcou o início da influência europeia na cultura, língua e economia do arquipélago.

Cultura
A cultura contemporânea das Ilhas Canárias é uma mistura de influências espanholas e tradições nativas guanches. A música local, a dança (como o tajaraste, uma dança e música rítmicas) e os festivais refletem esse patrimônio. O espanhol é a língua oficial, mas também se pode ouvir um dialeto local que preservou muitas palavras nativas.

Economia
A economia das Ilhas Canárias depende fortemente do turismo, que é sua principal fonte de receita. Anualmente, o arquipélago recebe mais de 12 milhões de turistas, atraídos por suas belas praias, paisagens pitorescas e uma rica oferta cultural. Além do turismo, as ilhas têm um setor agrícola

desenvolvido, incluindo a produção de bananas, tomates e outros frutos subtropicais que são exportados principalmente para os países da União Europeia.

Natureza e conservação ambiental
As Ilhas Canárias são o lar de muitas espécies endêmicas de plantas e animais que se adaptaram às condições ambientais únicas. Os parques nacionais, como o Teide em Tenerife ou Garajonay em La Gomera, estão protegidos como parte dos esforços para conservar esses ecossistemas únicos. A proteção ambiental também é um aspecto importante da política das autoridades locais, especialmente no contexto da crescente pressão derivada do desenvolvimento turístico intensivo.

Conclusão
As Ilhas Canárias, com suas diversas paisagens, rica história e cultura, bem como seu clima constante, representam um destino fascinante. Para muitos, são sinônimo de fuga da rotina diária, oferecendo tanto aventura quanto relaxamento. Sua importância como centro de pesquisa científica e conservação da natureza destaca seu valor não apenas para turistas, mas também para cientistas e ecologistas.

7.5.3. Artykuł o Wyspach Kanaryjskich – język hiszpański

Las Islas Canarias, también conocidas como las "Islas de la Eterna Primavera" debido a su clima templado durante todo el año, son un archipiélago ubicado en el Océano Atlántico, que forma una de las comunidades autónomas de España. Su ubicación única, cerca de las costas noroccidentales de África, las convierte en un popular destino turístico y un punto importante en el mapa de investigaciones biológicas y geológicas.

Geografía y clima
El archipiélago está compuesto por siete islas principales: Tenerife, Fuerteventura, Gran Canaria, Lanzarote, La Palma, La Gomera y El Hierro. Cada una ofrece un paisaje único, desde playas arenosas hasta majestuosos volcanes y densos bosques de laurisilva. Las Islas Canarias tienen un clima subtropical, moderado por las frescas Corrientes Canarias. Esto mantiene las temperaturas raramente por debajo de los 18°C en invierno o por encima de los 25°C en verano, atrayendo turistas durante todo el año.

Historia
Las Islas Canarias tienen una rica historia. Los primeros habitantes conocidos, los guanches, eran amazighs nativos que llegaron a las islas alrededor del año 1000 a.C. En el siglo XV, las Islas Canarias se convirtieron en objetivo de las expediciones europeas, principalmente de los españoles, quienes finalmente las anexaron a su reino. Este período de colonización también marcó el comienzo de la influencia europea en la cultura, el idioma y la economía del archipiélago.

Cultura
La cultura contemporánea de las Islas Canarias es una mezcla de influencias españolas y tradiciones nativas guanches. La música local, la danza (como

el tajaraste, un baile y música rítmicos) y los festivales reflejan este patrimonio. El español es el idioma oficial, pero también se puede escuchar un dialecto local que ha conservado muchas palabras nativas.

Economía
La economía de las Islas Canarias depende en gran medida del turismo, que es su principal fuente de ingresos. Anualmente, el archipiélago recibe más de 12 millones de turistas, atraídos por sus hermosas playas, paisajes pintorescos y una rica oferta cultural. Además del turismo, las islas tienen un sector agrícola desarrollado, incluyendo la producción de plátanos, tomates y otros frutos subtropicales que se exportan principalmente a los países de la Unión Europea.

Naturaleza y conservación del medio ambiente
Las Islas Canarias son hogar de muchas especies endémicas de plantas y animales que se han adaptado a las condiciones ambientales únicas. Los parques nacionales, como el Teide en Tenerife o Garajonay en La Gomera, están protegidos como parte de los esfuerzos para conservar estos ecosistemas únicos. La protección del medio ambiente también es un aspecto importante de la política de las autoridades locales, especialmente en el contexto de la creciente presión derivada del desarrollo turístico intensivo.

Conclusión
Las Islas Canarias, con sus diversos paisajes, rica historia y cultura, así como su clima constante, representan un destino fascinante. Para muchos, son sinónimo de escape de la rutina diaria, ofreciendo tanto aventura como relajación. Su importancia como centro de investigación científica y conservación de la naturaleza subraya su valor no solo para los turistas, sino también para científicos y ecologistas.

7.6. Teksty napisane w językach czeskim i słowackim

7.6.1. Artykuł o Czechosłowacji – język polski

Historia Czechosłowacji: Od powstania do rozpadu

Powstanie Czechosłowacji
Czechosłowacja, państwo istniejące w latach 1918–1992, została utworzona w wyniku upadku Austro-Węgier po zakończeniu I wojny światowej. Nowa republika ogłoszona została 28 października 1918 roku, obejmująca ziemie historyczne Czech, Moraw, Śląska oraz Słowacji i Rusi Zakarpackiej. Kształtowanie się państwa czeskosłowackiego było wynikiem dążeń narodowych Czechów i Słowaków, których liderami byli odpowiednio Tomáš Masaryk i Milan Rastislav Štefánik. Masaryk, pierwszy prezydent Czechosłowacji, stał się symbolem stabilności i demokracji.

Okres międzywojenny i II wojna światowa
Czechosłowacja w okresie międzywojennym była postrzegana jako jedna z bardziej stabilnych i demokratycznych republik w Europie Środkowej. Jednakże, różnice etniczne wewnątrz państwa, szczególnie silne napięcia między Czechami a Niemcami sudeckimi, stawały się coraz bardziej problematyczne. W wyniku układu monachijskiego z 1938 roku, terytorium

Czechosłowacji zostało znacząco zmniejszone, a w marcu 1939 roku państwo zostało rozwiązane, a jego terytorium zajęte przez Niemcy.

Powojenne zmiany i komunizm
Po zakończeniu II wojny światowej Czechosłowacja została przywrócona w granicach z 1937 roku, ale bez Sudetów, które pozostały w Niemczech. W 1948 roku, w wyniku zamachu stanu, władzę przejęli komuniści, co zainaugurowało czterodekadową erę rządów jednopartyjnych pod egidą Komunistycznej Partii Czechosłowacji. Rządy te były charakteryzowane przez centralne planowanie gospodarcze, represje polityczne oraz bliskie związki z ZSRR.

Praska Wiosna i jej upadek
W 1968 roku doszło do liberalizującego ruchu znanego jako Praska Wiosna. Alexander Dubček, lider partii, próbował wprowadzić „socjalizm z ludzką twarzą", co oznaczało większą swobodę prasy, wolność słowa i ograniczenie kontroli partii. Jednak te reformy szybko zostały zatrzymane przez interwencję wojsk Układu Warszawskiego, co brutalnie zakończyło krótki okres liberalizacji.

Rozpad Czechosłowacji
Pod koniec lat 80. XX wieku, w atmosferze ogólnoświatowych przemian demokratycznych, także Czechosłowacja doświadczyła serii protestów, które doprowadziły do aksamitnej rewolucji w 1989 roku. Rewolucja ta zakończyła komunistyczną władzę i wprowadziła system wielopartyjny z Václavem Havlem jako prezydentem. Mimo tych pozytywnych zmian, napięcia między liderami czeskimi a słowackimi nasiliły się, co ostatecznie doprowadziło do pokojowego rozpadu państwa na Republikę Czeską i Republikę Słowacką 1 stycznia 1993 roku.

Podsumowanie
Historia Czechosłowacji jest przykładem złożoności narodowych i etnicznych dążeń w Europie Środkowej. Kraj ten, mimo licznych wyzwań, przez większość swojego istnienia był miejscem względnej stabilności i rozwoju, choć ostatecznie nie uniknął rozpadu, który stał się finałem jego burzliwej historii.

7.6.2. Artykuł o Czechosłowacji – język czeski

Historie Československa: Od vzniku po rozpad

Vznik Československa
Československo, stát existující v letech 1918–1992, bylo založeno po pádu Rakousko-Uherska na konci první světové války. Nová republika byla vyhlášena 28. října 1918 a zahrnovala historická území Čech, Moravy, Slezska, Slovenska a Podkarpatské Rusi. Formování československého státu bylo výsledkem národních snah Čechů a Slováků, jejichž vůdci byli Tomáš Garrigue Masaryk a Milan Rastislav Štefánik. Masaryk, první prezident Československa, se stal symbolem stability a demokracie.

Meziválečné období a druhá světová válka

Československo bylo v meziválečném období považováno za jednu z nejstabilnějších a demokratičtějších republik ve střední Evropě. Nicméně, etnické rozdíly uvnitř státu, zejména rostoucí napětí mezi Čechy a německými Sudetskými Němci, se stávaly čím dál tím problematickějšími. V důsledku Mnichovské dohody z roku 1938 došlo k výraznému zmenšení území Československa, a v březnu 1939 byl stát rozpuštěn a jeho území obsazeno Německem.

Poválečné změny a komunismus
Po skončení druhé světové války bylo Československo obnoveno ve svých hranicích z roku 1937, avšak bez Sudet, které zůstaly v Německu. V roce 1948 došlo k puči, po kterém se moci chopili komunisté, což zahájilo čtyři dekády trvající éru jednopartyjních vlád pod vedením Komunistické strany Československa. Tyto režimy byly charakterizovány centrálním plánováním ekonomiky, politickými represími a úzkými vazbami na SSSR.

Pražské jaro a jeho pád
V roce 1968 došlo k liberalizačnímu hnutí známému jako Pražské jaro. Alexander Dubček, vůdce strany, se pokusil zavést "socialismus s lidskou tváří", což znamenalo větší svobodu tisku, svobodu slova a omezení kontroly strany. Avšak tyto reformy byly rychle zastaveny intervencí vojsk Varšavské smlouvy, což brutálně ukončilo krátké období liberalizace.

Rozpad Československa
Ke konci 80. let 20. století, v atmosféře globálních demokratických změn, Československo také zažilo řadu protestů, které vedly k sametové revoluci v roce 1989. Tato revoluce ukončila komunistickou vládu a zavedla vícestranický systém s Václavem Havlem jako prezidentem. Přestože tyto změny byly pozitivní, napětí mezi českými a slovenskými lídry se zintenzivnilo, což nakonec vedlo k mírovému rozpadu státu na Českou republiku a Slovenskou republiku 1. ledna 1993.

Závěr
Historie Československa je příkladem složitosti národních a etnických aspirací ve střední Evropě. Tento stát, přes mnohé výzvy, byl většinu své existence místem relativní stability a rozvoje, ačkoliv nakonec nedokázal uniknout rozpadu, který se stal závěrečným aktem jeho bouřlivé historie.

7.6.3. Artykuł o Czechosłowacji – język słowacki

História Československa: Od vzniku po rozpad

Vznik Československa
Československo, štát existujúci v rokoch 1918–1992, vznikol po páde Rakúsko-Uhorska na konci prvej svetovej vojny. Nová republika bola vyhlásená 28. októbra 1918 a zahŕňala historické územia Čiech, Moravy, Sliezska, Slovenska a Podkarpatskej Rusi. Formovanie československého štátu bolo výsledkom národných snáh Čechov a Slovákov, ktorých lídrami boli Tomáš Garrigue Masaryk a Milan Rastislav Štefánik. Masaryk, prvý prezident Československa, sa stal symbolom stability a demokracie.

Medzivojnové obdobie a druhá svetová vojna

Československo bolo v medzivojnovom období považované za jednu z najstabilnejších a demokratickejších republík v strednej Európe. Avšak etnické rozdiely vnútri štátu, najmä rastúce napätie medzi Čechmi a nemeckými Sudetskými Nemcami, sa stávali čoraz problematickejšími. V dôsledku Mníchovskej dohody z roku 1938 došlo k výraznému zmenšeniu územia Československa a v marci 1939 bol štát rozpustený a jeho územie obsadené Nemeckom.

Povojnové zmeny a komunizmus
Po skončení druhej svetovej vojny bolo Československo obnovené v hraniciach z roku 1937, avšak bez Sudiet, ktoré zostali v Nemecku. V roku 1948 došlo k prevratu, po ktorom sa moci chopili komunisti, čo zahájilo štyri desaťročia trvajúce obdobie jednopartejných vlád pod vedením Komunistickej strany Československa. Tieto režimy boli charakterizované centrálnym plánovaním ekonomiky, politickými represiami a úzkymi väzbami na ZSSR.

Pražská jar a jej pád
V roku 1968 došlo k liberalizačnému hnutiu známemu ako Pražská jar. Alexander Dubček, líder strany, sa pokúsil zaviesť „socializmus s ľudskou tvárou", čo znamenalo väčšiu slobodu tlače, slobodu slova a obmedzenie kontroly strany. Avšak tieto reformy boli rýchlo zastavené intervenciou vojsk Varšavskej zmluvy, čo brutálne ukončilo krátke obdobie liberalizácie.

Rozpad Československa
Na konci 80. rokov 20. storočia, v atmosfére globálnych demokratických zmien, Československo tiež zažilo rad protestov, ktoré viedli k zamatovej revolúcii v roku 1989. Táto revolúcia ukončila komunistickú vládu a zaviedla viacstranícky systém s Václavom Havelom ako prezidentom. Napriek týmto pozitívnym zmenám sa napätie medzi českými a slovenskými lídrami zintenzívnilo, čo nakoniec viedlo k mierovému rozpadu štátu na Českú republiku a Slovenskú republiku 1. januára 1993.

Záver
História Československa je príkladom zložitosti národných a etnických aspirácií v strednej Európe. Tento štát, napriek mnohým výzvam, bol väčšinu svojej existencie miestom relatívnej stability a rozvoja, hoci nakoniec neunikol rozpadu, ktorý sa stal záverečným aktom jeho búrlivej histórie.

7.7. Teksty napisane w językach duńskim i norweskim

7.7.1. Artykuł o Skandynawii – język polski

Skandynawia: Region Innowacji i Zrównoważonego Rozwoju

Wstęp
Skandynawia, region północnej Europy, składający się z trzech krajów: Danii, Norwegii i Szwecji, jest często postrzegana jako wzór zrównoważonego rozwoju, innowacyjności oraz wysokiej jakości życia. Finlandia i Islandia, choć niekiedy zaliczane do krajów skandynawskich, geograficznie i kulturowo należą do regionu nordyckiego. Skandynawia jest

znana z surowych, ale pięknych krajobrazów, zaawansowanych technologii oraz silnego zobowiązania do ochrony środowiska i równości społecznej.

Geografia i Klimat
Skandynawia charakteryzuje się różnorodnością geograficzną – od rozległych górskich łańcuchów w Norwegii, przez zielone lasy w Szwecji, po płaskie tereny rolne w Danii. Region ten jest również znany z surowego, chłodnego klimatu z długimi, mroźnymi zimami i krótkimi, ale jasnymi latami, co znacząco wpływa na styl życia i kulturę mieszkańców.

Gospodarka
Skandynawia jest domem dla wielu innowacyjnych przedsiębiorstw i ma jedne z najwyższych na świecie wskaźników PKB per capita. Duże znaczenie w gospodarce regionu mają przemysły takie jak telekomunikacja, biotechnologie, a także energia odnawialna. W Norwegii szczególne znaczenie ma przemysł wydobywczy, w Danii dominuje sektor usług i technologie zielone, a Szwecja jest znana z zaawansowanego przemysłu motoryzacyjnego i technologicznego.

Społeczeństwo i Kultura
Skandynawskie społeczeństwa są jednymi z najbardziej egalitarnych na świecie, z silnie rozbudowanymi systemami opieki społecznej, edukacji i zdrowia. Wysoki poziom życia, niskie nierówności dochodowe oraz zobowiązanie do równości płci są kluczowymi elementami skandynawskiego modelu społecznego. Kultura regionu jest głęboko zakorzeniona w historii Wikingów, a także w tradycjach takich jak święto św. Łucji czy święta midsommar.

Innowacje i Edukacja
Kraje skandynawskie są wiodącymi graczami na arenie globalnej w dziedzinie edukacji i innowacji. Systemy edukacyjne w regionie kładą duży nacisk na kreatywność, samodzielne myślenie i naukę przez działanie. Uniwersytety takie jak Uniwersytet w Kopenhadze, Uniwersytet w Oslo czy Uniwersytet w Sztokholmie są cenione za badania naukowe i rozwój technologiczny.

Środowisko
Ochrona środowiska jest jednym z filarów skandynawskiej polityki. Wysokie standardy ekologiczne, promocja energii odnawialnej oraz zaangażowanie w międzynarodowe umowy klimatyczne sprawiają, że kraje te są pionierami w dziedzinie zrównoważonego rozwoju. Inicjatywy takie jak ekologiczne miasta, rozwój zielonej energii i zrównoważony transport są obecne na każdym kroku.

Wyjątkowość regionu
Unikalność Skandynawii wykracza poza jej granice. Społeczna świadomość, zaawansowane technologie i zobowiązanie do zrównoważonego rozwoju czynią ten region nie tylko miejscem godnym odwiedzenia, ale również wzorem do naśladowania dla innych krajów i społeczeństw na całym świecie.

Zakończenie
Skandynawia jest regionem, który nieprzerwanie fascynuje i inspiruje. Jej mieszkańcy żyją w zgodzie z naturą, przy jednoczesnym rozwijaniu technologii i utrzymywaniu wysokiej jakości życia. To połączenie

innowacji, kultury i tradycji sprawia, że Skandynawia jest prawdziwie wyjątkowym miejscem na mapie świata.

7.7.2. Artykuł o Skandynawii – język duński

Skandinavien: Innovation og bæredygtig udvikling

Introduktion
Skandinavien, en region i Nordeuropa bestående af tre lande: Danmark, Norge og Sverige, ses ofte som et forbillede for bæredygtig udvikling, innovation og høj livskvalitet. Finland og Island, selvom de nogle gange regnes for skandinaviske lande, tilhører geografisk og kulturelt den nordiske region. Skandinavien er kendt for sine barske, men smukke landskaber, avancerede teknologier og en stærk forpligtelse til miljøbeskyttelse og social lighed.

Geografi og klima
Skandinavien er kendetegnet ved sin geografiske diversitet - fra de omfattende bjergkæder i Norge, de grønne skove i Sverige til de flade landbrugsarealer i Danmark. Regionen er også kendt for sit strenge, kølige klima med lange, frostfulde vintre og korte, men lyse somre, hvilket har en markant indflydelse på indbyggernes livsstil og kultur.

Økonomi
Skandinavien er hjemsted for mange innovative virksomheder og har nogle af verdens højeste BNP pr. indbygger. Vigtige økonomiske sektorer i regionen omfatter telekommunikation, bioteknologi samt vedvarende energi. I Norge er minedrift særligt betydningsfuldt, i Danmark dominerer serviceindustrien og grønne teknologier, mens Sverige er kendt for sin avancerede bil- og teknologiindustri.

Samfund og kultur
Skandinaviske samfund er blandt de mest egalitære i verden med meget udviklede systemer for social velfærd, uddannelse og sundhed. Høj livskvalitet, lave indkomstuligheder og en forpligtelse til kønslighed er nøgleelementer i den skandinaviske sociale model. Kulturen i regionen er dybt rodfæstet i vikingernes historie samt traditioner såsom Sankt Lucias dag og midsommarfester.

Innovation og uddannelse
De skandinaviske lande er førende på den globale scene inden for uddannelse og innovation. Uddannelsessystemerne i regionen lægger stor vægt på kreativitet, selvstændig tænkning og læring gennem handling. Universiteter såsom Københavns Universitet, Universitetet i Oslo og Stockholms Universitet er højt værdsatte for deres forskning og teknologiske udvikling.

Miljø
Miljøbeskyttelse er en af de grundlæggende søjler i skandinavisk politik. Høje miljøstandarder, fremme af vedvarende energi og engagement i internationale klimaaftaler gør, at disse lande er pionerer inden for

bæredygtig udvikling. Initiativer som økologiske byer, udvikling af grøn energi og bæredygtig transport er synlige overalt.

Regionens unikke karakter
Skandinaviens unikhed strækker sig ud over dens grænser. Social bevidsthed, avanceret teknologi og et engagement i bæredygtig udvikling gør denne region ikke kun til et værdigt besøgsmål, men også til et forbillede for andre lande og samfund verden over.

Afslutning
Skandinavien er en region, der konstant fascinerer og inspirerer. Dens indbyggere lever i harmoni med naturen, mens de udvikler teknologi og opretholder en høj livskvalitet. Denne kombination af innovation, kultur og tradition gør Skandinavien til et virkelig unikt sted på verdenskortet.

7.7.3. Artykuł o Skandynawii – język norweski

Skandinavia: Innovasjon og bærekraftig utvikling

Innledning
Skandinavia, en region i Nord-Europa som består av tre land: Danmark, Norge og Sverige, er ofte sett på som et forbilde for bærekraftig utvikling, innovasjon og høy livskvalitet. Finland og Island, selv om de noen ganger regnes som skandinaviske land, tilhører geografisk og kulturelt den nordiske regionen. Skandinavia er kjent for sine barske, men vakre landskap, avanserte teknologier og et sterkt engasjement for miljøvern og sosial likhet.

Geografi og klima
Skandinavia kjennetegnes av sin geografiske mangfoldighet – fra de omfattende fjellkjedene i Norge, de grønne skogene i Sverige, til de flate landbruksområdene i Danmark. Regionen er også kjent for sitt strenge, kjølige klima med lange, frostfulle vintre og korte, men lyse somre, noe som har en betydelig innvirkning på innbyggernes livsstil og kultur.

Økonomi
Skandinavia er hjemsted for mange innovative selskaper og har noen av verdens høyeste BNP per innbygger. Viktige økonomiske sektorer i regionen inkluderer telekommunikasjon, bioteknologi, samt fornybar energi. I Norge er utvinningsindustrien spesielt viktig, i Danmark dominerer servicesektoren og grønn teknologi, mens Sverige er kjent for sin avanserte bil- og teknologiindustri.

Samfunn og kultur
Skandinaviske samfunn er blant de mest egalitære i verden, med svært utviklede systemer for sosial velferd, utdanning og helse. Høy livskvalitet, lave inntektsulikheter og et engasjement for kjønnslikestilling er nøkkelkomponenter i den skandinaviske sosiale modellen. Kulturen i regionen er dypt forankret i vikinghistorien, samt tradisjoner som St. Lucia-dagen og midsommar-feiringer.

Innovasjon og utdanning

De skandinaviske landene er ledende aktører på den globale arenaen innen utdanning og innovasjon. Utdanningssystemene i regionen legger stor vekt på kreativitet, selvstendig tenkning og læring gjennom handling. Universiteter som Københavns Universitet, Universitetet i Oslo og Stockholms Universitet er høyt verdsatt for sin forskning og teknologiske utvikling.

Miljø
Miljøvern er en av de grunnleggende søylene i skandinavisk politikk. Høye miljøstandarder, fremming av fornybar energi og engasjement i internasjonale klimaavtaler gjør at disse landene er pionerer innen bærekraftig utvikling. Initiativer som økologiske byer, utvikling av grønn energi og bærekraftig transport er synlige overalt.

Regionens unikhet
Skandinavias unikhet strekker seg utover dens grenser. Sosial bevissthet, avansert teknologi og engasjement for bærekraftig utvikling gjør denne regionen ikke bare til et verdig besøksmål, men også til et forbilde for andre land og samfunn over hele verden.

Avslutning
Skandinavia er en region som stadig fascinerer og inspirerer. Dens innbyggere lever i harmoni med naturen, samtidig som de utvikler teknologi og opprettholder en høy livskvalitet. Denne kombinasjonen av innovasjon, kultur og tradisjon gjør Skandinavia til et virkelig unikt sted på verdenskartet.

7.8. Teksty napisane w językach niemieckim i niderlandzkim

7.8.1. Artykuł o Morzu Bałtyckim – język polski

Morze Bałtyckie, znane również jako Bałtyk, jest jednym z najmłodszych i najbardziej dynamicznie zmieniających się mórz na naszej planecie. Charakteryzuje się unikalnymi właściwościami zarówno pod względem geograficznym, jak i ekologicznym. Jest to morze półzamknięte, położone w Europie Północnej, otoczone przez dziewięć krajów: Polskę, Litwę, Łotwę, Estonię, Rosję, Finlandię, Szwecję, Danię i Niemcy. W niniejszym wypracowaniu omówię jego charakterystykę, znaczenie gospodarcze i ekologiczne, a także wyzwania, przed którymi stoi.

Geografia i środowisko naturalne
Bałtyk jest stosunkowo płytkim morzem o średniej głębokości zaledwie 55 metrów, co sprawia, że jest szczególnie podatne na zanieczyszczenia i zmiany środowiskowe. Jego najgłębszy punkt, Rowień Landsort, osiąga zaledwie 459 metrów. Morze Bałtyckie ma około 377 000 km² i jest połączone z Morzem Północnym cieśninami duńskimi, co wpływa na jego słoność i ekosystem. Średnia zasolenie Bałtyku wynosi od 0,3% w Zatoce Botnickiej do około 2% w cieśninach duńskich, co jest znacznie niższe niż w większości innych mórz, co przyczynia się do unikalności jego fauny i flory.

Bioróżnorodność i ekosystem
Morze Bałtyckie charakteryzuje się bogatym ekosystemem, który obejmuje wiele gatunków ryb, takich jak dorsz bałtycki, śledź, czy łosoś bałtycki.

Ponadto, wody te są domem dla licznych gatunków ptaków wodnych i ssaków morskich, w tym fok szarych i fok pospolitych. Unikalne warunki środowiskowe Bałtyku stwarzają idealne warunki dla roślinności morskiej, w tym dla rozległych ławic traw morskich, które są kluczowe dla utrzymania zdrowia ekosystemu. Niestety, obserwuje się negatywne skutki nadmiernego połowu i eutrofizacji, co prowadzi do zmniejszenia bioróżnorodności i degradacji siedlisk naturalnych.

Znaczenie gospodarcze
Morze Bałtyckie odgrywa kluczową rolę w gospodarce regionu. Jest ważnym szlakiem transportowym, który łączy liczne porty i przyczynia się do handlu międzynarodowego. Rybołówstwo, choć obecnie ograniczone z powodu przetrzebienia zasobów rybnych, nadal ma duże znaczenie lokalne. Turystyka również odgrywa istotną rolę, przyciągając do regionu miliony turystów, którzy chcą podziwiać malownicze krajobrazy i korzystać z dobrodziejstw przybrzeżnych kurortów. Ponadto, region Bałtyku staje się ważnym obszarem dla energetyki, w tym dla rozwoju morskich farm wiatrowych.

Wyzwania i ochrona środowiska
Jednym z największych wyzwań dla Morza Bałtyckiego jest ochrona jego delikatnego ekosystemu przed dalszą degradacją. Problem zanieczyszczenia wód, zwłaszcza związkami azotu i fosforu, które powodują eutrofizację, jest nadal bardzo poważny. Ponadto, morze to boryka się z zagrożeniami takimi jak zanieczyszczenia chemiczne, w tym ciężkie metale i mikroplastiki, które wpływają na zdrowie życia morskiego i ludzi. Równie istotne są działania na rzecz ograniczenia skutków zmian klimatycznych, które mogą prowadzić do dalszych zmian w ekosystemie.

Podsumowanie
Morze Bałtyckie jest nie tylko pięknym, ale i niezwykle ważnym zbiornikiem wodnym, od którego zależy dobrobyt i zdrowie wielu społeczności. Aby zapewnić jego przetrwanie dla przyszłych pokoleń, konieczne są dalsze badania, ścisła współpraca międzynarodowa i zdecydowane działania na rzecz ochrony środowiska. Odpowiedzialne zarządzanie zasobami naturalnymi i zrównoważony rozwój mogą przyczynić się do ochrony tego wyjątkowego ekosystemu, umożliwiając jego regenerację i zachowanie unikalnej bioróżnorodności.

7.8.2. Artykuł o Morzu Bałtyckim – język niemiecki

Die Ostsee, auch bekannt als Baltisches Meer, ist eines der jüngsten und dynamischsten Meere unseres Planeten. Sie zeichnet sich durch ihre einzigartigen geografischen und ökologischen Eigenschaften aus. Dieses halbgeschlossene Meer liegt in Nordeuropa und wird von neun Ländern umgeben: Polen, Litauen, Lettland, Estland, Russland, Finnland, Schweden, Dänemark und Deutschland. In diesem Essay werde ich seine Merkmale, seine wirtschaftliche und ökologische Bedeutung sowie die Herausforderungen, vor denen es steht, erörtern.

Geographie und natürliche Umwelt
Die Ostsee ist ein relativ flaches Meer mit einer durchschnittlichen Tiefe von nur 55 Metern, was sie besonders anfällig für Verschmutzung und

Umweltveränderungen macht. Ihr tiefster Punkt, die Landsort-Tief, erreicht gerade einmal 459 Meter. Das Baltische Meer umfasst etwa 377.000 km² und ist durch die dänischen Meerengen mit der Nordsee verbunden, was seine Salinität und sein Ökosystem beeinflusst. Die durchschnittliche Salinität der Ostsee variiert von 0,3% im Bottnischen Meerbusen bis zu etwa 2% in den dänischen Meerengen, was deutlich niedriger ist als in den meisten anderen Meeren und zur Einzigartigkeit seiner Fauna und Flora beiträgt.

Biodiversität und Ökosystem
Die Ostsee zeichnet sich durch ein reiches Ökosystem aus, das viele Fischarten wie den Ostsee-Dorsch, Hering und Ostsee-Lachs umfasst. Darüber hinaus sind diese Gewässer Heimat für zahlreiche Arten von Wasservögeln und Meeressäugetieren, einschließlich Grau- und Kegelrobben. Die einzigartigen Umweltbedingungen der Ostsee bieten ideale Bedingungen für die marine Vegetation, einschließlich ausgedehnter Seegraswiesen, die für die Gesundheit des Ökosystems von entscheidender Bedeutung sind. Leider gibt es negative Auswirkungen durch Überfischung und Eutrophierung, was zu einem Rückgang der Biodiversität und der Degradation natürlicher Lebensräume führt.

Wirtschaftliche Bedeutung
Die Ostsee spielt eine entscheidende Rolle in der Wirtschaft der Region. Sie ist eine wichtige Transportroute, die zahlreiche Häfen verbindet und zum internationalen Handel beiträgt. Obwohl die Fischerei aufgrund der Überfischung der Fischbestände derzeit eingeschränkt ist, bleibt sie lokal von großer Bedeutung. Auch der Tourismus spielt eine wichtige Rolle und zieht Millionen von Touristen in die Region, die die malerischen Landschaften genießen und die Annehmlichkeiten der Küstenorte nutzen möchten. Darüber hinaus wird das Baltikum zu einem wichtigen Bereich für die Energiegewinnung, einschließlich der Entwicklung von Offshore-Windparks.

Herausforderungen und Umweltschutz
Eine der größten Herausforderungen für die Ostsee ist der Schutz ihres empfindlichen Ökosystems vor weiterer Degradation. Das Problem der Wasserverschmutzung, insbesondere durch Stickstoff- und Phosphorverbindungen, die Eutrophierung verursachen, ist nach wie vor sehr ernst. Darüber hinaus kämpft das Meer mit Bedrohungen wie chemischen Kontaminationen, einschließlich Schwermetallen und Mikroplastiken, die das Leben im Meer und die menschliche Gesundheit beeinträchtigen. Ebenso wichtig sind Maßnahmen zur Begrenzung der Auswirkungen des Klimawandels, die zu weiteren Veränderungen im Ökosystem führen können.

Zusammenfassung
Die Ostsee ist nicht nur ein wunderschönes, sondern auch ein äußerst wichtiges Gewässer, von dem das Wohl und die Gesundheit vieler Gemeinschaften abhängen. Um ihr Überleben für zukünftige Generationen zu sichern, sind weitere Forschungen, enge internationale Zusammenarbeit und entschlossenes Handeln zum Umweltschutz notwendig. Verantwortungsvolles Ressourcenmanagement und nachhaltige Entwicklung können dazu beitragen, dieses einzigartige Ökosystem zu schützen, seine Regeneration zu ermöglichen und seine einzigartige Biodiversität zu bewahren.

7.8.3. Artykuł o Morzu Bałtyckim – język niderlandzki

De Oostzee, ook bekend als de Baltische Zee, is een van de jongste en meest dynamisch veranderende zeeën op onze planeet. Het is geografisch en ecologisch uniek en wordt omringd door negen landen: Polen, Litouwen, Letland, Estland, Rusland, Finland, Zweden, Denemarken en Duitsland. In dit essay zal ik de kenmerken, economische en ecologische belang, en de uitdagingen waar het voor staat, bespreken.

Geografie en natuurlijke omgeving
De Oostzee is relatief ondiep, met een gemiddelde diepte van slechts 55 meter, waardoor het bijzonder kwetsbaar is voor vervuiling en milieuveranderingen. Het diepste punt, de Landsort Diepte, bereikt slechts 459 meter. De zee beslaat ongeveer 377.000 km² en is via de Deense zeestraten verbonden met de Noordzee, wat de zoutgehalte en het ecosysteem beïnvloedt. De gemiddelde zoutgehalte van de Oostzee varieert van 0,3% in de Botnische Golf tot ongeveer 2% in de Deense zeestraten, wat veel lager is dan in de meeste andere zeeën en bijdraagt aan de uniciteit van zijn fauna en flora.

Biodiversiteit en ecosysteem
De Oostzee kenmerkt zich door een rijk ecosysteem dat vele vissoorten omvat, zoals de Oostzee-kabeljauw, haring en Oostzee-zalm. Bovendien zijn deze wateren thuis voor talrijke soorten watervogels en zeezoogdieren, inclusief grijze en gewone zeehonden. De unieke milieuomstandigheden van de Oostzee bieden ideale omstandigheden voor zeevegetatie, inclusief uitgestrekte zeegrasweiden, die cruciaal zijn voor de gezondheid van het ecosysteem. Helaas zijn er negatieve effecten door overbevissing en eutrofiëring, wat leidt tot een afname van de biodiversiteit en degradatie van natuurlijke habitats.

Economisch belang
De Oostzee speelt een cruciale rol in de economie van de regio. Het is een belangrijke transportroute die talrijke havens verbindt en bijdraagt aan de internationale handel. Hoewel de visserij momenteel beperkt is vanwege de uitputting van visbestanden, blijft het lokaal van groot belang. Toerisme speelt ook een belangrijke rol, trekt miljoenen toeristen naar de regio die willen genieten van de pittoreske landschappen en gebruik willen maken van de voorzieningen van kustresorts. Daarnaast wordt de Baltische regio een belangrijk gebied voor energieproductie, inclusief de ontwikkeling van offshore windparken.

Uitdagingen en milieubescherming
Een van de grootste uitdagingen voor de Oostzee is het beschermen van haar gevoelige ecosysteem tegen verdere degradatie. Het probleem van watervervuiling, met name door stikstof- en fosforverbindingen die eutrofiëring veroorzaken, is nog steeds zeer ernstig. Daarnaast worstelt de zee met bedreigingen zoals chemische vervuilingen, waaronder zware metalen en microplastics, die het mariene leven en de menselijke gezondheid beïnvloeden. Even belangrijk zijn maatregelen om de effecten van klimaatverandering te beperken, die verdere veranderingen in het ecosysteem kunnen veroorzaken.

Conclusie
De Oostzee is niet alleen een prachtig maar ook een uiterst belangrijk waterlichaam, waarvan het welzijn en de gezondheid van vele gemeenschappen afhangen. Om haar voortbestaan voor toekomstige generaties te waarborgen, zijn verdere onderzoeken, nauwe internationale samenwerking en vastberaden acties voor milieubescherming noodzakelijk. Verantwoord beheer van natuurlijke hulpbronnen en duurzame ontwikkeling kunnen bijdragen aan de bescherming van dit unieke ecosysteem, waardoor het zich kan regenereren en zijn unieke biodiversiteit kan behouden.

7.9. Teksty napisane w językach włoskim i francuskim

7.9.1. Artykuł o Alpach – język polski

Alpy – europejski łańcuch górski pełen różnorodności
Alpy stanowią jedne z najbardziej znanych i najczęściej odwiedzanych gór na świecie. Rozciągają się na długość ponad 1200 kilometrów przez osiem krajów europejskich: Francję, Szwajcarię, Włochy, Monako, Liechtenstein, Austrię, Niemcy oraz Słowenię. Charakteryzują się niezwykłą różnorodnością geograficzną, kulturową oraz ekologiczną, co czyni je wyjątkowym miejscem zarówno dla miłośników przyrody, jak i entuzjastów aktywnego wypoczynku.

Geografia i klimat
Alpy są najwyższym łańcuchem górskim w Europie, z najwyższym szczytem Mont Blanc, który osiąga wysokość 4810 metrów n.p.m. Góry te dzielą się na kilka głównych sekcji: Alpy Zachodnie, Centralne i Wschodnie, które różnią się geologicznie i topograficznie. Alpy Zachodnie są najbardziej masywne, z najwyższymi szczytami, podczas gdy Alpy Wschodnie są niższe i bardziej zróżnicowane.

Klimat w Alpach jest równie zmienny i zależy od wysokości oraz lokalizacji. Niższe partie charakteryzują się klimatem umiarkowanym, podczas gdy wyższe rejony są zdominowane przez warunki alpejskie, z chłodnymi temperaturami i obfitymi opadami śniegu, co sprzyja rozwojowi licznych kurortów narciarskich.

Flora i fauna
Bioróżnorodność Alp jest imponująca – region ten jest domem dla ponad 13,000 gatunków roślin i wielu gatunków zwierząt. Flora Alp zmienia się wraz z wysokością, od bujnych lasów liściastych i iglastych na niższych wysokościach, po alpejskie łąki pełne kwiatów i skąpe roślinność na wyższych. Fauna obejmuje takie gatunki jak kozica alpejska, świstak, orzeł przedni oraz niedźwiedź brunatny, choć ten ostatni jest już rzadko spotykany.

Kultura i historia
Alpy są także regionem o bogatej historii i kulturze. Od dawnych czasów góry te były miejscem, gdzie krzyżowały się różne kultury europejskie, co widoczne jest w architekturze, tradycjach oraz językach regionalnych. Alpejskie doliny są pełne malowniczych wiosek i miasteczek, gdzie lokalne zwyczaje są pielęgnowane z wielkim szacunkiem.

Turystyka i sport

Alpy są rajem dla miłośników sportów zimowych – od narciarstwa i snowboardingu, po bardziej ekstremalne formy aktywności jak lodowcowe wspinaczki. Latem góry te przyciągają miłośników trekkingu, wspinaczki skałkowej, paralotniarstwa oraz wielu innych form aktywnego wypoczynku. Co roku miliony turystów odwiedzają Alpy, co czyni je jednym z najważniejszych ośrodków turystycznych w Europie.

Ochrona środowiska
Zarówno zwiększająca się liczba turystów, jak i zmiany klimatyczne stanowią wyzwania dla Alp. Erozja gleby, topnienie lodowców oraz zanieczyszczenie są problemami, z którymi region ten musi się mierzyć. W odpowiedzi na te zagrożenia, wiele obszarów Alp zostało objętych ochroną w ramach różnych programów ochrony przyrody, mających na celu zachowanie unikalnego dziedzictwa naturalnego i kulturowego tego regionu.

Podsumowując, Alpy nie są tylko pasmem górskim – to złożony ekosystem, który wymaga zrozumienia, szacunku i ochrony. Jego wyjątkowe walory sprawiają, że jest to miejsce wyjątkowe na mapie Europy i świata, przyciągające ludzi swoim pięknem i różnorodnością przez cały rok.

7.9.2. Artykuł o Alpach – język francuski

Les Alpes – chaîne de montagnes européenne pleine de diversité
Les Alpes sont l'une des chaînes de montagnes les plus célèbres et les plus visitées au monde. Elles s'étendent sur plus de 1200 kilomètres à travers huit pays européens : la France, la Suisse, l'Italie, Monaco, le Liechtenstein, l'Autriche, l'Allemagne et la Slovénie. Elles se caractérisent par une diversité géographique, culturelle et écologique remarquable, ce qui en fait un lieu privilégié aussi bien pour les amoureux de la nature que pour les amateurs de loisirs actifs.

Géographie et climat
Les Alpes sont la plus haute chaîne de montagnes en Europe, avec le Mont Blanc comme point culminant à 4810 mètres d'altitude. Cette chaîne montagneuse se divise en plusieurs sections principales : les Alpes occidentales, centrales et orientales, qui diffèrent géologiquement et topographiquement. Les Alpes occidentales sont les plus massives avec les sommets les plus élevés, tandis que les Alpes orientales sont plus basses et plus diversifiées.

Le climat dans les Alpes est également variable et dépend de l'altitude et de la localisation. Les parties basses bénéficient d'un climat tempéré, tandis que les régions plus élevées sont dominées par des conditions alpines avec des températures fraîches et des chutes de neige abondantes, favorisant le développement de nombreux centres de ski.

Flore et faune
La biodiversité des Alpes est impressionnante – la région abrite plus de 13 000 espèces de plantes et de nombreuses espèces animales. La flore des Alpes varie avec l'altitude, des forêts denses de feuillus et de conifères dans les basses altitudes aux prairies alpines fleuries et à la végétation clairsemée dans les hauteurs. La faune comprend des espèces telles que le

bouquetin des Alpes, le marmot, l'aigle royal et l'ours brun, bien que ce dernier soit maintenant rarement rencontré.

Culture et histoire
Les Alpes sont également une région riche en histoire et en culture. Depuis des temps immémoriaux, ces montagnes ont été un carrefour où se croisaient différentes cultures européennes, ce qui est visible dans l'architecture, les traditions et les langues régionales. Les vallées alpines regorgent de villages pittoresques et de petites villes où les coutumes locales sont entretenues avec un grand respect.

Tourisme et sport
Les Alpes sont un paradis pour les amateurs de sports d'hiver – du ski et du snowboard aux formes plus extrêmes d'activités telles que l'escalade sur glacier. En été, ces montagnes attirent les amateurs de trekking, d'escalade, de parapente et de nombreuses autres formes de loisirs actifs. Chaque année, des millions de touristes visitent les Alpes, ce qui en fait l'un des principaux centres touristiques en Europe.

Protection de l'environnement
L'augmentation du nombre de touristes et les changements climatiques représentent des défis pour les Alpes. L'érosion des sols, la fonte des glaciers et la pollution sont des problèmes auxquels cette région doit faire face. En réponse à ces menaces, de nombreuses zones des Alpes ont été protégées dans le cadre de différents programmes de conservation de la nature, visant à préserver le patrimoine naturel et culturel unique de cette région.

En conclusion, les Alpes ne sont pas seulement une chaîne de montagnes – elles constituent un écosystème complexe qui nécessite compréhension, respect et protection. Leurs atouts uniques en font un lieu exceptionnel sur la carte de l'Europe et du monde, attirant les gens par leur beauté et leur diversité tout au long de l'année.

7.9.3. Artykuł o Alpach – język włoski

Le Alpi – Catena montuosa europea piena di diversità
Le Alpi sono una delle catene montuose più famose e visitate al mondo. Si estendono per oltre 1200 chilometri attraverso otto paesi europei: Francia, Svizzera, Italia, Monaco, Liechtenstein, Austria, Germania e Slovenia. Sono caratterizzate da una notevole diversità geografica, culturale ed ecologica, il che le rende un luogo privilegiato tanto per gli amanti della natura quanto per gli appassionati di attività all'aperto.

Geografia e clima
Le Alpi sono la catena montuosa più alta d'Europa, con la cima più elevata, il Monte Bianco, che raggiunge i 4810 metri sul livello del mare. Questa catena montuosa è divisa in varie sezioni principali: Alpi Occidentali, Centrali e Orientali, che differiscono geologicamente e topograficamente. Le Alpi Occidentali sono le più massicce con le vette più alte, mentre le Alpi Orientali sono più basse e variegate.

Il clima nelle Alpi è altrettanto variabile e dipende dall'altitudine e dalla posizione. Le parti più basse godono di un clima temperato, mentre le regioni più elevate sono dominate da condizioni alpine con temperature fresche e abbondanti nevicate, che favoriscono lo sviluppo di numerosi resort sciistici.

Flora e fauna
La biodiversità delle Alpi è impressionante – la regione ospita oltre 13.000 specie di piante e molte specie animali. La flora alpina cambia con l'altitudine, da fitte foreste di latifoglie e conifere nelle altitudini più basse a praterie alpine fiorite e vegetazione rada nelle zone più elevate. La fauna include specie come lo stambecco, la marmotta, l'aquila reale e l'orso bruno, sebbene quest'ultimo sia ora raramente incontrato.

Cultura e storia
Le Alpi sono anche una regione ricca di storia e cultura. Da tempi immemorabili, queste montagne sono state un crocevia di diverse culture europee, come si può vedere nell'architettura, nelle tradizioni e nelle lingue regionali. Le valli alpine sono piene di villaggi pittoreschi e piccole città dove le usanze locali sono mantenute con grande rispetto.

Turismo e sport
Le Alpi sono un paradiso per gli amanti degli sport invernali – dallo sci e lo snowboard a forme più estreme di attività come l'arrampicata su ghiacciaio. In estate, queste montagne attirano gli appassionati di trekking, arrampicata, parapendio e molte altre forme di attività all'aperto. Ogni anno, milioni di turisti visitano le Alpi, rendendole uno dei principali centri turistici in Europa.

Protezione dell'ambiente
L'aumento del numero di turisti e i cambiamenti climatici rappresentano sfide per le Alpi. L'erosione del suolo, lo scioglimento dei ghiacciai e l'inquinamento sono problemi che questa regione deve affrontare. In risposta a queste minacce, molte aree delle Alpi sono state protette attraverso vari programmi di conservazione della natura, volti a preservare il patrimonio naturale e culturale unico di questa regione.

In conclusione, le Alpi non sono solo una catena montuosa – sono un ecosistema complesso che richiede comprensione, rispetto e protezione. Le loro qualità uniche le rendono un luogo eccezionale sulla mappa d'Europa e del mondo, attirando persone per la loro bellezza e diversità tutto l'anno.

7.10. Teksty napisane w językach hiszpańskim i rumuńskim

7.10.1. Artykuł encyklopedyczny o Hiszpanii – język hiszpański

España, también denominado Reino de España,nota 1 es un país soberano transcontinental, miembro de la Unión Europea, constituido en Estado social y democrático de derecho, cuya forma de gobierno es la monarquía parlamentaria. Su territorio, con capital en Madrid,30 está organizado en diecisiete comunidades autónomas, formadas a su vez por cincuenta provincias; y dos ciudades autónomas.

España se sitúa tanto al sur de Europa Occidental como en el norte de África. En Europa, ocupa la mayor parte de la península ibérica, conocida como España peninsular, y las islas Baleares (en el mar Mediterráneo occidental); en África se hallan las ciudades de Ceuta (en la península tingitana) y Melilla (en el cabo de Tres Forcas), las islas Canarias (en el océano Atlántico nororiental) y otras posesiones mediterráneas denominadas «plazas de soberanía». El municipio de Llivia, en los Pirineos, constituye un exclave rodeado totalmente por territorio francés. Completa el conjunto de territorios una serie de islas e islotes frente a las propias costas peninsulares. Tiene una extensión de 505 370 km²,11 por lo que es el cuarto país más extenso del continente, tras Rusia, Ucrania y Francia,nota 2 y con una altitud media de 650 metros sobre el nivel del mar, uno de los países más montañosos de Europa. Su población supera los 47 millones de habitantes, aunque la densidad de población es reducida.31 13 El territorio peninsular comparte fronteras terrestres con Francia y con Andorra al norte, con Portugal al oeste y con el territorio británico de Gibraltar al sur. En sus territorios africanos, comparte fronteras terrestres y marítimas con Marruecos. Comparte con Francia la soberanía sobre la isla de los Faisanes en la desembocadura del río Bidasoa y cinco facerías pirenaicas.32

De acuerdo con la Constitución, y según su artículo 3.1, «el castellano es la lengua española oficial del Estado. Todos los españoles tienen el deber de conocerla y el derecho a usarla».3 En 2012, era la lengua materna del 82 % de los españoles.33 Según el artículo 3.2, «las demás lenguas españolas serán también oficiales en las respectivas Comunidades Autónomas de acuerdo con sus Estatutos».3 El idioma español o castellano, segunda lengua materna más hablada del mundo y con casi 600 millones de hispanohablantes,34 es uno de los más importantes legados del acervo cultural e histórico de España en el mundo. Perteneciente culturalmente a la Europa Latina y heredero de una vasta influencia grecorromana, España alberga también la cuarta colección más numerosa del mundo de sitios declarados Patrimonio de la Humanidad por la Unesco.35

Es un país desarrollado —goza de la segunda esperanza de vida más elevada del mundo— y de altos ingresos, cuyo PIB coloca a la economía española en la decimocuarta posición mundial (2021).36 Gracias a sus características únicas, España es una gran potencia turística y se erige como el segundo país más visitado del mundo —más de 83 millones de turistas en 2019— y el segundo país del mundo en ingresos económicos provenientes del turismo internacional.3738 Tiene un índice de desarrollo humano muy alto (0,904), según el informe de 2020 del Programa de la ONU para el Desarrollo.15 España también tiene una notable proyección internacional a través de su pertenencia a múltiples organizaciones internacionales como Naciones Unidas, el Consejo de Europa, la Organización Mundial del Comercio, la Organización de Estados Iberoamericanos, la OCDE, la OTAN y la Unión Europea —incluidos dentro de esta al espacio Schengen y la Eurozona—, además de ser miembro de facto del G20.

La primera presencia constatada de homínidos del género Homo se remonta a 1,2 millones de años antes del presente, como atestigua el descubrimiento de una mandíbula de un Homo aún sin clasificar en el yacimiento de

Atapuerca.39 En el siglo III a. C., se produjo la intervención romana en la Península, lo que conllevó a una posterior conquista de lo que, más tarde, se convertiría en Hispania. En el Medievo, la zona fue conquistada por distintos pueblos germánicos y por los musulmanes, llegando estos a tener presencia durante algo más de siete centurias. No es hasta el siglo XV, con la unión dinástica de Castilla y Aragón y la culminación de la Reconquista, junto con la posterior anexión navarra, cuando se puede hablar de la cimentación de «España», como era denominada en el exterior.404142 Ya en la Edad Moderna, los monarcas españoles dominaron el primer imperio de ultramar global, que abarcaba territorios en los cinco continentes,nota 3 dejando un vasto acervo cultural y lingüístico por el globo. A principios del xix, tras sucesivas guerras en Hispanoamérica, pierde la mayoría de sus territorios en América, acrecentándose esta situación con el desastre del 98.

7.10.2. Artykuł encyklopedyczny o Hiszpanii – język rumuński

Spania, cunoscută și sub denumirea de Regatul Spaniei, nota 1 este o țară suverană transcontinentală, membră a Uniunii Europene, constituită ca stat social și democratic de drept, a cărui formă de guvernare este monarhia parlamentară. Teritoriul său, cu capitala la Madrid30, este organizat în șaptesprezece comunități autonome, formate la rândul lor din cincizeci de provincii; și două orașe autonome.

Spania este situată atât în sudul Europei de Vest, cât și în nordul Africii. În Europa, ocupă cea mai mare parte a Peninsulei Iberice, cunoscută drept Spania continentală, și Insulele Baleare (în vestul Mării Mediterane); În Africa se află orașele Ceuta (în peninsula Tingitana) și Melilla (în capul Tres Forcas), Insulele Canare (în nord-estul Oceanului Atlantic) și alte posesiuni mediteraneene numite „pătrate ale suveranității”. Municipiul Llivia, din Pirinei, constituie o exclavă complet înconjurată de teritoriul francez. Setul de teritorii este completat de o serie de insule și insulițe în largul coastelor peninsulare. Are o suprafață de 505.370 km²11, făcând-o a patra țară ca mărime de pe continent, după Rusia, Ucraina și Franța, nota 2, și cu o altitudine medie de 650 de metri deasupra nivelului mării, una dintre cele mai muntoase țări din Europa. Populația sa depășește 47 de milioane de locuitori, deși densitatea populației este scăzută.31 13 Teritoriul peninsular împarte granițe terestre cu Franța și cu Andorra la nord, cu Portugalia la vest și cu teritoriul britanic Gibraltar la sud. În teritoriile sale africane, are granițe terestre și maritime cu Marocul. Împărtășește cu Franța suveranitatea asupra insulei Fazanilor de la gura de vărsare a râului Bidasoa și a cinci facerías pirineene.32

În conformitate cu Constituția și potrivit articolului 3.1 al acesteia, „Castilia este limba oficială spaniolă a statului. Toți spaniolii au datoria de a o cunoaște și dreptul de a o folosi.”3 În 2012, era limba maternă a 82% dintre spanioli.33 Potrivit articolului 3.2, „celelalte limbi spaniole vor fi, de asemenea, oficiale în Comunitățile respective.Autonome în conformitate cu Statutele sale”.3 Limba spaniolă sau castiliană, a doua cea mai vorbită limbă maternă din lume și cu aproape 600 de milioane de vorbitori de spaniolă,34 este una dintre cele mai

importante moșteniri ale patrimoniului cultural și istoric al Spania in lume... Aparținând cultural Europei Latine și moștenitoarea unei vaste influențe greco-romane, Spania găzduiește, de asemenea, cea de-a patra cea mai mare colecție de situri din patrimoniul mondial UNESCO din lume.35

Este o țară dezvoltată - se bucură de a doua cea mai mare speranță de viață din lume - și de venituri mari, al cărei PIB plasează economia spaniolă pe locul paisprezecea în lume (2021).36 Datorită caracteristicilor sale unice, Spania este o mare putere turistică. și se situează ca a doua țară cea mai vizitată din lume — peste 83 de milioane de turiști în 2019 — și a doua țară din lume ca venituri economice din turismul internațional.3738 Are un indice de dezvoltare umană foarte ridicat (0,904), conform datelor Raportul 2020 al Programului ONU pentru Dezvoltare.15 Spania are, de asemenea, o proiecție internațională notabilă prin apartenența sa la mai multe organizații internaționale precum Națiunile Unite, Consiliul Europei, Organizația Mondială a Comerțului, Organizația Statelor Ibero-Americane, OCDE, NATO și Uniunea Europeană — inclusiv în cadrul acesteia spațiul Schengen și zona euro—, pe lângă faptul că sunt de facto membră a G20.

Prima prezență confirmată a hominidelor din genul Homo datează cu 1,2 milioane de ani înainte de prezent, fapt dovedit de descoperirea unei mandibule Homo încă neclasificate la situl Atapuerca.39 În secolul al III-lea î.Hr. C., a avut loc intervenția romană în Peninsula, care a dus la o cucerire ulterioară a ceea ce avea să devină mai târziu Hispania. În Evul Mediu, zona a fost cucerită de diferite popoare germanice și de musulmani, care au avut prezență de mai bine de șapte secole. Abia în secolul al XV-lea, odată cu unirea dinastică a Castiliei și Aragonului și a punctului culminant al Recuceririi, odată cu anexarea ulterioară a Navarei, se poate vorbi de întemeierea „Spaniei", așa cum era numită în străinătate.40 4142 Deja în epoca modernă, monarhii spanioli dominau primul imperiu global de peste mări, care cuprindea teritorii de pe cinci continente, nota 3 lăsând o vastă moștenire culturală și lingvistică pe tot globul. La începutul secolului al XIX-lea, după războaie succesive din America Latină, și-a pierdut majoritatea teritoriilor din America, sporind această situație odată cu dezastrul din 1998.

7.10.3. Artykuł encyklopedyczny o Rumunii – język hiszpański

Rumania, país del sudeste de Europa. La capital nacional es Bucarest. Rumania fue ocupada por las tropas soviéticas en 1944 y se convirtió en un satélite de la Unión de Repúblicas Socialistas Soviéticas (URSS) en 1948. El país estuvo bajo el régimen comunista desde 1948 hasta 1989, cuando el régimen del líder rumano Nicolae Ceaușescu fue derrocado. En 1990 se celebraron elecciones libres. En 2004 el país se unió a la Organización del Tratado del Atlántico Norte (OTAN), y en 2007 se convirtió en miembro de la Unión Europea (UE).

El paisaje rumano es aproximadamente un tercio montañoso y un tercio boscoso, con el resto formado por colinas y llanuras. El clima es templado y marcado por cuatro estaciones distintas. Rumania goza de una considerable riqueza de recursos naturales: tierras fértiles para la agricultura;

pastos para el ganado; bosques que proporcionan maderas duras y blandas; reservas de petróleo; metales, incluyendo oro y plata en las montañas Apuseni; numerosos ríos que suministran hidroelectricidad; y una costa del Mar Negro que es el sitio de puertos y centros turísticos.

El pueblo rumano deriva gran parte de su carácter étnico y cultural de la influencia romana, pero esta antigua identidad ha sido remodelada continuamente por la posición de Rumania a horcajadas sobre las principales rutas de migración continental. Los rumanos se consideran a sí mismos como los descendientes de los antiguos romanos que conquistaron el sur de Transilvania bajo el emperador Trajano en 105 CE y de los dacios que vivían en las montañas al norte de la llanura del Danubio y en la cuenca de Transilvania. En el momento de la retirada romana bajo el emperador Aureliano en 271, los colonos romanos y los dacios se habían casado entre sí, lo que resultó en una nueva nación. Tanto las raíces latinas de la lengua rumana como la fe ortodoxa oriental a la que se adhieren la mayoría de los rumanos surgieron de la mezcla de estas dos culturas.

7.10.4. Artykuł encyklopedyczny o Rumunii – język rumuński

România, țară din sud-estul Europei. Reședința sa este orașul București. România a fost ocupată de trupele sovietice în 1944 și a devenit satelit al Uniunii Republicilor Sovietice Socialiste (U.S.S.R.) în 1948. Țara a fost sub dominație comunistă din 1948 până în 1989, când regimul liderului român Nicolae Ceaușescu a fost răsturnat. Alegerile libere au avut loc în 1990. În 2004, țara a aderat la Organizația Tratatului Atlanticului de Nord (NATO), iar în 2007 a devenit membră a Uniunii Europene (UE).

Peisajul românesc este de aproximativ o treime muntos și o treime împădurit, restul fiind alcătuit din dealuri și câmpii. Clima este temperată și marcată de patru anotimpuri distincte. România se bucură de o bogăție considerabilă de resurse naturale: terenuri fertile pentru agricultură; pășuni pentru animale; păduri care asigură păduri dure și moi; rezervele de petrol; metale, inclusiv aur și argint în Munții Apuseni; numeroase râuri care furnizează energie hidroelectrică; și o coastă a Mării Negre care este locul atât al porturilor, cât și al stațiunilor.

Poporul român derivă o mare parte din caracterul său etnic și cultural din influența romană, dar această identitate străveche a fost remodelată continuu de poziția României pe principalele rute continentale de migrație. Românii se consideră descendenții vechilor romani care au cucerit sudul Transilvaniei sub împăratul Traian în anul 105 și ai dacilor care locuiau în munții de la nord de Câmpia Dunăreană și în Bazinul Transilvaniei. Până la retragerea romană sub împăratul Aurelian în 271, coloniștii romani și dacii s-au căsătorit, rezultând o nouă națiune. Atât rădăcinile latine ale limbii române, cât și credința ortodoxă răsăriteană la care aderă majoritatea românilor au apărut din amestecul acestor două culturi.